A Cut-Free Proof Theory for Boolean BI (via Display Logic)

James Brotherston*
Dept. of Computing
Imperial College, London, UK

August 3, 2009

Abstract

We give a display calculus proof system for Boolean BI (BBI) based on Belnap's general display logic. We show that cut-elimination holds in our system and that it is sound and complete with respect to the usual notion of validity for BBI. We then show how to constrain proof search in the system (without loss of generality) by means of a series of proof transformations. By doing so, we gain some insight into the problem of decidability for BBI.

1 Introduction

O'Hearn and Pym's logic of bunched implications BI [16] is a well-known substructural logic whose formulas have a natural declarative interpretation as statements about *resource*. Formulas of the logic freely combine the standard additive units and connectives of propositional logic $(\top, \bot, \neg, \wedge, \vee, \rightarrow)$ with a multiplicative (a.k.a. "linear") unit $\top^*$, conjunction $*$ and implication $-\!\!*$. There are two main flavours of BI, induced by two alternative treatments of the additive connectives. If the latter are interpreted intuitionistically, then the resulting logic is called (intuitionistic) BI, while the variant in which they are instead considered classically is known as Boolean BI (BBI).

There has been considerable interest in the theory and application of BI and its variants among the computer science research community over the last few years, primarily because the logic's concise representation of resource makes it a viable basis for reasoning about resource-manipulating programs [12]. Strikingly, however, theoretical developments in BI have focused for the most part on its intuitionistic version, whereas the numerous program verification applications of BI — notably separation logic [20] and its spin-offs, including various flavours of shape analysis [4, 5, 6, 7] — are mainly based on its Boolean variant.

*Research supported by an EPSRC postdoctoral fellowship.

These applications typically rely heavily on theorem-proving (e.g., in order to remove redundant disjuncts of an invariant in shape analysis), and stand to benefit substantially from further development of the theory of BBI. Indeed, of four basic theoretical questions about BBI — completeness with respect to a class of models, cut-elimination, decidability and the existence of a well behaved type theory — only the first has so far been answered. In this paper, we give a positive answer to the second question, and in doing so make progress towards establishing the status of the third.

We consider BBI from the proof-theoretic viewpoint. For intuitionistic BI there is both a complete natural deduction proof system satisfying normalisation, and a complete sequent calculus satisfying cut-elimination [17]. In contrast, no such well-behaved syntactic proof systems for BBI presently exist in the literature[1]. Rather, proof systems for BBI are usually obtained by adding a suitable axiom or inference rule (e.g. the axiom $\neg\neg F \vdash F$) to the corresponding proof system for BI. Such additions typically break normalisation and cut-elimination properties, which is less than ideal both from the point of view of BBI as a *bona fide* logic in its own right and from the perspective of the aforementioned program analysis tools based on BBI. However, extending the BI sequent calculus (cf. [17]) to BBI without breaking cut-elimination in the process is problematic. The contexts in this calculus are "bunches", which are syntax trees whose leaves are formulas and whose internal nodes are labelled by either a semicolon or a comma, denoting respectively additive and multiplicative conjunction at the structural level. Weakening and contraction are then permitted with respect to the additive semicolon, but not the multiplicative comma. In order to cope with the tree structure of bunches, the left-introduction rules for logical connectives are formulated so as to apply at arbitrary positions within a bunch, whereas the right-introduction rules operate at only the top level of bunches due to the intuitionistic presentation. E.g., the rules for the conjunction $*$ are:

$$\frac{\Gamma(F_1, F_2) \vdash F}{\Gamma(F_1 * F_2) \vdash F}\,(*\mathrm{L}) \qquad \frac{\Gamma_1 \vdash F_1 \quad \Gamma_2 \vdash F_2}{\Gamma_1, \Gamma_2 \vdash F_1 * F_2}\,(*\mathrm{R})$$

where $\Gamma(\Delta)$ denotes a bunch Γ with a distinguished sub-bunch occurrence Δ. It is not obvious how to extend this calculus to the Boolean setting. On the one hand it seems clear that, as in Gentzen's sequent calculus LK for classical logic, the sequents should support multiple conclusions in some form (otherwise cut-free proofs of classical tautologies such as $\neg\neg F \vdash F$ are impossible). On the other hand, it is far from clear how to formulate multiple-conclusion versions of the rules for the multiplicative connectives or of the usual left-introduction rule for negation, owing to the failure of various distribution properties in BBI (e.g. $*$ distributes over $\vee$ but not vice versa). Pym reports that two-sided bunched systems suffer from problems with cut-elimination [17], but in any event it is not obvious how even to interpret bunched conclusions in BBI.

[1] A tableau proof system for a version of BBI based on functional rather than the usual relational monoidal semantics is given in [14], but it does not correspond straightforwardly to a cut-free sequent calculus.

Instead of a sequent calculus, we give in this paper a Belnap-style *display calculus* proof system for BBI. Display calculi can be seen as generalised sequent calculi, facilitating the combination of arbitrarily many "families" of logical connectives, each such family being assigned a corresponding set of structural connectives. This results in a very rich meta-level for proof judgements, and to cope with this one postulates special structural rules, called "display rules', which ensure that any proof judgement can be rearranged so as to "display" (i.e. bring to the top level) any given part of the judgement. Our system, $\mathrm{DL}_{\mathrm{BBI}}$, largely follows our previous formulation, with Calcagno, of a display calculus for CBI, which is a nonconservative extension of BBI including multiplicative versions of falsity, negation and disjunction [3]. However, $\mathrm{DL}_{\mathrm{BBI}}$ differs from this calculus, and other standard display calculi, in that its proof judgements are syntactically (and semantically) asymmetric, reflecting the lack of symmetry in the multiplicative connectives of BBI. Despite this asymmetry, $\mathrm{DL}_{\mathrm{BBI}}$ nevertheless enjoys the crucial display property, from which cut-elimination for $\mathrm{DL}_{\mathrm{BBI}}$ follows as a straightforward corollary of Belnap's cut-elimination proof for general display logic [1]. Soundness is an easy direct result, and completeness follows from the reduction of provability in $\mathrm{DL}_{\mathrm{BBI}}$ to provability in a Hilbert-style proof system which is known to be complete.

Although cut-elimination for $\mathrm{DL}_{\mathrm{BBI}}$ entails a traditional subformula property, it is clear that naive approaches to proof search in $\mathrm{DL}_{\mathrm{BBI}}$ will lead to divergence, due to the evident need to employ the aforementioned display rules and the usual structural rules such as contraction. The complexity arising in proof search is not unexpected since, by soundness and completeness, decidability of provability in $\mathrm{DL}_{\mathrm{BBI}}$ is equivalent to decidability of validity in BBI, which has been an open problem for some time. (One might be tempted to hope that it is possible to know whether any given display calculus is decidable merely by looking at it, but Kracht showed that this question is itself undecidable [13].) However, by adapting techniques from Restall [18], we show how to constrain proof search in $\mathrm{DL}_{\mathrm{BBI}}$ via a series of completeness-preserving proof reductions (of which cut-elimination is the first stage). In contrast to the situation for the relevant logics considered by Restall, however, the analysis fails to establish decidability of full BBI, seemingly due to the presence of Boolean negation. Nevertheless, we argue that our results ought to be of use in implementing proof search strategies for BBI.

The remainder of this paper is structured as follows. In Section 2 we give a brief overview of BBI. We formulate our display calculus $\mathrm{DL}_{\mathrm{BBI}}$ in Section 3, and outline the arguments for cut-elimination, soundness and completeness. In Section 4 we present our proof reduction strategy for $\mathrm{DL}_{\mathrm{BBI}}$, and discuss its implications for decidability of BBI and its fragments. Finally, in Section 5 we conclude with some comments and suggestions regarding future work.

2 Syntax and semantics of BBI

In this section we briefly recall the syntax of BBI and its standard algebraic semantics. We also recall a Hilbert-style proof system for BBI which is sound and complete with respect to the algebraic semantics.

We assume a fixed infinite set $\mathcal{V}$ of propositional variables, and write $\mathrm{Pow}(X)$ for the powerset of a set X.

Definition 2.1 (Formula)**.** *Formulas* of BBI are given by the following grammar, where P ranges over $\mathcal{V}$:

$$F ::= P \mid \top \mid \bot \mid \neg F \mid F \wedge F \mid F \vee F \mid F \to F \mid \top^* \mid F * F \mid F \mathrel{-\!\!*} F$$

We treat $\wedge$, $\vee$ and $*$ as having greater precedence than $\to$ and $\mathrel{-\!\!*}$.

Definition 2.2 (BBI-model)**.** A BBI-model is a relational commutative monoid $\langle R, \circ, e\rangle$, where $e \in R$ and $\circ : R \times R \to \mathrm{Pow}(R)$ are such that $\circ$ is commutative, with $r \circ e = \{r\}$ for all $r \in R$. We extend $\circ$ to $\mathrm{Pow}(R) \times \mathrm{Pow}(R)$ by $X \circ Y = \bigcup_{x \in X, y \in Y} x \circ y$. Under this extension, $\circ$ is required to be associative.

A BBI-model $\langle R, \circ, e\rangle$ is typically seen as an abstract model of resource, where R is a set of resources, $\circ$ denotes nondeterministic resource combination and e is the distinguished empty resource. An *environment* for $\langle R, \circ, e\rangle$ is a map $\rho : \mathcal{V} \to \mathrm{Pow}(R)$ interpreting propositional variables as subsets of R.

