# A SURVEY OF PROOF NETS AND MATRICES FOR SUBSTRUCTURAL LOGICS

SEAN A. FULOP

**Abstract.** This paper is a survey of two kinds of "compressed" proof schemes, the *matrix method* and *proof nets,* as applied to a variety of logics ranging along the substructural hierarchy from classical all the way down to the nonassociative Lambek system. A novel treatment of proof nets for the latter is provided. Descriptions of proof nets and matrices are given in a uniform notation based on sequents, so that the properties of the schemes for the various logics can be easily compared.

**§1. Introduction.** This paper provides a survey of two kinds of "compressed" proof schemes, the *matrix method* and *proof nets,* as applied to a variety of logics ranging along the substructural hierarchy [31] from classical all the way down to the nonassociative Lambek system. There appears to be a paucity of survey literature that discusses proof nets for a variety of logics in a uniform notation, and even less which discusses matrix methods in relation to proof nets. It is the author's hope that this paper can provide in one source a host of information and methodology for proof nets and matrices, which could allow further research extending and using these techniques to be more easily conducted. There are few new results presented here, but the available literature provides treatments of various logics which are incommensurate; we hope to rectify this situation by unifying the presentation to a common framework.

Section 2 provides the necessary background, reviewing Gentzen-type sequent calculi for a variety of logics. Section 3 introduces the approaches to "proof compression" which are the main subject of the paper. Section 4 presents the matrix method, which works for both classical and linear logic, in some detail. As background, we also rehearse Smullyan's "unifying notation" [35] of signed formulae which is central to the present paper. Section 5 describes proof nets for a variety of logics, beginning with the canonical case of multiplicative linear logic [16]. The proof nets are defined in a two-sided framework that can be directly applied to two-sided sequents, so that proof nets for the various logics can be readily

compared. From here it is possible to go both up and down the substructural hierarchy; after also considering proof nets for classical logic, two versions of the Lambek calculus are treated. It is observed how the Lambek systems, with their increasingly rigid structural requirements on the layout of the formulae in a sequent, require more strictly geometrical conditions on correct proof nets. The nonassociative Lambek calculus is here provided with a two-sided proof net system and a geometric correctness condition for the first time. The last sections of the paper briefly discuss complexity of proof procedures and the general problem of "identity of proofs."

## §2. Sequent systems.

**2.1. Classical propositional logic.** We begin the discussion with Gentzen's [15] sequent calculus for classical logic. This deductive system permits the proof of special *sequent* statements of the form $\Gamma \Rightarrow \Delta$. In a typical notation, $A, B, \ldots$ stand for proposition occurrences, while $\Theta, \Gamma, \ldots$ stand for sequences of proposition occurrences. A sequent in classical logic is often interpreted metalogically as a statement that the (possibly empty) formula sequence $\Delta$, the *succedent,* follows from the (possibly empty) formula sequence $\Gamma$, the *antecedent,* in a natural deduction or axiomatic system of classical logic. To permit this interpretation, the succedent must be understood as a disjunction of its formulae, while the antecedent must be understood as a conjunction of its formulae. The standard (after Gentzen) presentation of the classical sequent calculus involves sequents, as just described, which may have formulae on either side of the arrow; such a presentation is known as a *two-sided* sequent calculus. Gentzen's rules of inference for the classical sequent calculus may be presented as follows:

DEFINITION 1 (Classical sequent calculus [15]).

$$D \Rightarrow D \quad \text{(Axiom)} \qquad\qquad \frac{A, \Gamma \Rightarrow \Theta, B}{\Gamma \Rightarrow \Theta, A \rightarrow B} \quad (\rightarrow \text{R})$$

$$\frac{\Gamma \Rightarrow \Theta, A}{\neg A, \Gamma \Rightarrow \Theta} \quad (\neg \text{ L}) \qquad\qquad \frac{A, \Gamma \Rightarrow \Theta}{A \wedge B, \Gamma \Rightarrow \Theta}$$

$$\frac{A, \Gamma \Rightarrow \Theta}{\Gamma \Rightarrow \Theta, \neg A} \quad (\neg \text{ R}) \qquad\qquad \frac{B, \Gamma \Rightarrow \Theta}{A \wedge B, \Gamma \Rightarrow \Theta} \quad (\wedge \text{ L})$$

$$\frac{\Gamma \Rightarrow \Theta, A \quad B, \Delta \Rightarrow \Lambda}{A \rightarrow B, \Gamma, \Delta \Rightarrow \Theta, \Lambda} \quad (\rightarrow \text{ L}) \qquad\qquad \frac{\Gamma \Rightarrow \Theta, A \quad \Gamma \Rightarrow \Theta, B}{\Gamma \Rightarrow \Theta, A \wedge B} \quad (\wedge \text{ R})$$

$$\frac{A, \Gamma \Rightarrow \Theta \quad B, \Gamma \Rightarrow \Theta}{A \vee B, \Gamma \Rightarrow \Theta} \quad (\vee \text{ L}) \qquad \frac{\Gamma \Rightarrow \Theta, A}{\Gamma \Rightarrow \Theta, A \vee B}$$

$$\frac{\Gamma \Rightarrow \Theta, B}{\Gamma \Rightarrow \Theta, A \vee B} \quad (\vee \text{ R})$$

The above gives the so-called *logical rules* which show how the operators work. Because the left and right sides of a sequent are considered as sequences, to obtain classical logic one also requires Gentzen's *structural rules*—which are no less logical, in spite of the terms.

DEFINITION 2 (Structural rules for the left side).

(Thinning)

$$\frac{\Gamma \Rightarrow \Theta}{\Gamma, A \Rightarrow \Theta}$$

(Contraction)

$$\frac{\Gamma, A, A \Rightarrow \Theta}{\Gamma, A \Rightarrow \Theta}$$

(Exchange)

$$\frac{\Gamma_1, B, C, \Gamma_2 \Rightarrow \Theta}{\Gamma_1, C, B, \Gamma_2 \Rightarrow \Theta}$$

There are mirror-image structural rules for the right side of sequents also, which are omitted for space reasons here. A sequent calculus proof is then a tree-like figure with initial sequents (possibly axioms) at the top and a conclusion at the bottom called the *endsequent.* To prove a single formula of classical logic, the initial sequents must be axioms, and the endsequent must have this formula as the succedent with an empty antecedent. Some variations of the sequent calculus have been defined in which the antecedent and succedent are sets rather than sequences of formulae (e.g. [39]); in such a presentation the structural rules are "compiled in" to the definition of a sequent, and are not explicitly stated or used.

The only other rule permitted in a sequent calculus is known as "Cut," which certifies a kind of transitivity for sequents:

$$\frac{\Gamma \Rightarrow \Theta, D \qquad D, \Delta \Rightarrow \Lambda}{\Gamma, \Delta \Rightarrow \Theta, \Lambda}$$

Gentzen's important result was his "Hauptsatz" stating that the Cut rule can be eliminated with no loss of logical power for the system. The resulting Cut-free sequent system then enjoys the *subformula property,* meaning that "the formulae occurring in any [Cut-free] proof of a given endsequent are all subformulae of the endsequent" [39] (using an obvious definition of *subformula*). It is plain from inspecting the Cut rule that this cannot be a property of proofs which use it. Thanks to the subformula property,

a classical sequent proof search can be "goal-directed," working upward from the endsequent whose proof we seek by applying the inference rules in reverse. Of course, only Cut-free proofs can ever be discovered in this fashion.

A goal-directed deduction system is often called *analytic,* highlighting the idea that one analyzes the goal sequent to construct (or fail to construct) its proof. The opposite of this is then called a *synthetic* system, in which one works from the premises to the proven expression. A natural deduction system (e.g. [15]) is one example which is usually thought of as synthetic, since it is not generally used to construct a natural deduction proof "upwards" from the conclusion. It is worth noting, however, that natural deductions in the logics considered here can normally enjoy the subformula property, and so natural deduction can be regarded as closer to an analytic system than it is sometimes given credit for being. All of the proof methods discussed in this paper are analytic because our focus is on "compressed" goal-directed proof schemes, and so the logical systems to be discussed will be limited to their Cut-free versions. It is somewhat ironic that, as a referee pointed out, cut-free proofs are generally longer than proofs with cuts, so in one sense the compression of proofs is more difficult in the mathematically more pleasant realm of goal-directed theorem proving.

Smullyan [36] developed a classical deductive system called *analytic tableaux* based upon foundations laid by Beth [4]. It is definitionally equivalent to a Cut-free sequent calculus handling sets of formulae (thus doing without structural rules), but the inference rules are explicitly turned upside-down, highlighting the analytic approach by placing the desired goal expression at the top of the proof (called a tableau).

