ON LINEAR REWRITING SYSTEMS FOR BOOLEAN LOGIC AND SOME APPLICATIONS TO PROOF THEORY

ANUPAM DAS AND LUTZ STRASSBURGER

LIP, Université de Lyon, CNRS, ENS de Lyon, Université Claude-Bernard Lyon 1, Milyon
e-mail address: anupam.das@ens-lyon.fr

Inria, LIX
e-mail address: lutz@lix.polytechnique.fr

Abstract. Linear rules have played an increasing role in structural proof theory in recent years. It has been observed that the set of all sound linear inference rules in Boolean logic is already **coNP**-complete, i.e. that every Boolean tautology can be written as a (left- and right-)linear rewrite rule. In this paper we study properties of systems consisting only of linear inferences. Our main result is that the length of any 'nontrivial' derivation in such a system is bound by a polynomial. As a consequence there is no polynomial-time decidable sound and complete system of linear inferences, unless **coNP** = **NP**. We draw tools and concepts from term rewriting, Boolean function theory and graph theory in order to access some required intermediate results. At the same time we make several connections between these areas that, to our knowledge, have not yet been presented and constitute a rich theoretical framework for reasoning about linear TRSs for Boolean logic.

1. Introduction

Consider the following conjunction rule from a Gentzen-style sequent calculus:

$$\frac{\Gamma, A \qquad B, \Delta}{\Gamma, A \wedge B, \Delta} \tag{1.1}$$

where Γ and Δ are finite sequences of formulae. In this rule all the formulae in the premisses occur in the conclusion with the *same multiplicity*. In proof theory this is referred to as a *multiplicative* rule. This phenomenon can also be described as a *linear* rule in term rewriting. For instance, the proof rule above has logical behaviour induced by the following linear term rewriting rule,

$$(C \vee A) \wedge (B \vee D) \quad \to \quad C \vee (A \wedge B) \vee D \tag{1.2}$$

where C and D here represent the disjunction of the formulae in Γ and Δ respectively from (1.1).

This rule has been particularly important in structural proof theory, serving as the basis of Girard's *multiplicative linear logic* [Gir87]. A variant of (1.2), that will play some role in this paper is the following:

$$\mathsf{s} \quad : \quad A \wedge (B \vee C) \quad \to \quad (A \wedge B) \vee C \tag{1.3}$$

that we call here *switch*, following [Gug07, GS01, BT01], but that is also known as *weak distributivity* [BCST96].

This is an extended version of [DS15] which appeared in the proceedings of *RTA 2015*.

However the concept of linearity, or 'multiplicativity', itself is far more general. For instance, the advent of *deep inference* has introduced the following linear rule known as *medial* [BT01]:

$$\mathsf{m} \quad : \quad (A \wedge B) \vee (C \vee D) \quad \to \quad (A \vee C) \wedge (B \vee D) \tag{1.4}$$

This rule cannot be derived from (1.2), (1.3) or related rules, even when working modulo logical equivalence and logical constants. From the point of view of proof theory (1.4) is particularly interesting since it allows for *contraction*,

$$\mathsf{c} \quad : \quad A \vee A \quad \to \quad A \tag{1.5}$$

to be reduced to atomic form. For example consider the following transformation which reduces the logical complexity of a contraction step,

$$\begin{array}{ll} & \underline{(A \wedge B) \vee (A \wedge B)} \\ \underset{\mathsf{c}}{\to} & A \wedge B \end{array} \quad \rightsquigarrow \quad \begin{array}{ll} & \underline{(A \wedge B) \vee (A \wedge B)} \\ \underset{\mathsf{m}}{\to} & \underline{(A \vee A)} \wedge (B \vee B) \\ \underset{\mathsf{c}}{\to} & A \wedge \underline{(B \vee B)} \\ \underset{\mathsf{c}}{\to} & A \wedge B \end{array} \tag{1.6}$$

where redexes are underlined.