Definition 2.3 (Satisfaction)**.** Let $M = \langle R, \circ, e\rangle$ be a BBI-model. *Satisfaction* of a formula F by an environment ρ for M and a "resource" $r \in R$ is denoted $r \models_\rho F$ and defined by structural induction on F as follows:

$$\begin{array}{rcl}
r \models_\rho P & \Leftrightarrow & r \in \rho(P) \\
r \models_\rho \top & \Leftrightarrow & \text{always} \\
r \models_\rho \bot & \Leftrightarrow & \text{never} \\
r \models_\rho \neg F & \Leftrightarrow & r \not\models_\rho F \\
r \models_\rho F_1 \wedge F_2 & \Leftrightarrow & r \models_\rho F_1 \text{ and } r \models_\rho F_2 \\
r \models_\rho F_1 \vee F_2 & \Leftrightarrow & r \models_\rho F_1 \text{ or } r \models_\rho F_2 \\
r \models_\rho F_1 \to F_2 & \Leftrightarrow & r \models_\rho F_1 \text{ implies } r \models_\rho F_2 \\
r \models_\rho \top^* & \Leftrightarrow & r = e \\
r \models_\rho F_1 * F_2 & \Leftrightarrow & \exists r_1, r_2.\ r \in r_1 \circ r_2 \text{ and } r_1 \models_\rho F_1 \text{ and } r_2 \models_\rho F_2 \\
r \models_\rho F_1 \mathrel{-\!\!*} F_2 & \Leftrightarrow & \forall r', r''.\ \text{if } r'' \in r \circ r' \text{ and } r' \models_\rho F_1 \text{ then } r'' \models_\rho F_2
\end{array}$$

Definition 2.4 (Formula validity)**.** A formula F is said to be *true* in a BBI-model $M = \langle R, \circ, e\rangle$ if for any environment ρ for M and for all $r \in R$, we have $r \models_\rho F$. F is said to be *valid* if it is true in all BBI-models.

$\mathrm{HL}_{\mathrm{BBI}}$, the standard Hilbert-style proof system for BBI (cf. [17, 9]) is obtained by extending some fixed finite axiomatisation of classical propositional logic by the following axioms and inference rules:

$$\begin{array}{ll}
\text{(Ax 1)} & \vdash F \to \top^* * F \\
\text{(Ax 2)} & \vdash \top^* * F \to F \\
\text{(Ax 3)} & \vdash F * G \to G * F \\
\text{(Ax 4)} & \vdash F * (G * H) \to (F * G) * H
\end{array}$$

$$\frac{\vdash F \quad \vdash F \to G}{\vdash G}\,(\mathrm{MP}) \qquad \frac{\vdash F_1 \to G_1 \quad \vdash F_2 \to G_2}{\vdash F_1 * F_2 \to G_1 * G_2}\,(*)$$

$$\frac{\vdash F \to (G \mathrel{-\!\!*} H)}{\vdash (F * G) \to H}\,(\mathrel{-\!\!*}1) \qquad \frac{\vdash (F * G) \to H}{\vdash F \to (G \mathrel{-\!\!*} H)}\,(\mathrel{-\!\!*}2)$$

The following result is due to Galmiche and Larchey-Wendling (and independently proved by Yang).

Theorem 2.5 (Completeness [9])**.** *Any valid formula is* $\mathrm{HL}_{\mathrm{BBI}}$*-provable.*

3 $\mathrm{DL}_{\mathrm{BBI}}$: a display calculus for BBI

In this section we formulate a Belnap-style display calculus proof system for BBI. Our display calculus, $\mathrm{DL}_{\mathrm{BBI}}$, departs slightly from traditional display calculi in that the form of proof judgements is not symmetric. Nevertheless, our proof system admits the crucial display property, and satisfies cut-elimination.

The judgements of our proof system, called consecutions, are built from structures which generalise the "bunches" used in proof systems for BI (cf. [17]).

Definition 3.1 (Structure / Consecution)**.** *A-structures X* and *C-structures Y* are (mutually) defined by the grammar:

$$\begin{array}{rcl} X & ::= & F \mid \emptyset \mid \sharp Y \mid X;X \mid \varnothing \mid X,X \\ Y & ::= & F \mid \emptyset \mid \sharp X \mid Y;Y \mid X \multimap Y \end{array}$$

where F ranges over BBI-formulas. We use the generic name *structure* to refer to both A-structures and C-structures. If X is an A-structure and Y is a C-structure then $X \vdash Y$ is said to be a *consecution.*

Roughly speaking, A-structures and C-structures represent respectively *antecedent parts* and *consequent parts* of our consecutions (we defer a formal definition of these terms until later). The asymmetry between the two is caused by the absence of multiplicative versions of falsity, negation and disjunction in BBI, which leads us to employ a non-standard structural connective $\multimap$ in C-structures, interpreted as multiplicative implication $\mathrel{-\!\!*}$. The following definition gives the interpretation of our consecutions.

Definition 3.2 (Consecution validity)**.** For any consecution $X \vdash Y$ we define the BBI-formulas Ψ_X and Υ_Y by mutual structural induction as follows:

$$\begin{array}{rcl@{\qquad}rcl} \Psi_F & = & F & \Upsilon_F & = & F \\ \Psi_\emptyset & = & \top & \Upsilon_\emptyset & = & \bot \\ \Psi_{\sharp Y} & = & \neg\Upsilon_Y & \Upsilon_{\sharp X} & = & \neg\Psi_X \\ \Psi_{X_1;X_2} & = & \Psi_{X_1} \wedge \Psi_{X_2} & \Upsilon_{Y_1;Y_2} & = & \Upsilon_{Y_1} \vee \Upsilon_{Y_2} \\ \Psi_\varnothing & = & \top^* & \Upsilon_{X \multimap Y} & = & \Psi_X \mathrel{-\!\!*} \Upsilon_Y \\ \Psi_{X_1,X_2} & = & \Psi_{X_1} * \Psi_{X_2} & & & \end{array}$$

$X \vdash Y$ is said to be *valid* iff $\Psi_X \to \Upsilon_Y$ is a valid formula.

A consecution S is said to be *derivable from* another consecution S' (using some fixed set of proof rules) if, given a derivation of S', one can construct a derivation of S. That is, there is a derivation whose root is S and whose leaves are either axioms or occurrences of S'. The consecution S is said to be *interderivable with* S' if S is derivable from S' and vice versa. We write proof rules with a double-line between premise and conclusion to indicate that they are invertible, i.e. that the roles of premise and conclusion may be reversed.

Definition 3.3 (Display-equivalence)**.** Two consecutions $X \vdash Y$ and $X' \vdash Y'$ are said to be *display-equivalent*, written $X \vdash Y \equiv_D X' \vdash Y'$, if they are interderivable using only the following *display rules*:

$$\begin{array}{c} X;Y \vdash Z \\ \hline\hline X \vdash \sharp Y;Z \\ \hline\hline Y;X \vdash Z \end{array}\begin{array}{l} \text{(AD1a)} \\ \\ \text{(AD1b)} \end{array} \quad \begin{array}{c} X \vdash Y;Z \\ \hline\hline X;\sharp Y \vdash Z \\ \hline\hline X \vdash Z;Y \end{array}\begin{array}{l} \text{(AD2a)} \\ \\ \text{(AD2b)} \end{array} \quad \begin{array}{c} X \vdash Y \\ \hline\hline \sharp Y \vdash \sharp X \\ \hline\hline \sharp\sharp X \vdash Y \end{array}\begin{array}{l} \text{(AD3a)} \\ \\ \text{(AD3b)} \end{array} \quad \begin{array}{c} X,Y \vdash Z \\ \hline\hline X \vdash Y \multimap Z \\ \hline\hline Y,X \vdash Z \end{array}\begin{array}{l} \text{(MD1a)} \\ \\ \text{(MD1b)} \end{array}$$

We remark that $\equiv_D$ is indeed an equivalence relation.

Definition 3.4 (Antecedent and consequent parts)**.** We classify the substructure occurrences in a structure X as either *positive* or *negative* in X as follows:

- X is positive in X;
- if Z is negative (positive) in X' then Z is positive (negative) in $\sharp X'$;
- if Z is positive (negative) in X_1 or X_2 then Z is positive (negative) in $X_1;X_2$ and X_1,X_2;
- if Z is negative (positive) in X_1 or positive (negative) in X_2, then Z is positive (negative) in $X_1 \multimap X_2$.

Z is said to be an *antecedent (consequent) part* of a consecution $X \vdash Y$ if it is positive (negative) in X or negative (positive) in Y.

Definition 3.5 (Height)**.** Let Z be a substructure occurrence in a structure X. Define the *height of Z in X* as the length of the path from the root of X to the root of Z, when structures are viewed as trees. If Z is a structure occurrence in a consecution $X \vdash Y$, then define the *height of Z in $X \vdash Y$* as the height of Z in X if Z is a substructure occurrence in X and the height of Z in Y otherwise.

Lemma 3.6. *Let Z be a structure occurrence with height $h \neq 0$ in a consecution $X \vdash Y$. There is a consecution $X' \vdash Y'$ that is display-equivalent to $X \vdash Y$ and such that Z has height $< h$ in $X' \vdash Y'$. Moreover, Z is an antecedent (consequent) part of $X' \vdash Y'$ iff it is an antecedent (consequent) part of $X \vdash Y$.*

Proof. We proceed by cases as follows.