**2.2. Multiplicative linear logic .** Linear logic was introduced by Girard [16], and has since been the object of much study. For our discussion of compressed proof objects, only selected fragments will be used. We present a two-sided sequent formulation of the *unit-free* multiplicative fragment. It is two-sided in the previously used sense that the derivable sequents have possibly nonempty antecedent and succedent. This logic is commonly named $\mathrm{MLL}^-$.

DEFINITION 3 (Sequent calculus for $\mathrm{MLL}^-$ [27]).

$$D \Rightarrow D \quad \text{(Axiom)} \qquad\qquad \frac{\Gamma, A \Rightarrow \Delta}{\Gamma \Rightarrow A^\perp, \Delta} \quad (^\perp \text{R})$$

$$\frac{\Gamma \Rightarrow A, \Delta}{\Gamma, A^\perp \Rightarrow \Delta} \quad (^\perp \text{L}) \qquad\qquad \frac{\Gamma, A, B \Rightarrow \Delta}{\Gamma, A \otimes B \Rightarrow \Delta} \quad (\otimes \text{L})$$

$$\frac{\Gamma \Rightarrow A, \Delta \quad \Gamma' \Rightarrow B, \Delta'}{\Gamma, \Gamma' \Rightarrow A \otimes B, \Delta, \Delta'} \ (\otimes \text{ R}) \qquad \frac{\Gamma \Rightarrow A, B, \Delta}{\Gamma \Rightarrow A \mathbin{⅋} B, \Delta} \ (⅋ \text{ R})$$

$$\frac{\Gamma, A \Rightarrow \Delta \quad \Gamma', B \Rightarrow \Delta'}{\Gamma, \Gamma', A \mathbin{⅋} B \Rightarrow \Delta, \Delta'} \ (⅋ \text{ L}) \qquad \frac{\Gamma \Rightarrow A, \Delta \quad \Gamma', B \Rightarrow \Delta'}{\Gamma, \Gamma', A \multimap B \Rightarrow \Delta, \Delta'} \ (\multimap \text{ L})$$

$$\frac{\Gamma, A \Rightarrow B, \Delta}{\Gamma \Rightarrow A \multimap B, \Delta} \ (\multimap \text{ R})$$

Linear logic is *substructural,* which means that some of Gentzen's structural rules for classical logic are not allowed. The only one of Gentzen's structural rules that is allowed now is Exchange, so the sequents in essence keep track of formula occurrences, meaning each side of the sequent constitutes effectively a multiset of occurrences. We also have the Cut rule allowed, but it is eliminable as before, and we focus only on the Cut-free version. There follows an example proof of a sequent in MLL$^-$:

$$\frac{\dfrac{\dfrac{A \Rightarrow A \quad B \Rightarrow B}{A \mathbin{⅋} B \Rightarrow A, B} \ (⅋ \text{ R}) \quad C \Rightarrow C}{\dfrac{A \mathbin{⅋} B, C \Rightarrow A, B \otimes C}{\dfrac{A \mathbin{⅋} B \Rightarrow C \multimap A, B \otimes C}{A \mathbin{⅋} B, (B \otimes C)^{\perp} \Rightarrow C \multimap A} \ (\perp \text{ L})} \ (\multimap \text{ R})}}{} \ (\otimes \text{ R})$$

**2.3. Associative Lambek calculus.** Now, we discuss other substructural logics which take away more of the structural rules, both explicit and implicit. An important motivation for these logics is found in linguistics, where they serve as fundamental systems within theories of "categorial grammar" and its extension to "type-logical grammar" [28, 12]. Our first example is a logic that was first introduced as a "syntactic calculus" operating on formulae that were interpreted as linguistic parts of speech [23]. In this guise it is known as the (associative) Lambek calculus.

DEFINITION 4 (Lambek sequent calculus, [23]).

$$D \Rightarrow D \quad \text{(Axiom)} \qquad \frac{\Delta \Rightarrow B \quad \Gamma[A] \Rightarrow C}{\Gamma[\Delta, B \backslash A] \Rightarrow C} \ (\backslash \text{L})$$

$$\frac{\Delta \Rightarrow B \quad \Gamma[A] \Rightarrow C}{\Gamma[A/B, \Delta] \Rightarrow C} \ (/\text{L}) \qquad \frac{B, \Gamma \Rightarrow A}{\Gamma \Rightarrow B \backslash A} \ (\backslash \text{R})$$

$$\frac{\Gamma, B \Rightarrow A}{\Gamma \Rightarrow A/B} \ (/\text{R}) \qquad \frac{\Gamma[A, B] \Rightarrow C}{\Gamma[A \bullet B] \Rightarrow C} \ (\bullet \text{L})$$

$$\frac{\Gamma \Rightarrow A \quad \Delta \Rightarrow B}{\Gamma, \Delta \Rightarrow A \bullet B} \quad (\bullet R) \qquad \frac{\Delta \Rightarrow A \quad \Gamma[A] \Rightarrow C}{\Gamma[\Delta] \Rightarrow C} \quad (\text{Cut})$$

Lambek calculus (notated simply L, or $L_\epsilon$ when empty antecedents are permitted) is a positive logic in which none of Weakening, Contraction, or Exchange are permitted, so the logical consequence relation involves sequences of formulae. Thus we introduced the standard notation $\Gamma[\cdot]$, which means a formula sequence with a place identified for substitution which is matched by another use of the similar notation in the same inference rule. Once again, the logic enjoys Cut elimination so we deal solely with the Cut-free version. Associativity of the sequences is assumed, as a kind of implicit structural rule; we show what happens next when even this is removed.

**2.4. Nonassociative Lambek calculus.** The last system to be introduced is a version of Lambek calculus from which even the implicit structural rule of associativity is taken away. This nonassociative Lambek system NL was first described in 1961 [24] where it was motivated by linguistic applications, and it has since been recognized as fundamental within the area of type-logical grammars for linguistics [26, 12]. This system is especially useful for grammatical deductions because, without associative sequences, the sequent calculus handles binary trees of formulae that can be used to represent the syntactic structures of languages. The sequent presentation below does without the product operator '$\bullet$', because this is logically superfluous in a sequent formulation (as it is even in the associative system L above). The nonassociativity of the sequents is here emphasized by replacing the usual comma with the sequent-level operator '$\diamond$'. The sequent system enjoys Cut-elimination and is single-conclusion, so that all provable sequents have a single succedent formula.

DEFINITION 5 (NL sequent calculus, [24]).

$$A \Rightarrow A \quad (\text{Axiom}) \qquad \frac{\Delta \Rightarrow B \quad \Gamma[A] \Rightarrow C}{\Gamma[(\Delta \diamond B\backslash A)] \Rightarrow C} \quad (\backslash \text{ L})$$

$$\frac{\Delta \Rightarrow A \quad \Gamma[A] \Rightarrow C}{\Gamma[\Delta] \Rightarrow C} \quad (\text{Cut}) \qquad \frac{(\Gamma \diamond B) \Rightarrow A}{\Gamma \Rightarrow A/B} \quad (/ \text{ R})$$

$$\frac{\Delta \Rightarrow B \quad \Gamma[A] \Rightarrow C}{\Gamma[(A/B \diamond \Delta)] \Rightarrow C} \quad (/ \text{ L}) \qquad \frac{(B \diamond \Gamma) \Rightarrow A}{\Gamma \Rightarrow B\backslash A} \quad (\backslash \text{ R})$$

**§3. Proof compression.** A key application of analytic deductive methods has been automated deduction. A significant problem for the sequent/tableau systems in this arena is the inefficiency resulting from a large search space. Much progress has been made in the development of

efficient proof search by application of goal-directed logic programming methods such as resolution (e.g. [13]). The primary focus here will not be on efficient search for complete proofs, but rather on the question of how can an analytic proof be compressed, and thereby become a fundamentally different sort of object that can be viewed in a new way, and possibly constructed more efficiently.

It has been explained many times in the literature (e.g. [11]) that, even restricting attention to Cut-free proofs, the sequent and tableau calculi may validate numerous proofs of a given sequent. These several proofs may differ either "trivially" or non-trivially in the order of application of some of the rules. The propensity for the sequent/tableau systems to suffer from *spurious ambiguity* caused by trivial rule permutation has been explained in detail elsewhere [39], so here we simply note the fact and consider its ramifications and proposed remedies. In this paper, we will consider two kinds of "compressed proof objects," which differ philosophically with respect to the spurious ambiguities just mentioned. The first of these, the *matrix method,* constructs a minimal compressed proof object that is really nothing beyond a provability test. There is a compelling argument that a matrix proof of a sequent is not really a proof anymore, because not only trivially different, but also nontrivially different sequent/tableau proofs are all collapsed. A provable sequent has, by definition, exactly one matrix demonstrating its validity.