Until now the nature of linearity in Boolean logic has not been well understood, despite proving to be a concept of continuing interest in proof theory, cf. [Gug11], and category theory, cf. [Str07b, Lam07]. While switch and medial form the basis of usual deep inference systems, it has been known for some time that other options are available: there are linear rules that cannot be derived from just these two rules (even modulo logical equivalences and constants), first explicitly shown in [Str12]. The minimal known example, from [Das13], is the following:

$$\begin{array}{ll} & (u \vee (v \wedge v')) \wedge ((w \wedge w') \vee (x \wedge x')) \wedge ((y \wedge y') \vee z) \\ \to & (u \wedge (w \vee y)) \vee (w' \wedge y') \vee (v' \wedge x') \vee ((v \vee x) \wedge z) \end{array} \tag{1.7}$$

This example can be generalised to an infinite set of rules, where each rule is independent from all smaller rules. In fact, the situation is rather more intricate than this: the set of linear inferences, denoted L henceforth, is itself **coNP**-complete [Str12]. This means that *every* Boolean tautology can be written as a linear rule. This leads us to a natural question:

Question 1.1. *Can we find a complete 'basis' of linear inference rules?*

In other words, can proof theory itself be conducted in an entirely linear setting? Such an approach would be in stark contrast with the traditional approach of *structural* proof theory, which precisely emphasises the role of nonlinear behaviour via the structural rules, e.g. contraction and weakening. Clearly, this would also be advantageous from the point of view of space complexity in proofs.

The main result of this work is a negative answer to the above question: there is no complete polynomial-time decidable linear TRS that is complete for L, unless **coNP** = **NP**. Notice that the polynomial-time decidable criterion is essentially the most general condition one can impose without admitting a trivially positive answer to Question 1.1 (e.g. by allowing the basis to be L itself). It is also a natural condition arising from proof theory, via the Cook-Reckhow definition of an abstract proof system [CR74].

The high-level argument is as follows:

(1) Any linear derivation of 'nontrivial' linear inferences must have polynomial length.

(2) If a linear system is complete for L then arbitrary linear inferences can be derived from the 'nontrivial' fragment of L with derivations of polynomial length.
(3) Putting these together, a complete linear system must admit polynomial-size derivations for any linear inference, inducing a **NP** algorithm for L, and so **coNP** = **NP**.

Point (1) above represents the major technical contribution of this work. The proof requires us to work in three different settings: term rewriting, Boolean function theory and graph theory. Many of our intermediate results require elegant and novel interplays between these settings, taking advantage of their respective invariants, and this should be evident from our exposition. (2) essentially appeared before in [Das13]. We point out that the important point here is the *existence* of small derivations, rather than the ability to explicitly construct them efficiently.

Functions computed by linear terms of Boolean logic have been studied in Boolean function theory and circuit complexity for decades, where they are called "read-once functions" (e.g. in [CH11]).[1] They are closely related to positional games (first mentioned in [Gur82]) and have been used in amplification of approximation circuits, (first in [Val84], more generally in [DZ97]) as well as other areas. However, despite this, it seems that there has been little study on *logical* relationships between read-once functions, e.g. when one implies another. Many of the basic results and correspondences in this work, e.g. Proposition 4.4 and Theorem 4.6, have not appeared before, as far as we know, and themselves constitute interesting points of study.

This is a full version of the extended abstract [DS15], which was presented at the *RTA 2015* conference. In addition to providing full proofs for the various results, this version generally elaborates on many of the discussions in the previous version and gives a proof-theoretic context to this line of work. To this end we have included some further developments in Sections 7, 8 and 9 which are derived from our main result.

The structure of the paper is as follows. In Sections 2, 3 and 4 we present preliminaries on each of our three settings and some basic results connecting various concepts between them. In Section 5 and 6 we specialise to the setting of linear rewrite rules for Boolean logic and present our main results, Theorem 5.8 and Corollary 6.10. In Sections 7 and 8 we present some applications to deep inference proof theory, showing a form of *canonicity* for medial and some general consequences for the normalisation of deep inference proofs. In Section 9 we discuss a direction for future work in a graph-theoretic setting, and in Section 10 we present some concluding remarks, including relationships to models of linear logic and axiomatisations of Boolean algebras.

Acknowledgements. We would like to thank Paola Bruscoli, Kaustuv Chaudhuri, Alessio Guglielmi, Willem Heijltjes and others in the deep inference community for many fruitful discussions on these topics, as well as the anonymous referees for previous versions of this work for their useful comments.