Case Z occurs in X. We observe that since $h \neq 0$, we have $Z \neq X$ and so X is not of the form F, $\emptyset$ or $\varnothing$. The remaining subcases are treated as follows:

Subcase $X = \sharp W$. We proceed as follows:

$$\dfrac{\dfrac{\dfrac{\dfrac{\sharp W \vdash Y}{\sharp Y \vdash \sharp\sharp W}\,(\text{AD3a})}{\sharp\sharp\sharp W \vdash \sharp\sharp Y}\,(\text{AD3b})}{\sharp W \vdash \sharp\sharp Y}\,(\text{AD3a,b})}{\sharp Y \vdash W}\,(\text{AD3a})$$

and thus set $X' \vdash Y' = \sharp Y \vdash W$, which is clearly display-equivalent to $X \vdash Y = \sharp W \vdash Y$ as required. Also note that if Z is an antecedent (consequent) part of $\sharp W \vdash Y$ then it is a positive (negative) part of $\sharp W$, thus a negative (positive) part of W and so an antecedent (consequent) part of $\sharp Y \vdash W$. Furthermore, Z has a lower height in W than in $\sharp W$, thus a lower height in $\sharp Y \vdash W$ than in $\sharp W \vdash Y$, as required.

Subcase $X = W_1; W_2$. If Z occurs in W_1 then we proceed as follows:

$$\dfrac{W_1; W_2 \vdash Y}{W_1 \vdash \sharp W_2; Y}\,(\text{AD1a})$$

Otherwise, Z occurs in W_2 and we instead use rule (AD1b) in a similar fashion. The justification that the required properties hold is then similar to the subcase above.

Subcase $X = W_1, W_2$. If Z occurs in W_1 then we proceed as follows:

$$\dfrac{W_1, W_2 \vdash Y}{W_1 \vdash W_2 \multimap Y}\,(\text{MD1a})$$

Otherwise, Z occurs in W_2 and we instead use rule (MD1b) in a similar fashion. The justification that the required properties hold is then similar to the subcase above. This completes the case.

Case Z occurs in Y. Since $h \neq 0$, in particular $Z \neq Y$ and so Y is not of the form F or $\emptyset$. The remaining subcases are treated as follows:

Subcase $Y = \sharp W$. We proceed as follows:

$$\dfrac{\dfrac{X \vdash \sharp W}{\sharp\sharp W \vdash \sharp X}\,(\text{AD3a})}{W \vdash \sharp X}$$

The justification that the required properties hold is similar to the cases above.

Subcase $Y = W_1; W_2$. If Z occurs in W_1 then we proceed as follows:

$$\frac{X \vdash W_1; W_2}{X; \sharp W_2 \vdash W_1} \text{(AD2b)}$$

Otherwise, Z occurs in W_2 and we instead use rule (AD2a) in a similar fashion. The justification that the required properties hold is then similar to the cases above.

Subcase $Y = W_1 \multimap W_2$. If Z occurs in W_1 then we proceed as follows:

$$\frac{\dfrac{X \vdash W_1 \multimap W_2}{W_1, X \vdash W_2} \text{(MD1b)}}{W_1 \vdash X \multimap W_2} \text{(MD1a)}$$

Otherwise, Z occurs in W_2 and we proceed as follows:

$$\frac{X \vdash W_1 \multimap W_2}{X, W_1 \vdash W_2} \text{(MD1a)}$$

The justification that the required properties hold is then similar to the cases above. This completes the case and thus the proof. □

We can now prove the fundamental display theorem for $\mathrm{DL_{BBI}}$.

Theorem 3.7 (Display theorem)**.** *Let Z be a structure occurrence in a consecution $X \vdash Y$. If Z is an antecedent part of $X \vdash Y$ then there exists some structure W such that $Z \vdash W \equiv_D X \vdash Y$. Similarly, if Z is a consequent part of $X \vdash Y$ then there exists some structure W such that $W \vdash Z \equiv_D X \vdash Y$.*

Proof. By induction on the height h of Z in $X \vdash Y$. In the case $h = 0$, we are trivially done. In the case where $h > 0$, we can apply Lemma 3.6 and the induction hypothesis to obtain the required consecution. □

The process of of rearranging a consecution $X \vdash Y$ into the consecution $Z \vdash W$ or $W \vdash Z$ using the display rules is called *displaying* Z (or W), and Z (or W) is said to be *displayed* in the resulting consecution.

The proof rules for $\mathrm{DL_{BBI}}$ are given in Figure 1, and fall into three distinct categories. The identity rules consist of the identity axiom for propositional variables, a rule for display-equivalence, and a cut rule. The logical rules consist of a left and right introduction rule for each formula connective, in the style familiar from sequent calculus. Note that the formula introduced by a logical rule is displayed on the side of the consecution in which it is introduced; the display theorem allows us to write the logical rules in this form without loss of generality. The structural rules implement suitable associativity and unitary laws for the structural connectives, plus weakening and contraction with respect to the semicolon.

Identity rules:

$$\dfrac{}{P \vdash P}\,(\mathrm{Id}) \qquad \dfrac{X \vdash F \quad F \vdash Y}{X \vdash Y}\,(\mathrm{Cut}) \qquad \dfrac{X' \vdash Y'}{X \vdash Y}\ X \vdash Y \equiv_D X' \vdash Y' \ \ (\equiv_D)$$

Logical rules:

$$\dfrac{\emptyset \vdash X}{\top \vdash X}\,(\top\mathrm{L}) \qquad \dfrac{}{\bot \vdash \emptyset}\,(\bot\mathrm{L}) \qquad \dfrac{\sharp F \vdash X}{\neg F \vdash X}\,(\neg\mathrm{L})$$

$$\dfrac{}{\emptyset \vdash \top}\,(\top\mathrm{R}) \qquad \dfrac{X \vdash \emptyset}{X \vdash \bot}\,(\bot\mathrm{R}) \qquad \dfrac{X \vdash \sharp F}{X \vdash \neg F}\,(\neg\mathrm{R})$$

$$\dfrac{F; G \vdash X}{F \wedge G \vdash X}\,(\wedge\mathrm{L}) \qquad \dfrac{F \vdash X \quad G \vdash Y}{F \vee G \vdash X; Y}\,(\vee\mathrm{L}) \qquad \dfrac{X \vdash F \quad G \vdash Y}{F \to G \vdash \sharp X; Y}\,(\to\mathrm{L})$$

$$\dfrac{X \vdash F \quad Y \vdash G}{X; Y \vdash F \wedge G}\,(\wedge\mathrm{R}) \qquad \dfrac{X \vdash F; G}{X \vdash F \vee G}\,(\vee\mathrm{R}) \qquad \dfrac{X; F \vdash G}{X \vdash F \to G}\,(\to\mathrm{R})$$

$$\dfrac{\varnothing \vdash X}{\top^* \vdash X}\,(\top^*\mathrm{L}) \qquad \dfrac{F, G \vdash X}{F * G \vdash X}\,(*\mathrm{L}) \qquad \dfrac{X \vdash F \quad G \vdash Y}{F \mathrel{-\!\!*} G \vdash X \multimap Y}\,(\mathrel{-\!\!*}\mathrm{L})$$

$$\dfrac{}{\varnothing \vdash \top^*}\,(\top^*\mathrm{R}) \qquad \dfrac{X \vdash F \quad Y \vdash G}{X, Y \vdash F * G}\,(*\mathrm{R}) \qquad \dfrac{X, F \vdash G}{X \vdash F \mathrel{-\!\!*} G}\,(\mathrel{-\!\!*}\mathrm{R})$$

Structural rules:

$$\begin{array}{c} W; (X; Y) \vdash Z \\ \hline\hline (W; X); Y \vdash Z \end{array}\,(\mathrm{AAL}) \qquad \begin{array}{c} W \vdash (X; Y); Z \\ \hline\hline W \vdash X; (Y; Z) \end{array}\,(\mathrm{AAR}) \qquad \begin{array}{c} W, (X, Y) \vdash Z \\ \hline\hline (W, X), Y \vdash Z \end{array}\,(\mathrm{MAL})$$

$$\begin{array}{c} \emptyset; X \vdash Y \\ \hline\hline X \vdash Y \end{array}\,(\emptyset\mathrm{L}) \qquad \begin{array}{c} X \vdash Y; \emptyset \\ \hline\hline X \vdash Y \end{array}\,(\emptyset\mathrm{R}) \qquad \dfrac{\varnothing, X \vdash Y}{X \vdash Y}\,(\varnothing 1) \qquad \dfrac{X \vdash Y}{\varnothing, X \vdash Y}\,(\varnothing 2)$$

$$\dfrac{X \vdash Z}{X; Y \vdash Z}\,(\mathrm{WkL}) \qquad \dfrac{X \vdash Z}{X \vdash Y; Z}\,(\mathrm{WkR}) \qquad \dfrac{X; X \vdash Y}{X \vdash Y}\,(\mathrm{CtrL}) \qquad \dfrac{X \vdash Y; Y}{X \vdash Y}\,(\mathrm{CtrR})$$

Figure 1: Proof rules for $\mathrm{DL_{BBI}}$. W, X, Y, Z range over structures, F, G range over formulas, and P ranges over $\mathcal{V}$.