The second kind of compressed proof object to be considered here is the *proof net.* Proof nets were originally described for linear logic [16], and it has been claimed for that system that they compress proofs to "just the right extent," in the sense that any two sequent/tableau proofs which are *nontrivially different* will have distinct corresponding proof nets, while any two sequent/tableau proofs which differ only *spuriously* (i.e. by "harmless" permutations of the rules) will have the same proof net corresponding [37]. The philosophy behind proof nets is to always seek, if not achieve, this correspondence for the logic at hand, because a proof net is supposed to be something beyond a minimal provability test— proponents think of it as "the essence of a proof." There is, however, no current consensus among logicians as to what precisely should count as a *nontrivial* difference between two proofs.

§4.  **Matrix methods.** In classical and intuitionistic logic, the redundancies and other difficulties with standard proof calculi led to the matrix methods, independently invented by Bibel [5] and Andrews [3], but perfected by Wallen [39]. Here we follow Wallen's exposition, and the unifying notation of Smullyan's signed formulae will be of utmost importance.

**4.1.  Unifying notation of signed formulae.** It would be redundant to present the inference rules of Smullyan's tableau calculus for classical

logic, since they are essentially the same as those of Gentzen's sequent calculus. One important element of Smullyan's treatment that will be important for our discussion throughout, however, is his "unifying notation" which uses *signed formulae* classified into two basic varieties [35]. A signed formula is just a formula $P$ which is annotated by a sign, or *polarity,* which we will show as either $P^+$ or $P^-$. The sign is used to indicate the "negation environment" of the formula occurrence within a sequent, so that negative formula occurrences are all those that are within the scope of an explicit or implicit negation. An "implicit negation" environment is always (and only) the antecedent of a conditional or of a sequent. This definition comes from the truth-functional equivalence between formulae $A \rightarrow B$ and $\neg A \vee B$. Smullyan used signed formulae to avoid writing sequents directly with the sequent arrow; his rules for positive formulae exactly mirror the succedent (R) rules in the sequent calculus, while tableau rules for negative formulae mirror antecedent (L) rules in the sequent calculus.

Signed formulae are then classified by Smullyan into two fundamental kinds, which can be determined by inspecting the sequent rules shown above. The key question is whether the inference rule "branches," having two premisses. A branching rule governs a "signed formula of type B," which we may call *disjunctive,* after the canonical example of the rule ($\vee$ L). A rule with only one premiss, on the other hand, governs a "signed formula of type A," which we may term *conjunctive.* The conjunctive signed formulae in classical logic are these:

$$(A \wedge B)^-, (A \vee B)^+, (A \rightarrow B)^+, (\neg A)^+, (\neg A)^-$$

The disjunctive formulae are the others:

$$(A \vee B)^-, (A \wedge B)^+, (A \rightarrow B)^-$$

**4.2. Classical logic matrices.** Step one of the matrix method, and also ultimately of the proof net method, is to decompose the target sequent or formula into a tree of its subformulae that keeps track of the signs.

DEFINITION 6 ([39]). *A formula tree for a signed formula $A^g, g \in \{+, -\}$ is a tree of subformulae of A together with an assignment of a sign to each position $k$ of the formula tree. Each position $k$ then contains a signed formula* occurrence *from within A; the formula occurrence apart from its sign at $k$ is called the* label *of $k$. Let* lab$(k)$ *and* sgn$(k)$ *denote the label and sign of position $k$ respectively. For each position $k$, if* lab$(k)$ *occurs positively in $A^g$, then* sgn$(k) = g$. *If* lab$(k)$ *occurs negatively in $A^g$, then* sgn$(k)$ *is the opposite sign from $g$.*

DEFINITION 7 ([39]). *A path through formula $A^g$ is a subset of the positions of its formula tree, defined inductively:*

1. $\{k_0\}$, *the root position, is a path.*
2. *If $s, \alpha$ is a path, so is $(s - \{\alpha\}), \alpha_1, \alpha_2$, for conjunctive $\alpha$ having $\alpha_1, \alpha_2$ as immediate subformulae.*
3. *If $s, \alpha$ is a path, so is $(s - \{\alpha\}), \alpha_1$, for conjunctive $\alpha$ having $\alpha_1$ as its sole immediate subformulae (this is the case where $\alpha$ is a negation).*
4. *If $s, \beta$ is a path, so is $(s - \{\beta\}), \beta_1$, and so is $(s - \{\beta\}), \beta_2$, for disjunctive $\beta$ having $\beta_1, \beta_2$ as immediate subformulae.*

*The second through fourth clauses above can be regarded as path reduction steps. A completely reduced path will consist entirely of (signed) positions labeled by atoms, and is called an atomic path.*

The above formulation can be easily extended from signed formulae to two-sided sequents of signed formulae. The simplest way is to decompose each of the antecedent formulae and succedent formulae separately. The antecedent formulae are negatively signed, while the succedent formulae are positively signed, and one decomposes the whole set of signed formulae into a set of formula trees as above, treating the compound tree as a single graph with "multiple roots." The above definitions of a path through the tree and an atomic path can then be applied mutatis mutandis.

DEFINITION 8 (Matrix representation [39]).

1. *If signed formula $A^g$ is conjunctive, its matrix representation is a row matrix having as element(s) the signed component(s) found immediately below in its formula tree.*
2. *If signed formula $A^g$ is disjunctive, its matrix representation is a column matrix having as elements the signed components found immediately below in its formula tree.*
3. *If signed formula $A^g$ is atomic, it is its own matrix representation.*

A completed matrix for a signed formula must have every subformula in every submatrix converted to matrix representation; matrices are to be nested as needed. The matrix representation extends to signed two-sided classical sequents by a simple adaptation of the procedure described above for sequent trees. The matrix of a sequent is then simply a "row matrix" whose elements are the respective matrices of the constituent formulae. This fact can be related to the "metalogical" view of a sequent in which the antecedent formulae are conjoined while the succedent formulae are disjoined; observe that a conjunction in the antecedent and a disjunction in the succedent are each formulae of conjunctive type, and so a row matrix is the correct form for each.

Every atomic path through a signed formula (or sequent) is now represented by the sequence of signed atoms encountered on a left-right sequence of steps through the columns in its completed matrix—submatrices are to be stepped through as well in this procedure [39]. When a step

enters a column matrix, only one row is selected (nondeterministically) for the atomic path, while the other is ignored.

The key idea at the core of the matrix method is that of a spanning set of connections.

DEFINITION 9. *A connection is a pair of atomic positions in some path through $A^g$, which have the same label but opposite signs. A given set of connections is said to* span *$A^g$ iff every atomic path through $A^g$ contains a connection from the set.*

THEOREM 10 ([39]). *For signed propositional formula $A^+$, the existence of a spanning set of connections for it ensures its provability in classical logic, and vice versa.*

The above definition and theorem concerning a spanning set of connections also extends in a simple fashion mutatis mutandis to signed two-sided sequents. For clarity, this may be stated as follows:

COROLLARY 11. *For signed sequent $\Gamma^- \Rightarrow \Delta^+$, the existence of a spanning set of connections for it ensures its provability in classical sequent calculus, and vice versa.*

An example sequent provable in classical logic is shown in (1); the corresponding matrix of the sequent is shown in (2).

(1)   $\neg A,\ B \to A \Rightarrow \neg B$

(2)   $\begin{bmatrix} [A^+] & \begin{bmatrix} B^+ \\ A^- \end{bmatrix} \end{bmatrix} \quad [B^-]$

This matrix presents two atomic paths: $(A^+, B^+, B^-)$ and $(A^+, A^-, B^-)$. The spanning set of connections is then $\{\langle A^+, A^- \rangle, \langle B^+, B^- \rangle\}$, the existence of which shows the sequent to be provable in classical logic. Any sequent of classical propositional logic can be tested for provability using our adaptation of Wallen's matrix method to two-sided sequents. The sequent calculus (or tableaux method) can now be regarded as extremely inefficient methods of checking that every atomic path through the goal sequent tree contains a connection from a spanning set!

**4.3. MLL$^-$.** The matrix method for linear logic was presented by Galmiche [14]. The matrix representation of a sequent is obtained from the signed sequent tree just as with classical logic. The matrix for the sequent proven above in subsection 2.2 is shown in (3); a spanning set of connections for this matrix is given in (4).