[1] These have been studied in various forms and under different names. The first appearance we are aware of is in [Che67], and also the seminal paper of [Gur77] characterising these functions. The book we reference presents an excellent and comprehensive introduction to the area.

2. Preliminaries on rewriting theory

We work in the setting of first-order term rewriting as defined in the Terese textbook, *Term Rewriting Systems* [Ter03]. We will use the same notation for all symbols except the connectives, for which we use more standard notation from proof theory. In particular we will use $\bot$ and $\top$ for the truth constants, reserving 0 and 1 for the inputs and outputs of Boolean functions, introduced later.

We adopt two particular conventions which differ from usual definitions in the literature:

(1) A term rewriting system (TRS) is usually defined as an arbitrary set of rewrite rules. Here we insist that the set of instances of these rules, or reduction steps, is polynomial-time decidable.
(2) Rewriting modulo an equivalence relation usually places no restriction on the source and target of a reduction step. Here we insist that they must be *distinct* modulo the equivalence relation.

The motivation for (1) is that we wish to be as general as possible without admitting trivial results. If we allowed all sets then a complete system could be specified quite easily indeed. Furthermore, that an inference rule is easily or feasibly checkable is a usual requirement in proof theory, and in proof complexity this is formalised by the same condition (1) on inference rules [CR74].

The motivation for (2) is that we fundamentally care about weak normalisation, e.g. Theorem 6.9, but it will be useful to make statements resembling strong normalisation under this notion of rewriting modulo, e.g. Theorem 5.8. All equivalence relations that we consider in this work are polynomial-time decidable, and so this convention is consistent with 1. The same notion of rewriting modulo was also used in previous work [Das13].

Let us now consider Boolean logic in the term rewriting setting. Our language is built from the connectives $\bot, \top, \wedge, \vee$ and a set *Var* of propositional variables, typically denoted x, y, z etc. The set *Var* is equipped with an involution (i.e. self-inverse function) $\bar{\cdot} : \mathit{Var} \to \mathit{Var}$, such that $\bar{x} \neq x$ for all $x \in \mathit{Var}$. We call $\bar{x}$ the *dual* of x and, for each pair of dual variables, we arbitrarily choose one to be *positive* and the other to be *negative*.

The set *Ter* of formulae, or *terms*, is built freely from this signature in the usual way. Terms are typically denoted by s, t, u etc., and term and variable symbols may occur with superscripts and subscripts if required.

In this setting $\top$ and $\bot$ are considered the constant symbols of our language. We say that a term t is *constant-free* if $\top$ and $\bot$ do not occur in t.

We do not include a symbol for negation in our language. This is due to the fact that soundness of a rewrite step is only preserved under *positive* contexts. Instead we simply consider terms in negation normal form (NNF), which can be generated for arbitrary terms from positive and negative variables by the De Morgan laws:

$$\bar{\top} = \bot \qquad \bar{\bot} = \top \qquad \bar{\bar{x}} = x \qquad \overline{s \vee t} = \bar{s} \wedge \bar{t} \qquad \overline{s \wedge t} = \bar{s} \vee \bar{t}$$

We say that a term is *negation-free* if it does not contain any negative variables. We write $\mathit{Var}(t)$ to denote the set of variables occurring in t. We say that a term t is *linear* if, for each $x \in \mathit{Var}(t)$, there is exactly one occurrence of x in t. The *size* of a term t, denoted $|t|$, is the total number of variable and function symbols occurring in t. A *substitution* is a mapping $\sigma \colon \mathit{Var} \to \mathit{Ter}$ from the set of variables to the set of terms such that $\sigma(x) \neq x$ for only finitely many x. The notion of substitution is extended to all terms, i.e. a map

$Ter \to Ter$, in the usual way. A (one-hole) *context* is a term with a single 'hole' $\square$ occurring in place of a subterm. Below are three examples:

$$C_1[\square] := y \wedge (z \vee \square) \qquad C_2[\square] := \square \vee (w \wedge x) \qquad C_3[\square] := (w \wedge x) \vee (y \wedge (z \vee \square)) \tag{2.1}$$

We may write $C_i[t]$ to denote the term obtained by replacing the occurrence of $\square$ in $C_i[\square]$ with t. We may also replace holes with other contexts to derive new contexts. For example, notice that $C_3[\square]$ in (2.1) is equivalent, modulo commutativity of $\vee$, to $C_2[C_1[\square]]$.