Proposition 3.8 (Soundness)**.** *Any* $\mathrm{DL}_{\mathrm{BBI}}$*-provable consecution is valid.*

Proof. Soundness follows as usual from the fact that each proof rule of $\mathrm{DL}_{\mathrm{BBI}}$ is locally sound in that, if each of its premises is valid, then so is its conclusion. We show how to treat the rule ($-\!\!*$L):

$$\frac{X \vdash F \quad G \vdash Y}{F -\!\!* G \vdash X \multimap Y}\ (-\!\!*\mathrm{L})$$

Suppose that the conclusion of this rule is invalid. By definition of validity for consecutions and formulas (Defns. 3.2 and 2.4), this means that the formula $(F -\!\!* G) \rightarrow (\Psi_X -\!\!* \Upsilon_Y)$ is false in some BBI-model $M = \langle R, \circ, e\rangle$. So for some $r \in R$ (and environment ρ) we have $r \models_\rho F -\!\!* G$ but $r \not\models_\rho \Psi_X -\!\!* \Upsilon_Y$. The latter of these implies there exist $r', r'' \in R$ such that $r'' \in r \circ r'$ and $r' \models_\rho \Psi_X$ but $r'' \not\models_\rho \Upsilon_Y$. Suppose first that $r' \not\models_\rho F$, then since $r' \models_\rho \Psi_X$ we have that the formula $\Psi_X \rightarrow F$ is false in M and hence invalid, so the premise $X \vdash F$ is invalid. Otherwise, we have $r' \models_\rho F$ and, since we have $r'' \in r \circ r'$ and $r \models_\rho F -\!\!* G$, we must have $r'' \models_\rho G$. But then we have $r'' \models_\rho G$ and $r'' \not\models_\rho \Upsilon_Y$, which implies that the premise $G \vdash Y$ is invalid. So validity of both premises implies validity of the conclusion, as required. The other rule cases are similar, although note that local soundness of the rule ($\equiv_D$) follows from the local soundness of each of the display rules (cf. Defn. 3.3). □

The identity axiom (Id) can be extended from variables to arbitrary formulas.

Proposition 3.9 (Identity)**.** $F \vdash F$ *is* $\mathrm{DL}_{\mathrm{BBI}}$*-provable for any formula* F*.*

Proof. By structural induction on F. We show the case $F = F_1 -\!\!* F_2$.

$$\cfrac{\cfrac{\cfrac{\begin{matrix}(\text{I.H.}) \\ \vdots \\ F_1 \vdash F_1\end{matrix} \quad \begin{matrix}(\text{I.H.}) \\ \vdots \\ F_2 \vdash F_2\end{matrix}}{F_1 -\!\!* F_2 \vdash F_1 \multimap F_2}\ (-\!\!*\mathrm{L})}{F_1 -\!\!* F_2, F_1 \vdash F_2}\ (\equiv_D)}{F_1 -\!\!* F_2 \vdash F_1 -\!\!* F_2}\ (-\!\!*\mathrm{R})$$

The other cases are similar. □

Since we already have a complete Hilbert system for BBI (cf. Section 2), completeness can be proven in the standard manner for display calculi (cf. [10]).

Lemma 3.10. *For any consecution* $X \vdash Y$*, the consecutions* $X \vdash \Psi_X$ *and* $\Upsilon_Y \vdash Y$ *are* $\mathrm{DL}_{\mathrm{BBI}}$*-provable.*

Proof. By mutual induction on the structure of X and Y. We show how to treat the case $Y = X' \multimap Y'$, using the rule symbol (=) to denote rewriting a consecution according to the definitions of Ψ_{-} and Υ_{-}.

$$\dfrac{\dfrac{\begin{matrix}\text{(I.H.)}\\ \vdots\\ X' \vdash \Psi_{X'}\end{matrix} \quad \begin{matrix}\text{(I.H.)}\\ \vdots\\ \Upsilon_{Y'} \vdash Y'\end{matrix}}{\Psi_{X'} \mathrel{-\!*} \Upsilon_{Y'} \vdash X' \multimap Y'}\,(\mathrel{-\!*}\text{L})}{\Upsilon_{X' \multimap Y'} \vdash X' \multimap Y'}\,(=)$$

The other induction step cases are similar. When X or Y is a formula, we invoke Proposition 3.9. □

Theorem 3.11 (Completeness)**.** *Any valid consecution is* $\mathrm{DL}_{\mathrm{BBI}}$*-provable.*

Proof. If a consecution $X \vdash Y$ is valid then $\Psi_X \rightarrow \Upsilon_Y$ is a valid formula, and hence $\mathrm{HL}_{\mathrm{BBI}}$-provable by completeness (Theorem 2.5). It can be easily verified that the axioms and inference rules of $\mathrm{HL}_{\mathrm{BBI}}$ and its *modus ponens* rule are all derivable in $\mathrm{DL}_{\mathrm{BBI}}$ under the embedding $F \mapsto (\emptyset \vdash F)$ from formulas to consecutions. For example, in the case of the inference rule $(\mathrel{-\!*}1)$ we proceed as follows:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\emptyset \vdash F \rightarrow (G \mathrel{-\!*} H) \qquad \dfrac{\begin{matrix}\text{(Prop. 3.9)}\\ \vdots\\ F \vdash F\end{matrix} \quad \dfrac{\begin{matrix}\text{(Prop. 3.9)}\\ \vdots\\ G \vdash G\end{matrix} \quad \begin{matrix}\text{(Prop. 3.9)}\\ \vdots\\ H \vdash H\end{matrix}}{G \mathrel{-\!*} H \vdash G \multimap H}\,(\mathrel{-\!*}\text{L})}{F \rightarrow (G \mathrel{-\!*} H) \vdash \sharp F ; G \multimap H}\,(\rightarrow\text{L})}{\emptyset \vdash \sharp F ; G \multimap H}\,(\text{Cut})}{\emptyset ; F \vdash G \multimap H}\,(\equiv_D)}{F \vdash G \multimap H}\,(\emptyset\text{L})}{F, G \vdash H}\,(\equiv_D)}{F * G \vdash H}\,(*\text{L})}{\dfrac{\emptyset ; F * G \vdash H}{\emptyset \vdash F * G \rightarrow H}\,(\rightarrow\text{R})}\,(\emptyset\text{L})$$

Thus $\emptyset \vdash \Psi_X \rightarrow \Upsilon_Y$ is $\mathrm{DL}_{\mathrm{BBI}}$-provable, and we can construct a $\mathrm{DL}_{\mathrm{BBI}}$ proof of $X \vdash Y$ as follows:

$$
\begin{prooftree}
\AxiomC{Lemma 3.10}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$X \vdash \Psi_X$}
\AxiomC{$\vdots$}
\noLine
\UnaryInfC{$\emptyset \vdash \Psi_X \to \Upsilon_Y$}
\AxiomC{Prop. 3.9}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$\Psi_X \vdash \Psi_X$}
\AxiomC{Prop. 3.9}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$\Upsilon_Y \vdash \Upsilon_Y$}
\RightLabel{$(\to\mathrm{L})$}
\BinaryInfC{$\Psi_X \to \Upsilon_Y \vdash \sharp\Psi_X ; \Upsilon_Y$}
\RightLabel{(Cut)}
\BinaryInfC{$\emptyset \vdash \sharp\Psi_X ; \Upsilon_Y$}
\RightLabel{$(\equiv_D)$}
\UnaryInfC{$\emptyset ; \Psi_X \vdash \Upsilon_Y$}
\RightLabel{$(\emptyset\mathrm{L})$}
\UnaryInfC{$\Psi_X \vdash \Upsilon_Y$}
\AxiomC{Lemma 3.10}
\noLine
\UnaryInfC{$\vdots$}
\noLine
\UnaryInfC{$\Upsilon_Y \vdash Y$}
\RightLabel{(Cut)}
\BinaryInfC{$\Psi_X \vdash Y$}
\RightLabel{(Cut)}
\BinaryInfC{$X \vdash Y$}
\end{prooftree}
$$

□

We say a $\mathrm{DL}_{\mathrm{BBI}}$ proof is *cut-free* if it contains no instances of the rule (Cut). The following definition is taken from Belnap [1]. By a *constituent* of a structure or consecution we mean an occurrence of one of its substructures.

Definition 3.12 (Parameters / congruence)**.** Let I be an instance of a $\mathrm{DL}_{\mathrm{BBI}}$ proof rule R. Note that I is obtained by assigning structures to the structure variables occurring in R and formulas to the formula variables occurring in R.

Any constituent of the consecutions in I occurring as part of structures assigned to structure variables in I are defined to be *parameters* of I. All other constituents are defined to be *non-parametric* in I, including those assigned to formula variables.

Constituents occupying similar positions in occurrences of structures assigned to the same structure variable are defined to be *congruent* in I.

We remark that congruence as defined above is an equivalence relation. We can now prove cut-elimination for $\mathrm{DL}_{\mathrm{BBI}}$ using the standard Belnap-style argument.

Theorem 3.13 (Cut-elimination)**.** *Any* $\mathrm{DL}_{\mathrm{BBI}}$ *proof can be transformed into a cut-free* $\mathrm{DL}_{\mathrm{BBI}}$ *proof (of the same consecution).*

Proof. Belnap's original analysis of display logic [1] guarantees cut-elimination (Theorem 3.13) provided the proof rules of $\mathrm{DL}_{\mathrm{BBI}}$ satisfy the following 8 conditions, which are stated with reference to an instance I of a proof rule R. (Here, we state a stronger, combined version of Belnap's original conditions C6 and C7, following Kracht [13], since the rules satisfy this stronger condition.) In each case, we indicate how to verify that the condition holds for our rules.

C1. *Preservation of formulas.* Each formula which is a constituent of some premise of I is a subformula of some formula in the conclusion of I.
Verification. One observes that, in each rule, no formula variable or structure variable is lost when passing from the premises to the conclusions.

C2. *Shape-alikeness of parameters.* Congruent parameters are occurrences of the same structure.
Verification. Immediate from the definition of congruence.

C3. *Non-proliferation of parameters.* No two constituents in the conclusion of I are congruent to each other.
Verification. One just observes that, for each rule, each structure variable occurs exactly once in the conclusion.