(3)   $\begin{bmatrix} A^- \\ B^- \end{bmatrix} \quad \begin{bmatrix} B^+ \\ C^+ \end{bmatrix} \quad \begin{bmatrix} C^- & A^+ \end{bmatrix}$

(4)   $\{\langle A^+, A^- \rangle, \langle B^+, B^- \rangle, \langle C^+, C^- \rangle\}$

Reflecting the differences between MLL$^-$ and classical logic, it is no longer sufficient that the sequent matrix possesses a spanning set of connections, however. Galmiche stated the additional requirements for the set $\mathcal{S}$ of connections to *linearly span* a matrix in the following way: 1) All atomic occurrences in the matrix occur exactly once with each polarity in $\mathcal{S}$; 2) no pair of connections in $\mathcal{S}$ has overlapping elements; 3) $\mathcal{S}$ is a minimal spanning set. It is easy to see that the set of connections in (4) does linearly span the above matrix for the sequent. Galmiche stated the theorem that a sequent of MLL$^-$ is provable just in case its matrix possesses a set of connections which linearly spans it. It is notable that, despite presenting this as a proven fact, Galmiche never really proved it in his paper. It is nevertheless possible for us to appreciate, at a glance at least, how the additional conditions defining a linearly spanning set derive from the characteristic that MLL handles multisets of formulae (cf. condition 1), effectively keeping track of formula occurrences and not allowing contractions of repeated formulae (cf. condition 2) or extraneous formulae (cf. condition 3) into a proof.
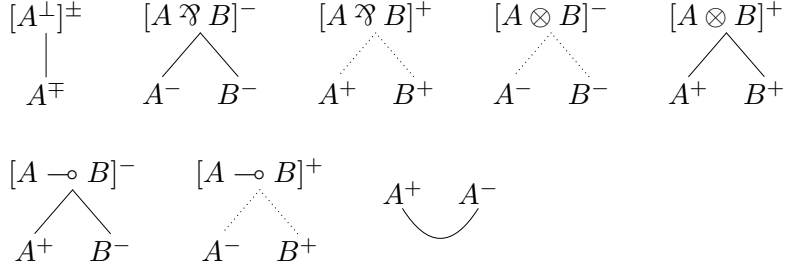
Having discussed the matrix method and signed formulae, it will be much easier to understand proof nets, a subject to which we turn next.

§5. **Proof net methods.** The sequent and tableau methods yield too many possible proofs of a given sequent, and have an undesirable amount of notational redundancy for automated theorem proving applications. The matrix method described above has certainly eliminated the redundancy, but now there are in a sense too few proofs of a given sequent for some applications; in fact, each provable sequent has precisely one matrix. This may be acceptable for theorem proving, but there are theoretical reasons to desire a proof representation that captures "the essence of a proof." This notion is related to the general problem of the identity of proofs [10], and the compressed proof objects known as *proof nets* have been put forth as solving this problem for MLL, at least [37].

**5.1. MLL$^-$.** Our discussion of proof nets must begin with MLL$^-$, since Girard [16] developed linear logic and proof nets at the same time. The proof nets are in fact a subclass of the decomposition graphs known as proof structures. Just as the matrix of a sequent is produced from the decomposition of the signed formulae while distinguishing conjunctive from disjunctive formulae, a *proof structure* is a special graph that is drawn from a formula or sequent decomposition, also keeping track of the polarities and the conjunctive or disjunctive nature of the signed formulae. The subgraphs which are drawn for each type of formula decomposition are traditionally called *links*; to complete the proof structure, pairs of atoms having opposite polarity are linked together by edges called *axiom links*.

The MLL$^-$ link schemata which may be used to decompose formulae in a proof structure are shown below. It should be mentioned that these link drawings are to be viewed as representing graphs without further geometric properties, so that the specific orientation of a link drawing or whether edges cross is unimportant. A complete proof structure for the provable MLL sequent already studied above follows the link presentation below. The resulting graph has two sorts of edges, which serve to distinguish the conjunctive from the disjunctive binary formula occurrences (negation is excluded from the conjunctive/disjunctive dichotomy for this purpose). The binary links shown with solid lines are the disjunctive formulae, traditionally called *times* links (typified by the link for the times connective $\otimes$ in a positive context), while the dotted lines show the conjunctive formulae, traditionally called *par* links (and typified by the link for the par $\invamp$ connective in a positive context). Axiom links are shown with curved lines in the proof structures. Some formal definitions follow.

**MLL$^-$ links:**

$$[A^\perp]^\pm \quad [A \invamp B]^- \quad [A \invamp B]^+ \quad [A \otimes B]^- \quad [A \otimes B]^+$$
$$A^\mp \qquad A^- \quad B^- \qquad A^+ \quad B^+ \qquad A^- \quad B^- \qquad A^+ \quad B^+$$

$$[A \multimap B]^- \quad [A \multimap B]^+ \qquad A^+ \quad A^-$$
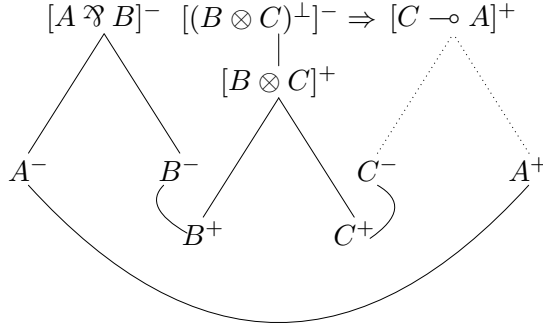$$A^+ \quad B^- \qquad A^- \quad B^+$$

Formula(e) on top of each link are called *conclusion,* and formulae on the bottom of a link are called *premiss.* The axiom link is unique in having no premisses and two conclusions.

**MLL$^-$ proof structure:**

DEFINITION 12 ([27]). *A proof structure $\langle S, \mathcal{L} \rangle$ consists of a set $S$ of signed formulae and a set $\mathcal{L}$ of links over $S$ (from the above possibilities). A proof structure must also satisfy the conditions:*

- *Every formula in $S$ is at most once the premiss of a link;*
- *Every formula in $S$ is exactly once the conclusion of a link.*

$$[A \,℘\, B]^- \quad [(B \otimes C)^\perp]^- \;\Rightarrow\; [C \multimap A]^+$$

$$[B \otimes C]^+$$

$$A^- \qquad B^- \qquad\qquad C^- \qquad\qquad A^+$$

$$B^+ \qquad\qquad C^+$$

The proof structure exemplified above is *two-sided,* because it can be created for a sequent with formulae on both sides of the arrow $\Rightarrow$. It is possible to enumerate all possible proof structures for any sequent now by decomposing all connectives until we reach the atomic formulae, and then connecting positive to negative atoms using axiom links [27]. The two-sided means of presenting a proof structure is, however, not common in literature on linear logic, and has never been fully described in published literature.[1] In the literature, MLL proof structures are almost invariably one-sided—meaning they can be constructed only for a one-sided sequent calculus with empty antecedents. Moreover, such structures cannot involve the implication or negation operators overtly as above, because they furthermore do not keep track of the polarities of formulae. For our purposes, the usual one-sided proof structures for MLL obscure the fundamental relationship with matrix methods and the unifying notation of signed formulae, so here we stick with the two-sided dialect.

Completing a proof structure for an MLL sequent is an important step toward demonstrating provability of the sequent, but it is not yet sufficient. A proof structure for a provable sequent is known as a *proof net,* and only those structures which satisfy an additional correctness condition are indeed proof nets. An impressive list of alternative correctness conditions for MLL proof nets has arisen from years of research on the topic, beginning with the original "long trip" condition of Girard [16]. This condition is somewhat cumbersome for our purpose here, so we will first describe a popular correctness condition due to Danos and Regnier [8].

Take a proof structure as a graph; let us call it $\mathcal{P}$. Now, let $\sigma(\mathcal{P})$ be a new graph derived from $\mathcal{P}$ by deleting some edges. Specifically, $\sigma$ acts to delete one edge nondeterministically from each par link in $\mathcal{P}$, and is called a *DR switching.*

---

[1]The presentation here is derived from class lecture notes [37].

THEOREM 13 (Danos-Regnier correctness condition). *A proof structure $\mathcal{P}$ is a proof net if and only if every DR switching $\sigma(\mathcal{P})$ of it yields a connected acyclic graph.*

The Danos-Regnier switching condition is easy to apply to small proof structures—the structure presented above is easily seen to satisfy it—but has exponential complexity because of the need to check the result of every DR switching of a proof structure for acyclicity and connectedness [27].