Definition 2.1 (Rewrite rules)**.** A *rewrite rule* is an expression $l \to r$, where l and r are terms, such that $l \neq r$. We write $\rho : l \to r$ to express that the rule $l \to r$ is called ρ. In this rule we call l the *left hand side (LHS)* of ρ, and r its *right hand side (RHS)*. We say that ρ is *left-linear* (resp. *right-linear*) if l (resp. r) is a linear term. We say that ρ is *linear* if it is both left- and right-linear. We write $s \underset{\rho}{\to} t$ to express that $s \to t$ is a *reduction step* of ρ, i.e. that $s = C[\sigma(l)]$ and $t = C[\sigma(r)]$ for some substitution σ and some context $C[\square]$.

Definition 2.2 (Term rewriting systems)**.** A *term rewriting system* (TRS) is a set of rewrite rules whose reduction steps are decidable in polynomial time. The *one-step* reduction relation of a TRS R is $\underset{R}{\to}$, where $s \underset{R}{\to} t$ if $s \underset{\rho}{\to} t$ for some $\rho \in R$. A *linear (term rewriting) system* is a TRS whose rules are all linear.

Definition 2.3 (Derivations)**.** A *derivation* under a binary relation $\underset{R}{\to}$ on Ter is a finite sequence $\pi : t_0 \underset{R}{\to} t_1 \underset{R}{\to} \cdots \underset{R}{\to} t_l$. In this case we say that π has *length* l. We also write $\overset{*}{\underset{R}{\to}}$ to denote the reflexive transitive closure of $\underset{R}{\to}$.

Definition 2.4 (Rewriting modulo)**.** For an equivalence relation $\sim$ on Ter and a TRS R, we define the relation $\underset{R/\sim}{\longrightarrow}$ as follows:

$$s \underset{R/\sim}{\longrightarrow} t \quad \text{iff} \quad \text{there are } s' \text{ and } t' \text{ such that } s \sim s' \underset{R}{\to} t' \sim t \text{ and } s' \nsim t' \quad .$$

An $R/\sim$ derivation is also called an *R-derivation modulo* $\sim$.

We write AC to denote the smallest congruence relation generated by the following four equations for associativity and commutivity of $\wedge$ and $\vee$:

$$(x \wedge y) \wedge z = x \wedge (y \wedge z) \qquad (x \vee y) \vee z = x \vee (y \vee z) \qquad x \wedge y = y \wedge x \qquad x \vee y = y \vee x$$

Note that AC contains only linear equations. The following set of equations for the truth constants, that we call U, also contains only linear equations:

$$x \vee \bot = x = \bot \vee x \qquad x \wedge \top = x = \top \wedge x \qquad \top \vee \top = \top \qquad \bot \wedge \bot = \bot$$

We denote by ACU the combined system of AC and U. We will also need the system U' that extends U by the following rules:

$$x \vee \top = \top = \top \vee x \qquad x \wedge \bot = \bot = \bot \wedge x$$

Notice that these are not linear in the sense of [Das13], but are considered linear in our more general setting. We denote by ACU' the combined system of AC and U'.

It turns out that this equivalence relation relates precisely those linear terms that compute the same Boolean function, as we will see later.

3. Preliminaries on relation webs

In this section we restrict our attention to negation-free constant-free linear terms and study their syntactic structure, in the form of *relation webs* [Gug07, Str07a].

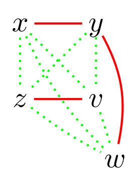
We make use of *labelled graphs* with their standard terminology. For a graph G we denote its *vertex set* or set of *nodes* as $V(G)$, and the set of its *labelled edges* as $E(G)$. We say "$x \stackrel{\star}{\text{———}} y$ in G" to express that the edge $\{x, y\}$ is labelled $\star$ in the graph G. A set $X \subseteq V(G)$ is a $\star$-*clique* if every pair $x, y \in X$ has a $\star$-labelled edge between them. A *maximal* $\star$-clique is a $\star$-clique that is not contained in any larger $\star$-clique.