C4. *Position-alikeness of parameters.* Congruent parameters are either all antecedent or all consequent parts of their respective consecutions.
Verification. One observes that, in each rule, no structure variable occurs both as an antecedent part and a consequent part.

C5. *Display of principal constituents.* If a formula is nonparametric in the conclusion of I, it is either the entire antecedent or the entire consequent of that conclusion. Such a formula is said to be *principal* in I.
Verification. It is easy to verify that the only non-parametric formulas in the conclusions of our rules are the two occurrences of P in (Id) and those occurring in the introduction rules for the logical connectives in Figure 1, which obviously satisfy the condition.

C6/7. *Closure under substitution for parameters.* Each rule is closed under simultaneous substitution of arbitrary structures for congruent formulas which are parameters.
Verification. This condition is satisfied because no restrictions are placed on the structural variables used in our rules.

C8. *Eliminability of matching principal formulas.* If there are inferences I_1 and I_2 with respective conclusions $X \vdash F$ and $F \vdash Y$ and with F principal in both inferences, then either $X \vdash Y$ is equal to one of $X \vdash F$ and $F \vdash Y$, or there is a derivation of $X \vdash Y$ from the premises of I_1 and I_2 in which every instance of cut has a cut-formula which is a proper subformula of F.
Verification. There are only two cases to consider. If F is atomic then $X \vdash F$ and $F \vdash Y$ are both instances of (Id). Thus we must have $X \vdash F = F \vdash Y = X \vdash Y$, and are done. Otherwise F is non-atomic and introduced in I_1 and I_2 respectively by the right and left introduction rule for the main connective of F. In this case, a derivation of the desired form can be obtained using only the display rule ($\equiv_D$) and cuts on subformulas of F. For example, if the considered cut is of the form:

$$\dfrac{\dfrac{\begin{array}{c}\vdots\\ X, F \vdash G\end{array}}{X \vdash F -\!\!* G}\,(-\!\!*\mathrm{R}) \qquad \dfrac{\begin{array}{c}\vdots\\ Y \vdash F\end{array} \quad \begin{array}{c}\vdots\\ G \vdash Z\end{array}}{F -\!\!* G \vdash Y \multimap Z}\,(-\!\!*\mathrm{L})}{X \vdash Y \multimap Z}\,(\mathrm{Cut})$$

then the cut is reduced as follows:

$$\dfrac{\dfrac{\dfrac{\dfrac{\begin{matrix}\vdots\\ Y\vdash F\end{matrix} \quad \dfrac{\begin{matrix}\vdots\\ X,F\vdash G\end{matrix}}{F\vdash X\multimap G}\,(\equiv_D)}{Y\vdash X\multimap G}\,(\text{Cut})}{X,Y\vdash G}\,(\equiv_D) \qquad \begin{matrix}\vdots\\ G\vdash Z\end{matrix}}{X,Y\vdash Z}\,(\text{Cut})}{X\vdash Y\multimap Z}\,(\equiv_D)$$

Similarly, a principal cut on the formula $F * G$ has the form:

$$\dfrac{\dfrac{\begin{matrix}\vdots\\ X\vdash F\end{matrix} \quad \begin{matrix}\vdots\\ Y\vdash G\end{matrix}}{X,Y\vdash F*G}\,(*\text{R}) \qquad \dfrac{\begin{matrix}\vdots\\ F,G\vdash Z\end{matrix}}{F*G\vdash Z}\,(*\text{L})}{X,Y\vdash Z}\,(\text{Cut})$$

This cut is reduced as follows:

$$\dfrac{\dfrac{\begin{matrix}\vdots\\ X\vdash F\end{matrix} \qquad \dfrac{\dfrac{\begin{matrix}\vdots\\ Y\vdash G\end{matrix} \quad \dfrac{\begin{matrix}\vdots\\ F,G\vdash Z\end{matrix}}{G\vdash F\multimap Z}\,(\equiv_D)}{Y\vdash F\multimap Z}\,(\text{Cut})}{F\vdash Y\multimap Z}\,(\equiv_D)}{X\vdash Y\multimap Z}\,(\text{Cut})}{X,Y\vdash Z}\,(\equiv_D)$$

This completes the verification of the conditions, and thus the proof.

□

Corollary 3.14 (Subformula property). *Any* DL_{BBI}*-provable consecution* $X \vdash Y$ *has a proof in which every formula occurrence is a subformula of a formula occurring in* $X \vdash Y$.

Proof. If $X \vdash Y$ is DL_{BBI}-provable then it has a cut-free proof by Theorem 3.13. By inspection of the rules, any formula occurring in the premises of a rule instance in this proof is a subformula of a formula occurring in its conclusion. Thus every formula occurring in this proof is a subformula of a formula in $X \vdash Y$. □

In spite of the subformula property, cut-free proof search in DL_{BBI} is complicated considerably by the presence of the display rule and the structural rules.

For example, the cut-free proof of the consecution $F \vdash (F * G) \vee (F * \neg G)$ shown in Figure 2 makes seemingly essential use of contraction on a structure which is introduced into the proof using unitary and display rules.

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\begin{array}{c}\text{Prop. 3.9}\\ \vdots\\ F \vdash F\end{array}\qquad \dfrac{\dfrac{\dfrac{\begin{array}{c}\text{Prop. 3.9}\\ \vdots\\ F \vdash F\end{array}\qquad \dfrac{\dfrac{\begin{array}{c}\text{Prop. 3.9}\\ \vdots\\ G \vdash G\end{array}}{\sharp G \vdash \sharp G}\,(\equiv_D)}{\sharp G \vdash \neg G}\,(\neg\text{R})}{F, \sharp G \vdash F * \neg G}\,(*\text{R})}{F, \sharp G \vdash F * G; F * \neg G}\,(\text{WkR})}{\sharp(F \multimap (F * G; F * \neg G)) \vdash G}\,(\equiv_D)}{F, \sharp(F \multimap (F * G; F * \neg G)) \vdash F * G}\,(*\text{R})}{(F, \sharp(F \multimap (F * G; F * \neg G))); \sharp(F * \neg G) \vdash F * G}\,(\text{WkL})}{\sharp(F \multimap (F * G; F * \neg G)) \vdash F \multimap (F * G; F * \neg G)}\,(\equiv_D)}{\varnothing; \sharp(F \multimap (F * G; F * \neg G)) \vdash F \multimap (F * G; F * \neg G)}\,(\text{WkL})}{\varnothing \vdash (F \multimap (F * G; F * \neg G)); (F \multimap (F * G; F * \neg G))}\,(\equiv_D)}{\varnothing \vdash F \multimap (F * G; F * \neg G)}\,(\text{CtrR})}{F, \varnothing \vdash F * G; F * \neg G}\,(\equiv_D)}{F \vdash F * G; F * \neg G}\,(\varnothing 1)}{F \vdash (F * G) \vee (F * \neg G)}\,(\vee\text{R})
$$

Figure 2: A DL_{BBI} proof of the consecution $F \vdash (F * G) \vee (F * \neg G)$.

4 Constraining proof search for DL_{BBI}

In this section we show how to reduce the search space for DL_{BBI} proofs by means of a series of (completeness-preserving) proof reductions. Our reductions are obtained by adapting Restall's techniques for showing the decidability of display calculi for a class of relevant logics [18]. However, several substantial alterations are necessary in order to make the proofs go through in the setting of BBI. Furthermore, our reductions are seemingly not sufficient to establish the decidability of DL_{BBI}, due to the potential for quite sophisticated redundancies to be created during proof search by Boolean negation in combination with weakening and contraction.

First, given any structure X we define its *reduced inversion* $\overline{X}$ as follows:

$$\overline{X} =_{\text{def}} \begin{cases} Y & \text{if } X = \sharp Y \text{ for some } Y \\ \sharp X & \text{otherwise} \end{cases}$$

Proposition 4.1 (Rule replacement)**.** *Provability in* $\mathrm{DL_{BBI}}$ *remains unaffected if the rules (Cut), ($\emptyset$L) and ($\emptyset$R) are excised and the rules ($\top$R), ($\bot$L), ($\wedge$R), ($\vee$L) and ($\rightarrow$L) are replaced by the following variants:*

$$\dfrac{X \vdash F \quad X \vdash G}{X \vdash F \wedge G}\,(\wedge R) \qquad \dfrac{F \vdash Y \quad G \vdash Y}{F \vee G \vdash Y}\,(\vee L) \qquad \dfrac{X \vdash F \quad X \vdash \sharp G}{F \rightarrow G \vdash \overline{X}}\,(\rightarrow L) \qquad \begin{array}{c} \dfrac{}{X \vdash \top}\,(\top R) \\ \dfrac{}{\bot \vdash X}\,(\bot L) \end{array}$$

Proof. (Sketch) First, provability in $\mathrm{DL_{BBI}}$ is obviously unaffected by the removal of (Cut), given cut-elimination (Theorem 3.13). The new versions of the modified proof rules are clearly sound, so for completeness we must show that the old versions are cut-free derivable using the new versions. This is trivial in the case of ($\top$R) and ($\bot$L). In the case of ($\rightarrow$L), we proceed as follows, using the rule symbol ($=$) to denote rewriting a consecution by the definition of $\overline{X}$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{X \vdash F}{X;\sharp Y \vdash F}\,(\text{WkL})}{\sharp\sharp(X;\sharp Y) \vdash F}\,(\equiv_D) \qquad \dfrac{\dfrac{\dfrac{G \vdash Y}{\sharp Y \vdash \sharp G}\,(\equiv_D)}{\sharp Y; X \vdash \sharp G}\,(\text{WkL})}{\sharp\sharp(X;\sharp Y) \vdash \sharp G}\,(\equiv_D)}{F \rightarrow G \vdash \overline{\sharp\sharp(X;\sharp Y)}}\,(\rightarrow\text{L})}{F \rightarrow G \vdash \sharp(X;\sharp Y)}\,(=)}{F \rightarrow G \vdash \sharp X; Y}\,(\equiv_D)$$

The other rule cases are similar. Thus any proof using the old rules can be converted to one using the new rules.