A more efficient condition was first presented in the PhD dissertation of Danos [7], and involves transforming a candidate proof structure by graph contractions (v. [17] for a formal definition of graph contraction). The two Danos contraction rules are presented as follows in [27]:
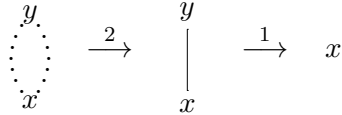


The basic idea is that two 'par' edges transform to a single 'times' edge just if they connect the same two vertices (this can only result from previous contractions), and any two vertices linked by a 'times' edge contract to one vertex.

THEOREM 14 (Danos contraction condition). *A proof structure is a proof net if and only if it contracts to a single vertex by successive application of the Danos contraction rules above.*

The figure sequence below shows the successive contraction of the proof structure presented above; formulae are irrelevant for this condition, and are replaced by simple vertex labels. The equivalence between the Danos-Regnier switching condition and the Danos contraction condition was proven very simply in [37].
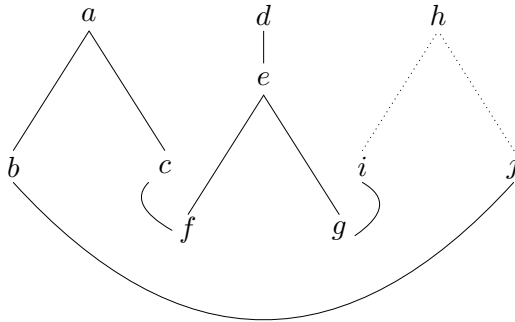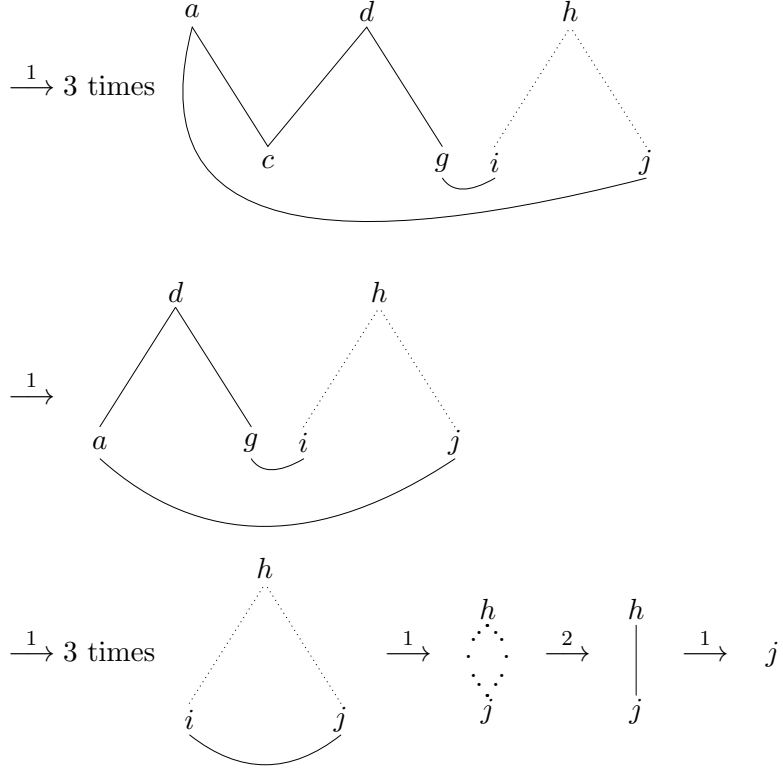
**5.2. Intuitionistic MLL.** We refer back to the sequent calculus rules for MLL$^-$; this system is rendered intuitionistic by endowing it with the single-conclusion property, by which all sequents must have just one succedent formula [27]. The positive fragment of this system with only $\otimes$ and $\multimap$ is known in the literature as multiplicative *intuitionistic* linear logic (MILL), and it has some thinly disguised early roots.

A kind of decomposition graph for MILL formulae was published by Kelly and Mac Lane in 1971 [21] in their study of coherence in categories, and is possibly the first work on a compressed proof object showing aspects of the matrix and proof net methods. The Kelly-MacLane graph of a MILL formula shows its decomposition to signed atoms; if linking atoms in opposite polarity pairs can be achieved, then one has essentially a proof structure, but a correctness criterion is still required for such a structure to be a proof net [27].

To build a proof structure in MILL, one begins as in MLL by decomposing the sequent into subformulae down to the atoms while keeping track of the polarities and the conjunctive/disjunctive property of the formula at each stage. The antecedent formulae are first given a negative sign while the succedent formula is given a positive sign. Beyond this there are just two operators, and the decomposition proceeds so that $[X \otimes Y]^{\pm}$

yields $X^{\pm}, Y^{\pm}$, while $[X \multimap Y]^{\pm}$ yields $X^{\mp}, Y^{\pm}$, as above in MLL. Signed formulae of the form $[X \otimes Y]^{-}, [X \multimap Y]^{+}$ are the conjunctive ones as in MLL, which are assigned a par link. The formal definition of a proof structure in MILL (without units) is the same as the above for MLL$^{-}$ mutatis mutandis, and either of the above correctness conditions for MLL proof nets carries over to the case of MILL [27].

Below we show two proof structures for posited sequents of MILL; only the first one is a correct proof net, in which each DR switching yields a connected acyclic graph. The second proof structure has a cycle following removal of the right branch of the par link, demonstrating the posited sequent to be underivable in MILL. We see that MILL proof nets are merely a subspecies of MLL nets, however, one reason to discuss this logic separately here is to highlight the much earlier literature [21] that first defined proof structures for this system, and was also first to make use of signed formulae in a linear logic system. MILL is in a sense also the archetypal logic in this family possessing the single-conclusion property.
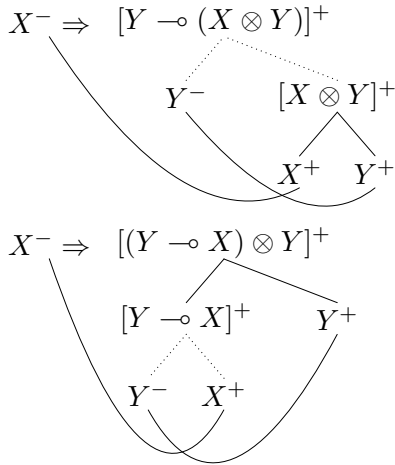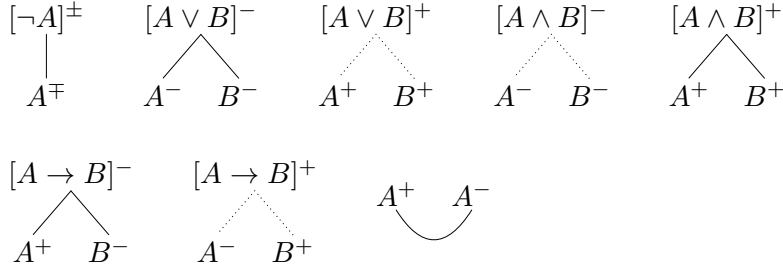


**5.3. Classical logic.** Classical (propositional) logic is actually quite similar to linear logic; all of the differences derive from the presence of Gentzen's structural rules of Weakening and Contraction. It is interesting to see how the definition of a proof net carries over to this case. Proof nets for classical logic have been developed by Robinson [32] following the two-sided paradigm given above for linear logic, in which there are distinct links for decomposing each connective in a positive versus a negative context. The system therefore derives naturally from Smullyan's unifying notation for classical logic, although Robinson did not cite this prior literature. The conjunctive and disjunctive links for the decomposition of signed formulae are very similar to the ones needed for MLL, and are presented below with adjustments to suit our notation here (leaving

aside the degenerate links which would handle the true and false units, not used here).

**Classical logic links:**

$$[\neg A]^{\pm} \qquad [A \vee B]^{-} \qquad [A \vee B]^{+} \qquad [A \wedge B]^{-} \qquad [A \wedge B]^{+}$$

$$A^{\mp} \qquad A^{-} \quad B^{-} \qquad A^{+} \quad B^{+} \qquad A^{-} \quad B^{-} \qquad A^{+} \quad B^{+}$$

$$[A \to B]^{-} \qquad [A \to B]^{+} \qquad A^{+} \quad A^{-}$$

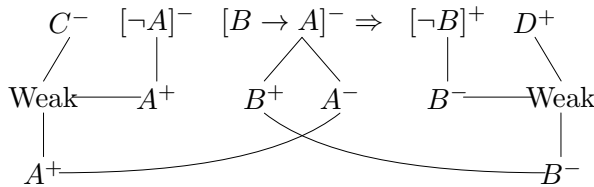$$A^{+} \quad B^{-} \qquad A^{-} \quad B^{+}$$

As Robinson showed, more is needed to obtain a kind of proof net that enjoys the same correctness conditions which govern MLL. Specifically, Robinson added links corresponding to the structural rules of Contraction and Weakening; the former are conjunctive while the latter are disjunctive. Once again our presentation changes the link notation somewhat to make it uniform with the presentation of MLL.