Analysing the term tree of a negation-free constant-free linear term t, notice that for each pair of variables x, y occurring in t, there is a unique connective $\star \in \{\wedge, \vee\}$ at the root of the smallest subtree containing the (unique) occurrences of x and y. Let us call this the *first common connective* of x and y in t.

Definition 3.1 (Relation webs)**.** The *(relation) web* $\mathcal{W}(t)$ of a constant-free negation-free linear term t is the complete graph whose vertex set is $Var(t)$, such that the edge between two variables x and y is labelled by their first common connective in t. We write $e_\wedge(t)$ (resp. $e_\vee(t)$) to be the number of $\wedge$- (resp. $\vee$-)labelled edges in $\mathcal{W}(t)$.

As a convention we will write $x \text{———} y$ if the edge $\{x, y\}$ is labelled by $\wedge$, and we write $x \cdots\cdots y$ if it is labelled by $\vee$.

Example 3.2. The term $t = ((x \vee w) \wedge y) \vee (z \wedge v)$ has the relation web:



We have that $e_\wedge(t) = 3$ and $e_\vee(t) = 7$.

Proposition 3.3. *Let t be a constant-free negation-free linear term with n variables, and let $e := \frac{1}{2}n(n-1)$. Then $e_\wedge(t), e_\vee(t) \leq e$, and $e_\wedge(t) + e_\vee(t) = e$.*

Proof. This follows from the fact that there are only e edges in a web, all of which must be labelled $\wedge$ or $\vee$. □

Remark 3.4 (Labels)**.** We point out that, instead of using labelled complete graphs, we could have also used unlabelled arbitrary graphs, since we have only two connectives ($\wedge$ and $\vee$) and so one could be specified by the lack of an edge. This is indeed done in some settings, e.g. the cooccurrence graphs of [CH11]. However, we use the current formulation in order to maintain consistency with the previous literature, e.g. [Gug07] and [Str07a], and since it helps write certain arguments, e.g. in Section 7, where we need to draw graphs with incomplete information.

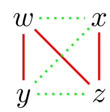
One of the reasons for considering relation webs is the following proposition, which allows to reason about equivalence classes modulo AC easily.

Proposition 3.5. *Constant-free negation-free linear terms are equivalent modulo AC if and only if they have the same web.*

Proof. This follows immediately from the definition and that AC preserves first common connectives. □

An important property of webs is that they have no minimal paths of length > 2. More precisely, we have the following:

Proposition 3.6. *A complete $\{\wedge, \vee\}$-labelled graph on X is the web of some negation-free constant-free linear term on X if and only if it contains no induced subgraphs of the form:*



(3.1)

A proof of this property can be found, for example, in [Möh89], [Ret93], [BdGR97], or [Gug07]. It is called *P_4-freeness* or *Z-freeness* or *N-freeness*, depending on the viewpoint. We will make crucial use of it when later reasoning with webs.

4. Preliminaries on Boolean functions

In this section we introduce the usual Boolean function models for terms of Boolean logic. A *Boolean function* on a (finite) set of variables $X \subseteq \mathit{Var}$ is a map $f\colon \{0,1\}^X \to \{0,1\}$. We identify $\{0,1\}^X$ with $\mathcal{P}(X)$, the powerset of X, i.e. we may specify an argument of a Boolean function by the subset of its variables assigned to 1.

A little more formally, a function $\nu : X \to \{0,1\}$ is specified by the set X_ν it indicates, i.e. $x \in X_\nu$ just if $\nu(x) = 1$. For this reason we may quantify over the arguments of a Boolean function by writing $Y \subseteq X$ rather than $\nu \in \{0,1\}^X$, i.e. we write $f(Y)$ to denote the value of f if the input is 1 for the variables in Y and 0 for the variables in $X \setminus Y$. Similarly, we write $f(\overline{Y})$ for the value of f when the variables in Y are 0 and the variables in $X \setminus Y$ are 1.