Finally, we need to show how to eliminate ($\emptyset$L) and ($\emptyset$R). We show how to treat ($\emptyset$L); the case of ($\emptyset$R) is exactly dual. First, we replace an instance of ($\emptyset$L) by an instance of (CtrL) as follows:

$$\dfrac{\emptyset; X \vdash Y}{X \vdash Y}\,(\emptyset\text{L}) \quad \Rightarrow \quad \dfrac{X; X \vdash Y}{X \vdash Y}\,(\text{CtrL})$$

We then replace by X all occurrences of $\emptyset$ corresponding to the indicated occurrence in the consecutions above the considered ($\emptyset$L), working bottom-up. (By “corresponding” we mean that a replaced $\emptyset$ occurring in the premise of a proof rule must occupy a similar position to a replaced $\emptyset$ in the conclusion. The notion of correspondence can be defined fully formally, e.g., by enumerating occurrences of $\emptyset$.) The result remains a proof because, with the modified versions of ($\top$R) and ($\bot$L) in place, all proof rules are closed under the substitution of arbitrary structures for occurrences of $\emptyset$. This completes the proof. □

From now on, we work with respect to the modified proof rules of $\mathrm{DL_{BBI}}$ given by Proposition 4.1.

Definition 4.2 ($\sharp$-reduction)**.** A structure is said to be *$\sharp$-reduced* if it has no substructures of the form $\sharp\sharp X$. A consecution $X \vdash Y$ is $\sharp$-reduced just in case both X and Y are $\sharp$-reduced, and a proof is $\sharp$-reduced just in case every consecution in it is $\sharp$-reduced.

The *$\sharp$-reduction* of a consecution S is the consecution obtained from S by successively replacing every structure occurrence of the form $\sharp\sharp X$ by the structure X until the result is $\sharp$-reduced.

We remark that any consecution is clearly display-equivalent to its $\sharp$-reduction. Thus a consecution is provable just in case its $\sharp$-reduction is provable.

Lemma 4.3. *Any provable $\sharp$-reduced consecution has a $\sharp$-reduced proof.*

Proof. Let π be a proof of a $\sharp$-reduced consecution and let π' be the tree obtained by replacing every consecution in π by its $\sharp$-reduction. We claim that π' is still a proof.

It suffices to show that each of the proof rules remains unaffected by $\sharp$-reduction, i.e., that any instance of a proof rule remains an instance of the same rule under $\sharp$-reduction. This property can easily be seen to be satisfied for all the rules that do not introduce or eliminate occurrences of $\sharp$. The remaining rules are ($\equiv_D$), ($\neg$L), ($\neg$R) and ($\rightarrow$L). The display rule ($\equiv_D$) is obviously all right, since any consecution is display equivalent to its $\sharp$-reduction and display-equivalence is transitive. So too are the negation-introduction rules ($\neg$L) and ($\neg$R), because $\sharp F$ is already $\sharp$-reduced (since F is a formula). In the case of ($\rightarrow$L), the modified left-introduction rule for implication, one observes that if the premises of ($\rightarrow$L) are $\sharp$-reduced then clearly so is the conclusion, because of the use of the reduced inversion $\overline{X}$ in place of $\sharp X$. So π' is indeed a proof, as required. □

Lemma 4.3 implies that, in order to search for a proof of a consecution S, it suffices to consider just those proofs involving only $\sharp$-reduced consecutions. This is something of a relief, since any consecution has infinitely many display-equivalent representations that are not $\sharp$-reduced. The original technique of eliminating stacked $\sharp$s from consecutions is due to Kracht [13].

The following series of definitions is intended to assist in dealing with the non-determinism in proof search caused by the structural rules by defining a notion of reduction on consecutions that eliminates further superfluous information. They follow the corresponding definitions in [18] but with some crucial differences, mainly due to the fact that in $\mathrm{DL}_{\mathrm{BBI}}$ we are allowed to contract consequent structures as well as antecedent ones.

Definition 4.4 (AD-equivalence)**.** Two consecutions S and S' are said to be *AD-equivalent*, written $S \equiv_{AD} S'$, if they are interderivable using the display rule ($\equiv_D$) and the structural associativity rules (AAL), (AAR) and (MAL).

Definition 4.5 (Nearness)**.** Two structure occurrences Z_1, Z_2 in a consecution S are said to be *near* (in S) if $S \equiv_{AD} Z_1; Z_2 \vdash W$ or $S \equiv_{AD} W \vdash Z_1; Z_2$ for some W (where Z_1 and Z_2 are the occurrences indicated).

Definition 4.6 (Superfluous $\varnothing$)**.** An occurrence of $\varnothing$ in a consecution S is said to be *superfluous* if $S \equiv_{AD} \varnothing, W \vdash Z$ for some W, Z (where $\varnothing$ is the occurrence indicated).

Definition 4.7 (Deletion)**.** Let Z be a structure occurrence in a consecution $X \vdash Y$. We say that Z can be *deleted* from $X \vdash Y$ if Z is a strict substructure of some structure W in $X \vdash Y$ where W is of the form $W_1; W_2$ or W_1, W_2 or $W_1 \multimap W_2$. (I.e., Z is not any of $X, Y, \overline{X}, \overline{Y}$.) The consecution obtained by deleting Z from $X \vdash Y$ is obtained by replacing any of $W \star Z$, $Z \star W$, $W \star \sharp Z$, $\sharp Z \star W$, $Z \multimap W$, $\sharp Z \multimap W$ by W, where $\star$ is comma or semicolon.

We remark that if X and Y are near in a consecution S then at least one of them can be deleted from S.

Definition 4.8 (Reduction)**.** For any $\sharp$-reduced consecution S, we define its *reduction* $r(S)$ to be the consecution obtained from S by iterating the following sequence of steps until a fixed point is reached:

1. For any two occurrences of the same structure that are near to one another, we delete the first such occurrence, from left to right, that can be deleted (there must be at least one such).
2. We delete any superfluous occurrences of $\varnothing$ that can be deleted. If a superfluous occurrence of $\varnothing$ cannot be deleted then the consecution must be of the form $\varnothing \vdash X \multimap Y$ or $\sharp(X \multimap Y) \vdash \sharp\varnothing$, and in both cases we reduce the consecution to $X \vdash Y$.

A consecution S is said to be *reduced* if $S = r(S)$.

Clearly a consecution is valid (hence provable) iff its reduction is valid (hence provable). Unfortunately, it will not be possible to exclusively consider reduced consecutions, which would amount to the total elimination of contraction and the rules for $\varnothing$. We therefore introduce a notion of "semi-reduction". Informally speaking, a consecution is semi-reduced if it is either reduced or one "reduction iteration step" away from being so.

Definition 4.9 (Semi-reduction)**.** A consecution S is said to be *semi-reduced* if it is reduced or contains a single superfluous occurrence of $\varnothing$ and/or a single structure occurrence near to another occurrence of the same structure such that, when these occurrences are eliminated from S by the reduction procedure given in Definition 4.8, the resulting consecution is reduced.

A proof is semi-reduced if every consecution appearing in it is semi-reduced.

We observe that the proof of the reduced consecution $F \vdash (F * G) \vee (F * \neg G)$ in Figure 2 is semi-reduced; the contraction introduces one structure near to an instance of the same structure, and exactly one superfluous $\varnothing$ is used (note that it is however *not* superfluous in the premise of the contraction instance).

Definition 4.10 (Irredundancy)**.** A proof is said to be *irredundant* just in case no consecution occurs twice on any branch in the proof.

Lemma 4.11. *Any provable reduced consecution has a proof that is both semi-reduced and irredundant.*

Proof. First, any semi-reduced proof that is not irredundant can then be converted to a proof that is both semi-reduced and irredundant by deleting the sections of proof between every pair of identical consecutions occurring on the same branch in the proof. Thus it suffices to show that any provable reduced consecution has a semi-reduced proof.

Let π be a proof of a reduced consecution S. By Lemma 4.3 we can assume without loss of generality that π is $\sharp$-reduced. Then define π' to be the tree obtained by replacing every consecution in π by its reduction. π' is almost a proof of S; its root is S because S is already reduced by assumption, and the leaves of π' are axioms because every axiom in π (i.e. instances of the conclusions of the rules (Id), ($\bot$L), ($\top$R) and ($\top^*$R)) remains an instance of the same rule under reduction. For each of the other proof rules, we show how to derive the reduction of its conclusion from the reductions of its premises, using only semi-reduced consecutions. Thus any proof of a reduced consecution can be converted to a semi-reduced proof by replacing each consecution by its reduction and inserting these derivations between nodes as required.

Cases ($\varnothing$1),($\varnothing$2),(CtrL),(CtrR). All of these rules collapse into the identity rule (the trivial case of ($\equiv_D$) in which the premise and conclusion are identical) under reduction. For example, in the case of ($\emptyset$R) it is clear that $r(X \vdash Y; \emptyset) = r(X \vdash Y)$ since the indicated occurrence of $\emptyset$ is superfluous in $X \vdash Y; \emptyset$. Similarly, in the case of (CtrL) we have $r(X; X \vdash Y) = r(X \vdash Y)$ since the two indicated occurrences of X are near in $X; X \vdash Y$.