**Structural links:**

$$A^{\pm} \qquad B^{\pm} \quad A^{\pm}$$

$$[\text{Cont}]^{\pm} \qquad [\text{Weak}]^{\pm}$$

$$A^{\pm} \quad A^{\pm} \qquad B^{\pm}$$
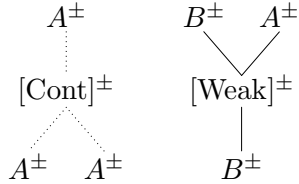
With these additions, a proof structure can be constructed for a classical sequent, relying on Definition 12 from the MLL case. The correctness conditions it must meet to be a proof net for a provable sequent are also carried over from MLL with no changes. An example is now shown.

**Classical proof net:**

$$C^{-} \quad [\neg A]^{-} \quad [B \to A]^{-} \Rightarrow [\neg B]^{+} \quad D^{+}$$

$$\text{Weak} \quad\quad A^{+} \qquad B^{+} \quad A^{-} \qquad B^{-} \quad\quad \text{Weak}$$

$$A^{+} \quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad B^{-}$$
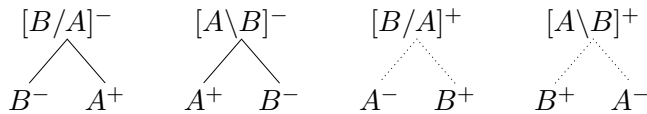
This classical proof net turns out to have no conjunctive links, so it has to be a connected acyclic graph as it is shown according to the Danos-Regnier condition, and indeed it is. Robinson also gave an elegant, simple explanation connecting this correctness condition to the unifying notation, to be restated now. If a proof net comes from a proof, the graph must be simply connected, which forces the switching condition in the following way. A disjunctive ('times') signed subformula $A \circ B$ for any operator $\circ$ has "branched" premises which come from separate subproofs, and so are not yet connected, so the occurrence of $A \circ B$ must be joined to both premises

otherwise the proof net would end up disconnected. On the other hand, a conjunctive ('par') signed subformula $C \circ D$ has premisses coming from the same subproof, so they are already connected. The formula $C \circ D$ must then be joined to exactly one premiss or the graph will contain a cycle. This explanation is also applicable to the linear logic cases. It is interesting that the only real difference between the $\mathrm{MLL}^-$ and classical proof net systems is the addition of the special links for Contraction and Weakening.

**5.4. Associative Lambek calculus.** So far we have discussed classical logic, which in essence handles formulae in sets, and two varieties of linear logic, which remove the Weakening and Contraction rules, and thereby keep track of occurrences of formulae. It is useful to note at this juncture that these logics have both matrix and proof net methods available for checking provability of sequents, neither of whose correctness conditions refer crucially to the geometrical arrangement of the proof object. It is apparent that a matrix of a sequent does not have any interesting geometrical properties; moreover, although a proof net is a kind of graph, there is nothing very "geometrical" about these proof nets so far. It is unimportant whether the link lines in a drawing of the graph cross, for example.

In fact, $\mathrm{L}_\epsilon$ is basically MILL without Exchange. The lack of Exchange (or "commutativity") has effectively split the linear implication into a pair of directionally sensitive implications notated with the slash operators. Each slash is interpreted as saying that the formula on top of the slash results when the formula under it is adjacent on that side. The newfound sensitivity of the logic to the arrangement of formulae in a sequence has a profound effect on the definition of a compressed proof procedure. Below, the binary links for proof nets in the Lambek system L are provided, following Roorda [33, 34]; this time, however, the geometry of the drawings as shown provides important information. The left-right arrangement of the subformulae in a decomposition link is now critical; one must swap the order of the subformulae with respect to the parent formula *in the positive links only*.[2]
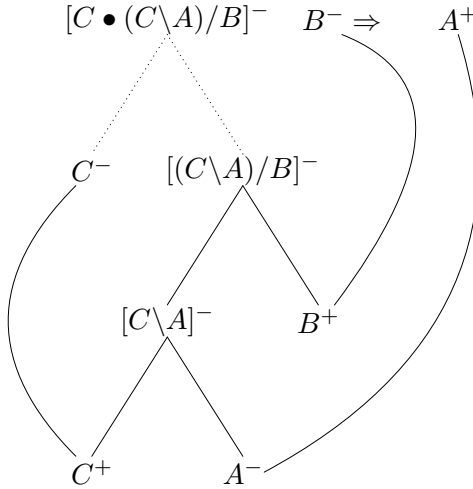
**Links for $\mathrm{L}_\epsilon$:**

$$[B/A]^- \qquad [A\backslash B]^- \qquad [B/A]^+ \qquad [A\backslash B]^+$$

$$B^- \quad A^+ \qquad A^+ \quad B^- \qquad A^- \quad B^+ \qquad B^+ \quad A^-$$

---

[2]One of the very few sources to provide these link drawings for Lambek proof nets [27] has got this condition backwards, unfortunately.

$[A \bullet B]^+$        $[A \bullet B]^-$

$B^+$    $A^+$        $A^-$    $B^-$

$A^+$    $A^-$
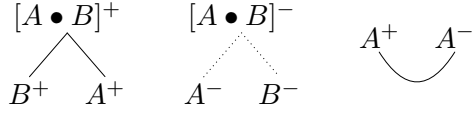
**L proof structure** It seems that the formal definition of a proof structure in L can be kept the same as for the systems above. An example is now shown.

$[C \bullet (C\backslash A)/B]^-$    $B^- \Rightarrow$    $A^+$

$C^-$    $[(C\backslash A)/B]^-$

$[C\backslash A]^-$    $B^+$

$C^+$    $A^-$

This example is in fact also a proof net for the provable sequent. This proof net satisfies the Danos-Regnier condition plus an additional requirement of planarity which was first proven necessary by Roorda [33]; each DR-switching graph is not only acyclic connected, but also planar as shown in the drawing.

Although this treatment here applies generally to the system $L_\epsilon$ allowing empty antecedents, it has been shown [22] that we may exclude all sequents with empty antecedent by an additional requirement about *subnets* of a proof net. A subnet is, in our notation, a down-closed subset of the nodes such that axiom links stay inside the substructure. Then, to exclude sequents with empty antecedent, it is sufficient to require that every subnet of a proof net possess at least two conclusions (i.e. local root formulae at the top).

A different presentation of a noncommutative linear logic was also shown to require planar proof nets [1], around the same time as Roorda's result about the Lambek system. The NCMLL system described in the reference is equivalent in its multiplicative fragment to another noncommutative logic [30], which in turn is a conservative extension of $L_\epsilon$ [2]. Thus it is beginning to look as if noncommutativity of the logic (i.e. lack of Exchange) leads directly to a new geometric requirement of planarity of the proof net. It is also not at all obvious that a version of the matrix

method could somehow be formulated for this kind of logic, for now the specific arrangement of the formulae is critical.

**5.5. Nonassociative Lambek calculus.** Despite having been discussed many times in the literature, the system NL has never had a proof net scheme defined for it in a way that relates clearly with the other proof net schemes presented here.[3] It turns out to be quite easy to adapt the proof nets for associative L, but an additional correctness condition is required that has never been developed in literature, and which makes the resulting nets even more "geometrical." This is the only novel result to be introduced in the present survey.

Let us discuss several examples of NL proof structures to develop the additional correctness condition. Example 1 shows the basic kind of structure for a provable sequent, which is planar just as in system L. A further correctness condition is needed in order to account for the effects of the parentheses, which govern the nonassociative structure of the antecedent. To develop this extra condition, we draw dotted boundaries from each pair of parentheses in the antecedent, extending around the first decomposition link whose active conclusion subformula is governed by that pair. Such boundaries in our proof nets will be called *parenthetical boundaries.* Examining the axiom links in the final structure, observe that only the link coming from the negative $B$ atom, which connects to the succedent, crosses the parenthetical boundary that contains it.
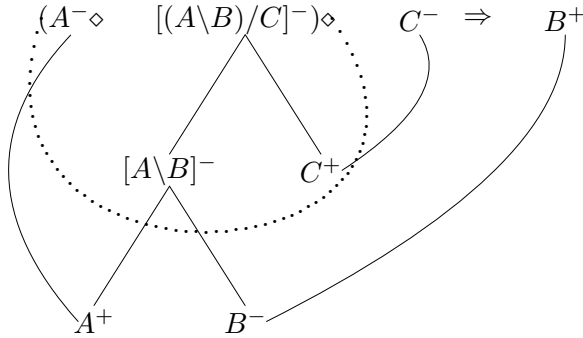
**Example 1**
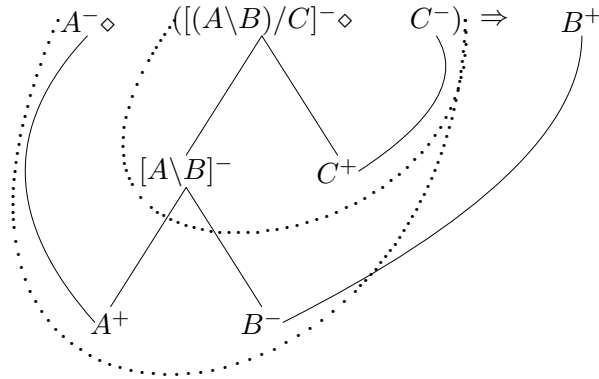


---

Example 2, by contrast, shows a similar proof structure for a non-provable sequent whose antecedent has the parentheses wrong. The structure is indeed planar, so the sequent would be provable in system L by invoking associativity, but observe that now both of the antecedent axiom links cross the first boundary (the outer boundary is not shown). The problem, in reality, is the $C$-link, because *the link from the positive atom crosses the parenthetical boundary which contains it.* We therefore state this as part of the correctness conditions.

THEOREM 15. *An NL-proof structure is a proof net for the decomposed sequent just in case:*

- *the Danos-Regnier switching condition, or other equivalent condition, holds of the structure;*
- *the structure is planar;*
- *no axiom link from a positive atom crosses the parenthetical boundary which contains it.*

Example 3 illustrates the structure for a more complicated provable sequent, and we observe that two axiom links cross boundaries, but neither involves a link from a positive atom crossing the boundary which contains it.

**Example 2**



**Example 3**
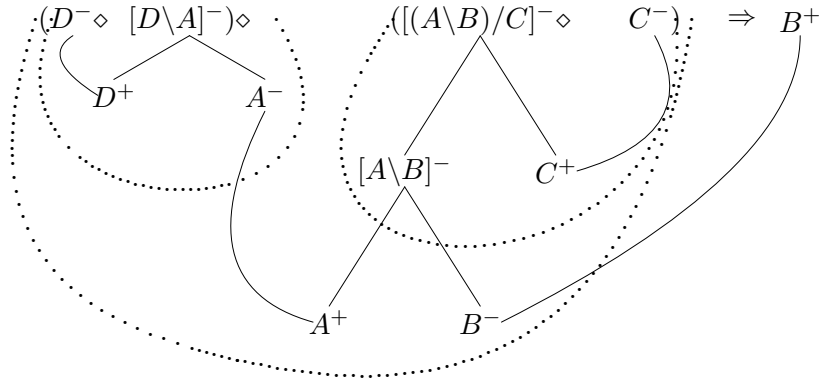
It is quite easy to see the necessity of this correctness condition, so a brief explanation should suffice here. Note first that for each slash operator in a provable NL sequent, there must be a pair of parentheses surrounding a formula which contains it, and also surrounding the neighboring occurrence of the subformula under the slash. Every atomic subformula under a top-level slash (i.e. one not itself within a proper subformula) in the antecedent of the sequent will decompose to a positive signed atom in the proof net, while the neighboring atom of the same label will show the opposite sign (cf. Example 1). With the parentheses in the right place, an axiom link connecting the two atoms will not cross a boundary determined by them; with parentheses in the wrong places, the positive atom will be contained within a boundary which does not contain its counterpart negative (cf. Example 2). This argument extends by a structural induction to more complicated formulae. In essence, the device of the boundaries is a way of checking the grouping action of the parentheses in the proof net.

Example 4 shows that care must be taken to draw the parenthetical boundaries when subformulae involving "useless types" occur in the sequent. Observe that the subformula $C/D$ has types $C$ and $D$ which are "useless," in that they do no work in reducing the sequent. When this is the case, we must draw the boundary all the way around the axiom links which connect the decompositions of the occurrences of $C/D$. After drawing the boundaries appropriately, we observe that once again no axiom link coming from a positive atom crosses a boundary which contains it. The proof structure for this provable sequent is then a proof net, under our newly formulated condition.

**Example 4**
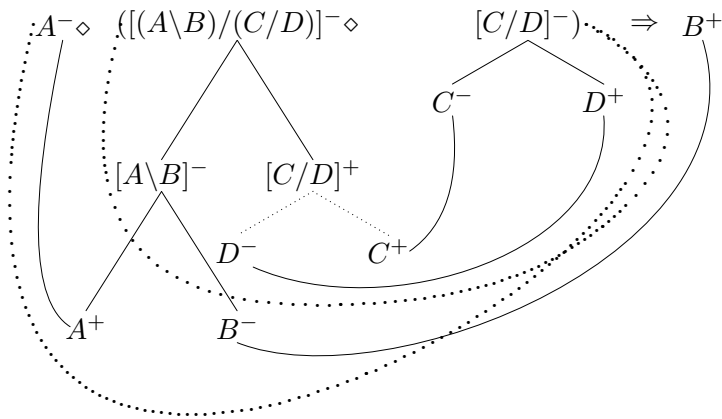


We have at last descended all the way down the substructural hierarchy. As the logics became more stringent in dealing with a specific arrangement of formulae, the conditions on proof nets became accordingly more

geometrical. Moreover, we noted that for those logics that do not deal with a specific arrangement of the formulae, it was possible to invoke the extremely compressed proof format of the matrix method. It is the author's hope that this unified discussion has illuminated the ways in which the "geometry" required of formulae in a logical consequence relation ends up being encoded into the "geometry of proofs" validating sequents for the logic. There are probably also some connections that could now be made with work that has explicitly represented proof nets topologically as cell complexes (e.g. [25]).

§6. Complexity issues. While complexity of proof methods is not our focus here it must at least be mentioned, since the improvement in efficiency offered by compressed proof objects is a major reason for their promotion and study. The complexity of the decision problem in MLL has been shown to be NP-complete [20], so no theorem-proving scheme can ever really be tractable. The best that can be hoped for is minimal intractability. Proof nets have mostly been studied for the time complexity of the proof verification problem, and along these lines, the Danos contraction condition as described above has complexity $O(n^2)$ in the size of the proof structure [27]. Guerrini [18], however, showed how to convert the contraction algorithm into one with linear complexity. Another way of developing a linear time correctness check was shown by Murawski and Ong [29]. Given the existence of linear-time algorithms to check correctness of a proof structure, the origin of the overall NP complexity is therefore the proof *construction* due to the sheer number of possible proof structures to be checked, because how to create axiom links can be indeterminate after expanding the formula tree.

Turning to the matrix method, the way to check correctness of a matrix involves traversing all paths through it to see whether there exists a (linearly) spanning set of connections for it. Now, while a matrix appears to be a proof object of a truly minimal size and graphic intricacy, the worst-case complexity of checking a matrix would seem to be exponential, on the order of $2^n$ in the length of the formula. This can happen in the case of a formula that involves nested disjunctive subformulae, which will yield nested column matrices through which all paths must be traversed. For this reason, the matrix method was dismissed out of hand by Hughes [19] as not even a "proof system," which has occasionally been defined [6] as a system in which proofs can at least be verified, if not constructed, in polynomial time. The matrix method does bear the singular feature that actually constructing the proof object is deterministic and linear-time. But, to borrow a common adage, if logic were that easy everyone would be doing it. The matrix method's powerfully simple proof construction leaves a large debt to be paid on the other end of the deal, when the

time comes to check it. So in rough terms, matrices are easy to build but potentially hard to verify (not unlike the case of truth tables),[4] while the opposite is true for proof nets. In practical applications, of course, all these considerations are less important than the ultimate competition among the average-case complexities, and discussing that is beyond our scope here.

§7. **Identity of proofs.** The *identity of proofs* problem remains a significant open research question in logic and proof theory [10].[5] Simply put, for any given logical system this is the question of when two apparently different proofs (of the same formula) ought to be regarded as fundamentally "identical." While at first glance this issue appears tangential to the main track of the present paper, it has to be addressed because it has so often been a central concern in the community researching proof nets for the system MLL, among others. Proof nets have usually been promoted as addressing this question directly [16], and have sometimes been claimed to actually solve the issue [37]. It will now be explained how such claims should be viewed as exaggerated.