Case ($\equiv_D$). We require to show that if S and S' are display-equivalent then so are $r(S)$ and $r(S')$. To see this, observe that S and S' are AD-equivalent, so that any two structure occurrences near in S are near in S' and vice versa, and any superfluous unit occurrences in S are superfluous in S' and vice versa. Thus $r(-)$ deletes the same structure occurrences in S and S'. It is clear that this operation preserves display-equivalence.

Cases (AAL), (AAR), (MAL). The premise and conclusion of each of these rules are AD-equivalent. Thus the same structure occurrences are deleted in the premise and conclusion by $r(-)$. It is clear that the rule instance is either preserved by reduction or collapses into the identity rule. E.g., in the case of (AAL), if the premise is $\sharp Z; (X; \emptyset) \vdash Z$ then the reduction of both premise and conclusion is $r(X \vdash Z)$.

Cases (WkL), (WkR). Consider (WkL), and write the reduced premise $r(X \vdash Z)$ as $X' \vdash Z'$. It is clear that the reduced conclusion $r(X; Y \vdash Z)$ is $X'; Y' \vdash Z'$ for some Y' unless the structure Y weakened in by the rule causes a structure to be deleted in X' or Z' that would otherwise be untouched (e.g.

if $X' = A;B$, $Z' = C$ and $Y = A$). We define W to be the structure obtained by deleting from Y any structure that is near in $X';Y \vdash Z'$ to some structure occurrence in X or Z, and any occurrence of a unit that is superfluous in $X';Y \vdash Z'$. This operation differs from the notion of deletion in Definition 4.7 in that Y may be entirely deleted. In this case, the rule collapses into the identity rule. Otherwise, we apply (WkL) to $X' \vdash Z'$ to obtain $X';W \vdash Z'$. It is clear that $r(X;Y \vdash Z) = X';W \vdash Z'$ as required. The case for (WkR) is similar.

Cases ($\top$L), ($\bot$R), ($\top^*$L). Consider ($\top^*$L), and suppose first that the reduced premise $r(\varnothing \vdash X)$ is of the form $\varnothing \vdash X'$. Then by applying ($\top^*$L) we obtain $\top^* \vdash X'$, which is possibly only semi-reduced (because the introduced instance of $\top^*$ may be near to another instance of $\top^*$ in X'). We note that the reduced conclusion $r(\top^* \vdash X)$ is of the form $\top^* \vdash X''$ because the indicated instance of $\top^*$ cannot be deleted from $\top^* \vdash X$. One observes that X' and X'' can differ in two respects. First, X' may contain a $\top^*$ that is near to the indicated $\top^*$ in $\top^* \vdash X'$, and thus is not in X''. If so, we apply display and contraction rules to remove it. Second, X'' may contain an instance of $\varnothing$ that is near to the indicated $\varnothing$ in $\varnothing \vdash X''$ and thus is not in X'. If so, we apply display and weakening rules to reintroduce it (note that this maintains semi-reduction). The result is then $\top^* \vdash X'' = r(\top^* \vdash X)$ as required. For example, if $X = \sharp\varnothing;\sharp\top^*$ then the reduced premise is $\varnothing \vdash \sharp\top^*$ and the reduced conclusion is $\top^* \vdash \sharp\varnothing$. We "patch" the rule instance in the manner described above via the following derivation:

$$\dfrac{\dfrac{\dfrac{\dfrac{\varnothing \vdash \sharp\top^*}{\top^* \vdash \sharp\top^*}\,(\top^*\mathrm{L})}{\top^* \vdash \sharp\varnothing;\sharp\top^*}\,(\mathrm{WkR})}{\top^*;\top^* \vdash \sharp\varnothing}\,(\equiv_D)}{\top^* \vdash \sharp\varnothing}\,(\mathrm{CtrL})$$

If $r(\varnothing \vdash X)$ is *not* of the form $\varnothing \vdash X'$ then the indicated instance of $\varnothing$ was deleted (because it was superfluous), and we must reintroduce it. Writing $r(\varnothing \vdash X) = Y \vdash Z$, we obtain the semi-reduced $r(\varnothing \vdash Y \multimap Z)$ by applying ($\varnothing$) and display-equivalence. This is of the form $\varnothing \vdash X'$, and we can then proceed as in the previous case.

The rules ($\top$L) and ($\bot$R) are treated similarly.

Cases ($\neg$L), ($\neg$R). Consider ($\neg$L), and observe that we have $r(\sharp F \vdash X)$ of the form $\sharp F \vdash X'$ and $r(\neg F \vdash X)$ of the form $\neg F \vdash X''$. We apply ($\neg$L) to the reduced premise to obtain $\neg F \vdash X'$, which is at least semi-reduced. As is similar to the case above, X' and X'' can differ in two ways. First X' may contain a $\neg F$ near to the indicated $\neg F$ in $\neg F \vdash X'$ that must be removed using contraction and display rules. Second, X'' may contain a $\sharp F$ that is not in X' because it was near to the indicated $\sharp F$ in $\sharp F \vdash X$, and which must be reintroduced using weakening and display rules. This process only involves semi-reduced consecutions, and the result is $r(\neg F \vdash X)$ as required.

The rule ($\neg$R) is treated similarly.

Cases ($\rightarrow$R), ($-\!\!*$R). Consider ($\rightarrow$R) and note that the reduced conclusion $r(X \vdash F \rightarrow G)$ is of the form $X'' \vdash F \rightarrow G$. Suppose the reduced premise $r(X; F \vdash G)$ is of the form $X'; F \vdash G$. Then we apply ($\rightarrow$R) to obtain the semi-reduced $X' \vdash F \rightarrow G$. As in previous cases, X' may have an extra $F \rightarrow G$ which must be removed by contraction, and may also be missing an F or G which must be reintroduced by weakening. The process uses only semi-reduced consecutions and the result is $X'' \vdash F \rightarrow G$ as required.

If $r(X; F \vdash G)$ is not of the form $X'; F \vdash G$ then it must instead be $F \vdash G$ as none of the unit reductions of Definition 4.8 is applicable, and the indicated F cannot be deleted because deletion occurs from left to right, and the indicated F is not near the indicated G. In this case we apply the unit rule ($\emptyset$L) to obtain the semi-reduced $\emptyset; F \vdash G$. We can then proceed as in the previous case.

The rule ($-\!\!*$R) is treated similarly.

Case ($\rightarrow$L), ($\wedge$R), ($\vee$L). Consider ($\rightarrow$L), and note that we have $r(X \vdash F)$ of the form $X' \vdash F$ and $r(X \vdash \sharp G)$ of the form $X'' \vdash \sharp G$. Now X' and X'' can only differ in that an F has been deleted in X' or in that a $\sharp G$ has been deleted in X''. If so, we reintroduce the missing instances using display and weakening rules to obtain $X''' \vdash F$ and $X''' \vdash \sharp G$, which are both semi-reduced. Then we apply ($\rightarrow$L) to obtain $F \rightarrow G \vdash \overline{X'''}$. This may not be $r(F \rightarrow G \vdash \overline{X}) = F \rightarrow G \vdash Y$ in that $\overline{X'''}$ contains an $F \rightarrow G$ near to the indicated instance of $F \rightarrow G$ in the reduced conclusion, in which case we must remove it using contraction. There can be no other differences since any units superfluous in the premises are superfluous in the conclusion, and any structures near in the premises are near in the conclusion, except for those considered.

The rules ($\wedge$R) and ($\vee$L) are treated similarly.

Cases ($-\!\!*$L), ($*$R). Consider ($-\!\!*$L), and note that we have $r(X \vdash F)$ of the form $X' \vdash F$ and $r(G \vdash Y)$ of the form $G \vdash Y'$. We can immediately apply ($-\!\!*$L) to obtain $F -\!\!* G \vdash X' \multimap Y'$. Note that the reduced conclusion is of the form $F -\!\!* G \vdash X''$. Now X'' and $X' \multimap Y'$ can differ in several respects. First, $X' \multimap Y'$ may be missing an instance of F or an instance of G that were near respectively to the indicated F and G in $X \vdash F$ and $G \vdash Y$, in which case said instances must be restored using display and weakening rules. Second, X' may be the unit $\varnothing$, in which case it is superfluous in $F -\!\!* G \vdash X' \multimap Y'$ and must be deleted using display and unit rules. Third, Y' may contain an instance of either $F -\!\!* G$ (if X is a superfluous $\varnothing$) or $X', F -\!\!* G$ (if X' is not a superfluous $\varnothing$) near to the indicated instances in $X', F -\!\!* G \vdash Y'$. In this case we use display and contraction rules to eliminate the duplicate structure instance. Note that the second and third conditions can hold simultaneously, but this is still allowed by semi-reduction.

The rule $(*\mathrm{R})$ is treated similarly.

Cases $(\wedge\mathbf{L})$, $(\vee\mathbf{R})$, $(*\mathbf{L})$. Consider $(\wedge\mathrm{L})$, and suppose that the reduced premise $r(F;G \vdash X)$ is of the form $F;G \vdash X'$. Then we can apply $(\wedge\mathrm{L})$ to obtain $F \wedge G \vdash X'$. The reduced conclusion is of the form $F \wedge G \vdash X''$ and, as in previous cases, X' and X'' may differ in two respects. First, X' may contain an instance of $F \wedge G$ near to the indicated instance in $F \wedge G \vdash X'$, and if so then we must remove it using contraction and display rules. Second, X' may be missing an instance of F or of G that was removed from X because it matched the indicated instance in $F;G \vdash X$, and if so then the instance must be reintroduced using weakening and display rules.