There may in general be more than one proof net for a provable sequent, however there can often be fewer possible proof nets than possible sequent proofs, even in a Cut-free system. Each proof net has often been viewed as representing an equivalence class of sequent proofs modulo "spurious ambiguity," while distinct proof nets will sequentialize respectively to full sequent proofs which are "nontrivially" distinct [37]. In lecture notes (op. cit.), Straßburger goes so far as to claim a theorem stating that two sequent calculus proofs in MLL translate to the same proof net iff they can be transformed into each other using only "trivial rule permutations." Yet, such a theorem seems to be circular, for in order to have this result one must assert in advance precisely what kinds of sequent rule permutations are held to be trivial and which are nontrivial. But it is this last issue that remains fundamentally a matter of debate!

Moreover (as Došen pointed out to me), on Straßburger's analysis, two sequent proofs which differ only by the presence of a useless Cut rule must be held to be nontrivially distinct, because the one with Cut will translate to a proof net involving a Cut link. Yet there is broad agreement among logicians that a sequent proof involving Cut should be regarded as "identical" to its Cut-free variant. Proof nets, therefore, should not be seen to have solved the identity of proofs question for any logical system. As for the matrix of a sequent, there can be only one, so as a

---

[4]Thanks to the referee for pointing out this similarity.

[5]This section owes a great debt to personal communications with Kosta Došen, and I herein communicate some of his arguments.

proof-theoretical object it does not address the fundamental question of "identity of proofs" other than to trivialize it.

§8. Concluding remarks. Past literature has rarely, if ever, connected all the topics and treatments touched on in the present paper. It is in the spirit of a new synthesis that the paper is offered, with the hope of a more complete understanding. We observe many connections between efforts to compress proof schemata, where the geometrical requirements of the compressed proof object arise out of the substructural nature of the logic. We also observe the connection between signed subformulae, which arise out of the concept of negation and the duality therefrom, and the "link" or "connection" notions which are central to the compressed proof objects, whether matrix or proof net.

REFERENCES

[1] V. Michele Abrusci, *Phase semantics and sequent calculus for pure noncommutative classical linear propositional logic*, **The Journal of Symbolic Logic**, vol. 56 (1991), no. 4, pp. 1403–1451.

[2] ———, *Lambek calculus, cyclic linear logic, noncommutative linear logic: language and sequent calculus*, **Proofs and linguistic categories** (V. Michele Abrusci and Claudia Casadio, editors), Cooperativa Libraria Universitaria Editrice Bologna, 1997, pp. 21–48.

[3] Peter B. Andrews, *Theorem proving via general matings*, **Journal of the ACM**, vol. 28 (1981), pp. 193–214.

[4] Evert W. Beth, **The foundations of mathematics**, North-Holland, Amsterdam, 1959.

[5] Wolfgang Bibel, *A theoretical basis for the systematic proof method*, **Mathematical foundations of computer science 1980** (Berlin) (P. Dembiński, editor), Lecture Notes in Computer Science, vol. 88, 9th Symposium, Springer-Verlag, 1980, pp. 154–167.

[6] Stephen A. Cook and Robert A. Reckhow, *The relative efficiency of propositional proof systems*, **The Journal of Symbolic Logic**, vol. 44 (1979), no. 1, pp. 36–50.

[7] Vincent Danos, *La logique linéaire appliquée à l'étude de divers processus de normalisation (principalement du λ-calcul)*, **Ph.D. thesis**, University of Paris VII, 1990.

[8] Vincent Danos and Laurent Regnier, *The structure of multiplicatives*, **Archive for Mathematical Logic**, vol. 28 (1989), pp. 181–203.

[9] Philippe de Groote and François Lamarche, *Classical non-associative Lambek calculus*, **Studia Logica**, vol. 71 (2002), no. 3, pp. 355–388.

[10] Kosta Došen, *Identity of proofs based on normalization and generality*, **Bulletin of Symbolic Logic**, vol. 9 (2003), no. 4, pp. 477–503.

[11] Roy Dyckhoff and Luís Pinto, *Permutability of proofs in intuitionistic sequent calculi*, **Theoretical Computer Science**, vol. 212 (1999), no. 1–2, pp. 141–155.

[12] Sean A. Fulop, **On the logic and learning of language**, Trafford, Victoria, British Columbia, 2004.

[13] Dov Gabbay and Nicola Olivetti, *Goal-directed proof theory*, Kluwer, Dordrecht, 2000.

[14] D. Galmiche, *Connection methods in linear logic and proof nets construction*, **Theoretical Computer Science**, vol. 232 (2000), pp. 231–272.

[15] Gerhard Gentzen, *Untersuchungen über das logische Schliessen*, **Math. Zeitschrift**, vol. 39 (1934), pp. 176–210, 405–431, English translation in [38].

[16] Jean-Yves Girard, *Linear logic*, **Theoretical Computer Science**, vol. 50 (1987), pp. 1–102.

[17] Jonathan L. Gross and Thomas W. Tucker, *Topological graph theory*, Wiley, New York, 1987.

[18] Stefano Guerrini, *Correctness of multiplicative proof nets is linear*, **14th IEEE symposium on logic in computer science**, 1999, pp. 454–463.

[19] Dominic J. D. Hughes, *Proofs without syntax*, **The Annals of Mathematics**, vol. 164 (2006), no. 3, pp. 1065–1076.

[20] Max Kanovich, *The multiplicative fragment of linear logic is NP-complete*, Preprint, University of Amsterdam, 1991.

[21] G. M. Kelly and S. Mac Lane, *Coherence in closed categories*, **J. Pure and Applied Algebra**, vol. 1 (1971), no. 1, 2, pp. 97–140; 219.

[22] François Lamarche and Christian Retoré, *Proof nets for the Lambek calculus — an overview*, **Proceedings of the 3rd Roma workshop: Proofs and linguistic categories**, 1996, pp. 241–262.

[23] Joachim Lambek, *The mathematics of sentence structure*, **American Mathematical Monthly**, vol. 65 (1958), pp. 154–170.

[24] ———, *On the calculus of syntactic types*, **Structure of language and its mathematical aspects** (Providence, RI) (Roman Jakobson, editor), Proceedings of Symposia in Applied Mathematics, American Mathematical Society, 1961, pp. 166–178.

[25] François Métayer, *Implicit exchange in multiplicative proofnets*, **Mathematical Structures in Computer Science**, vol. 11 (2001), pp. 261–272.

[26] Michael Moortgat, *Categorial type logics*, **Handbook of logic and language** (Johan van Benthem and Alice ter Meulen, editors), Elsevier, 1997.

[27] Richard Moot, *Proof nets for linguistic analysis*, **Ph.D. thesis**, Utrecht University, 2002.

[28] Glyn V. Morrill, **Type logical grammar: Categorial logic of signs**, Kluwer, Dordrecht, 1994.

[29] A. S. Murawski and C.-H. L. Ong, *Dominator trees and fast verification of proof nets*, **15th annual IEEE symposium on logic in computer science**, 2000, pp. 181–191.

[30] Mati Pentus, *Equivalence of multiplicative fragments of cyclic linear logic and noncommutative linear logic*, **Logical foundations of computer science, proc. 4th internat. symp.** (S. Adian and A. Nerode, editors), LNCS, vol. 1234, Springer, Berlin, 1997, pp. 306–311.

[31] Greg Restall, **An introduction to substructural logics**, Routledge, London, 2000.

[32] Edmund Robinson, *Proof nets for classical logic*, **Journal of Logic and Computation**, vol. 13 (2003), no. 5, pp. 777–797.

[33] Dirk Roorda, *Resource logics: Proof-theoretical investigations*, **Ph.D. thesis**, Universiteit van Amsterdam, 1991.

[34] ———, *Proof nets for Lambek calculus*, **Journal of Logic and Computation**, vol. 2 (1992), no. 2, pp. 211–231.

[35] Raymond M. Smullyan, *A unifying principle in quantification theory*, **Proc. Nat. Academy of Sciences**, vol. 49 (1963), no. 6, pp. 828–832.

[36] ——, **First order logic**, Springer-Verlag, Berlin, 1968.

[37] Lutz Strassburger, *Proof nets and the identity of proofs*, **Technical Report 6013**, INRIA, Cedex, France, 2006.

[38] M. E. Szabo (editor), **The collected papers of Gerhard Gentzen**, North-Holland, Amsterdam, 1969.

[39] Lincoln A. Wallen, **Automated proof search in non-classical logics: Efficient matrix proof methods for modal and intuitionistic logics**, The MIT Press, Cambridge, MA, 1990.

DEPT. OF LINGUISTICS
CALIFORNIA STATE UNIVERSITY, FRESNO
5245 N. BACKER AVE. PB92
FRESNO, CA 93740-8001
*E-mail*: sfulop@csufresno.edu