If $r(F;G \vdash X)$ is not of the form $F;G \vdash X'$ then there are two further possibilities to consider. First, $r(F;G \vdash X)$ may be of the form $F \vdash X'$ or $G \vdash X'$ because either F or G in the premise was deleted by reduction (note that both cannot be deleted). In that case, we can apply weakening to obtain the semi-reduced $F;G \vdash X'$, and then proceed as in the case above. Second, $r(F;G \vdash X)$ may be of the form $F \vdash \overline{G}$ because X is the superfluous unit $\emptyset$. In this case, we apply the rule $(\emptyset\mathrm{R})$ and the display rule to obtain the semi-reduced $F;G \vdash \emptyset$, and then proceed as in the case above.

The cases $(\vee\mathrm{R})$ and $(*\mathrm{L})$ are similar. This completes all the cases. □

Since a consecution is provable iff its reduction is provable, it clearly suffices for a proof search for an arbitrary consecution S to consider only semi-reduced, irredundant proofs of $r(S)$.

Somewhat surprisingly, the restriction to semi-reduced and irredundant proofs still does not yield a finite proof search space for $\mathrm{DL}_{\mathrm{BBI}}$, as can be seen by considering the following derivation:

$$\cfrac{\cfrac{\cfrac{\cfrac{\sharp(\sharp(X \multimap Y) \multimap Y) \vdash \sharp(X \multimap Y) \multimap Y}{X;\sharp(\sharp(X \multimap Y) \multimap Y) \vdash \sharp(X \multimap Y) \multimap Y}\,(\text{WkL})}{X \vdash (\sharp(X \multimap Y) \multimap Y);(\sharp(X \multimap Y) \multimap Y)}\,(\equiv_D)}{X \vdash \sharp(X \multimap Y) \multimap Y}\,(\text{CtrR})}{\sharp(X \multimap Y) \vdash X \multimap Y}\,(\equiv_D)$$

Note that the top and bottom consecutions are both reduced, and the derivation uses only semi-reduced consecutions. Thus we can obtain an infinite family of interderivable reduced consecutions of the form $\sharp Z \vdash Z$ in a semi-reduced proof [2]. Moreover, such derivation segments seemingly cannot be wholly elim-

[2] We could define the reduction of $\sharp Z \vdash Z$ to be $r(\emptyset \vdash Z)$, but this would force us to reintroduce the rules for $\emptyset$, and in any case the top and bottom consecutions in the derivation above would still be different under reduction. We also remark that the restriction to semi-reduced proofs is still important; without it, proof search in $\mathrm{DL}_{\mathrm{BBI}}$ would fail to terminate for entirely trivial reasons!

inated, since they generate multiplicative structure which may be needed to apply ($*$R) or ($-\!\!*$L).

Interestingly, BI, which can be obtained by dropping the classical axiom of excluded middle from the axiomatisation of BBI, is known to be decidable [8]. However, decidability of BBI does not follow from decidability of BI; indeed, Larchey-Wendling and Galmiche have recently demonstrated that there is an encoding of BI into BBI whereas the converse is not known to hold [14]. Thus it seems plausible that the addition of Boolean negation to BI results in a considerable increase in complexity.

5 Conclusions and future work

In this paper, we resolve the long-standing difficulty of the absence of a well-behaved proof theory for BBI, by formulating the $\mathrm{DL}_{\mathrm{BBI}}$ display calculus, using a nonsymmetric form of proof judgement (an approach also taken by Goré [11]). We also show to constrain proof search in $\mathrm{DL}_{\mathrm{BBI}}$ by means of proof reductions which impose bounds on the local applicability of the display and structural rules. These are adapted from Restall's reductions for relevant display calculi, which is not entirely a straightforward process because of the significant differences between these calculi and $\mathrm{DL}_{\mathrm{BBI}}$. Unlike in Restall's setting, our reductions do not entail decidability of $\mathrm{DL}_{\mathrm{BBI}}$, but they do serve to give some insight into the question of decidability, and they represent proof search optimisations that should assist in any putative implementation.

Of course, the main question left open by our developments is whether or not full BBI (equivalently $\mathrm{DL}_{\mathrm{BBI}}$) is indeed decidable, and it is not obvious to us which possibility is the more plausible, since the particular mix of structural rules is what typically makes the difference between a decidable display calculus and an undecidable one [19]. A salutary comparison is provided by the linear and relevance families of substructural logics, which provide seemingly similar expressivity to that of the BI family. The decision problem for these families falls into a spectrum of complexities for the decidable logics, while the most expressive members are known undecidable [15, 21] and the decidability of some important variants (such as multiplicative exponential linear logic) is still open. If BBI is in fact undecidable, these logics may provide hints as to an appropriate reduction of the decision problem for BBI. As things stand, our example in Section 4 shows that there are infinitely many interderivable representations of a consecution which are essentially multiplicative rather than additive, a feature that strikes us as ominously similar to the role of exponentials in linear logic. Alternatively, since provability in $\mathrm{DL}_{\mathrm{BBI}}$ is semi-decidable, another route to decidability would be via an enumeration of BBI models, if BBI were known to have the finite model property. Unfortunately, this too is currently open.

A very promising avenue for exploitation is the potential for theorem proving tools based upon $\mathrm{DL}_{\mathrm{BBI}}$, and its refinements suggested by our proof reductions. Such tools could provide a platform for checking pure entailments in separation and spatial logics (an essential step in Hoare-style verification and in shape analysis); the existing tools typically cannot deal with the full expressivity provided by BBI. Also, we think that our techniques should apply to a variety of other logical settings based on bunch-like structures.

Acknowledgements. We extend special thanks to Greg Restall for his helpful and enthusiastic technical advice, and to Lucas Dixon for encouraging the initial investigation. Thanks also to Cristiano Calcagno, Philippa Gardner, Peter O'Hearn and the East London Massive for useful discussions and feedback, and to the anonymous referees of an earlier version of this paper for comments that were very useful in improving the present version.

References

[1] Nuel D. Belnap, Jr. Display logic. *Journal of Philosophical Logic*, 11:375–417, 1982.

[2] James Brotherston. A cut-free proof theory for Boolean BI (via display logic). Unpublished note; available from `http://www.doc.ic.ac.uk/~jbrother`, 2009.

[3] James Brotherston and Cristiano Calcagno. Classical BI (A logic for reasoning about dualising resource). In *Proceedings of POPL-36*, pages 328–339, 2009.

[4] Cristiano Calcagno, Dino Distefano, Peter O'Hearn, and Hongseok Yang. Compositional shape analysis by means of bi-abduction. In *Proceedings of POPL-36*, pages 289–300, 2009.

[5] Bor-Yuh Evan Chang and Xavier Rival. Relational inductive shape analysis. In *Proceedings of POPL-35*, 2008.

[6] Wei-Ngan Chin, Cristina David, Huu Hai Nguyen, and Shengchao Qin. Enhancing modular OO verification with separation logic. In *Proceedings of POPL-35*, 2008.

[7] Dino Distefano and Matthew Parkinson. jStar: Towards practical verification for Java. In *Proceedings of OOPSLA*, pages 213–226. ACM, 2008.

[8] D. Galmiche, D. Mery, and D. Pym. The semantics of BI and resource tableaux. *Mathematical Structures in Computer Science*, 15:1033–1088, 2005.

[9] Didier Galmiche and Dominique Larchey-Wendling. Expressivity properties of Boolean BI through relational models. In *Proceedings of FSTTCS*, 2006.

[10] Rajeev Goré. On the completeness of classical modal display logic. In Heinrich Wansing, editor, *Proof Theory of Modal Logic*, pages 137–140. Kluwer Academic Publishers, 1996.

[11] Rajeev Goré. Gaggles, Gentzen and Galois: How to display your favourite substructural logic. *Logic Journal of the IGPL*, 6(5):669–694, 1998.

[12] Samin Ishtiaq and Peter W. O'Hearn. BI as an assertion language for mutable data structures. In *Proceedings of POPL-28*, 2001.

[13] Marcus Kracht. Power and weakness of the modal display calculus. In Heinrich Wansing, editor, *Proof Theory of Modal Logic*, pages 93–121. Kluwer Academic Publishers, 1996.

[14] Dominique Larchey-Wendling and Didier Galmiche. Exploring the relation between intuitionistic BI and Boolean BI: An unexpected embedding. *Mathematical Structures in Computer Science*, 19:1–66, 2009.

[15] Patrick Lincoln, John C. Mitchell, Andre Scedrov, and Natarajan Shankar. Decision problems for propositional linear logic. *Annals of Pure and Applied Logic*, 56(1–3):239–311, 1992.

[16] P.W. O'Hearn and D. J. Pym. The logic of bunched implications. *Bulletin of Symbolic Logic*, 5(2):215–244, June 1999.

[17] David Pym. *The Semantics and Proof Theory of the Logic of Bunched Implications.* Applied Logic Series. Kluwer, 2002. Errata and remarks (Pym 2004) maintained at `http://www.cs.bath.ac.uk/~pym/reductive-logic-errata.html`.

[18] Greg Restall. Displaying and deciding substructural logics 1: Logics with contraposition. *Journal of Philosophical Logic*, 27:179–216, 1998.

[19] Greg Restall. *An Introduction to Substructural Logics*. Routledge, 2000.

[20] John C. Reynolds. Separation logic: A logic for shared mutable data structures. In *Proceedings of 17th LICS*, 2002.

[21] Alasdair Urquhart. The undecidability of entailment and relevant implication. *Journal of Symbolic Logic*, 49(4):1059–1073, 1984.