For Boolean functions $f, g : \{0,1\}^X \to \{0,1\}$ we write $f \leq g$ if, for every $Y \subseteq X$, we have that $f(Y) \leq g(Y)$. Notice that the following can easily be shown to be equivalent:

(1) $f \leq g$.
(2) $f(Y) = 1 \Rightarrow g(Y) = 1$.
(3) $g(Y) = 0 \Rightarrow f(Y) = 0$.

We also write $f < g$ if $f \leq g$ but $f(Y) \neq g(Y)$ for some $Y \subseteq X$.

Definition 4.1. A Boolean function $f\colon \{0,1\}^X \to \{0,1\}$ is *monotone* iff $Y \subseteq Y' \subseteq X$ implies $f(Y) \leq f(Y')$.

Definition 4.2. Let f be a monotone Boolean function on a variable set X. A set $Y \subseteq X$ is a *minterm* (resp. *maxterm*) for f if it is a minimal set such that $f(Y) = 1$ (resp. $f(\overline{Y}) = 0$). The set of all minterms (resp. maxterms) of f is denoted $\mathit{MIN}(f)$ (resp. $\mathit{MAX}(f)$).

Observation 4.3. Monotone Boolean functions are uniquely determined by their minterms or by their maxterms. In particular, for two function f and g, we have $\mathit{MIN}(f) \neq \mathit{MIN}(g)$ iff $\mathit{MAX}(f) \neq \mathit{MAX}(g)$ iff there is a Y such that $f(Y) \neq g(Y)$.

Minterms and maxterms correspond to minimal DNF and CNF representations, respectively, of a monotone Boolean function. We refer the reader to [CH11] for an introduction to their theory. In this work we use them in a somewhat different way to Boolean function theory, in that we devise definitions of logical concepts such as entailment and, in the next section, what we call "triviality". The reason for this is to take advantage of the purely function-theoretic results stated in this section (e.g. Gurvich's Theorem 4.10 below) to derive our main results in Sections 5 and 6.

Proposition 4.4. *For monotone Boolean functions f, g on the same variable set, the following are equivalent:*

(1) $f \leq g$.
(2) $\forall S \in \mathit{MIN}(f).\ \exists S' \in \mathit{MIN}(g).\ S' \subseteq S$.
(3) $\forall T \in \mathit{MAX}(g).\ \exists T' \in \mathit{MAX}(f).\ T' \subseteq T$.

Proof. $1 \implies 2$. Let $f \leq g$ and suppose there is an $S \in \mathit{MIN}(f)$ such that there is no $S' \in \mathit{MIN}(g)$ with $S' \subseteq S$. Then $f(S) = 1$ and $g(S) = 0$, contradicting $f \leq g$.

$2 \implies 1$. Let Y be such that $f(Y) = 1$. Then there is a minterm $S \in \mathit{MIN}(f)$ with $S \subseteq Y$. By 2, there is a minterm $S' \in \mathit{MIN}(g)$ with $S' \subseteq S$, and therefore $S' \subseteq Y$. Therefore $g(Y) = 1$, by monotonicity, and so $f \leq g$.

$1 \implies 3$ and $3 \implies 1$ are proved similarly. □

A term t computes a Boolean function $\{0,1\}^{\mathit{Var}(t)} \to \{0,1\}$, in the usual way, and negation-free terms can easily be seen to compute monotone Boolean functions. Thus, we can speak of minterms and maxterms of a negation-free term t, referring to the minterms and maxterms of the function computed by t. This allows us to give a graph-theoretic formulation of minterms and maxterms using concepts from the previous section. We give the following alternative definition of minterms and maxterms:

Proposition 4.5 (Inductive definition of minterms and maxterms)**.** *Let t be a linear term. A set $S \subseteq \mathit{Var}(t)$ is a minterm of t if and only if:*

- $t = x$ *and* $S = \{x\}$.
- $t = t_1 \vee t_2$ *and* S *is a minterm of* t_1 *or of* t_2.
- $t = t_1 \wedge t_2$ *and* $S = S_1 \cup S_2$ *where each* S_i *is a minterm of* t_i.

Dually, a set $T \subseteq \mathit{Var}(t)$ is a maxterm of t if and only if:

- $t = x$ *and* $T = \{x\}$.
- $t = t_1 \vee t_2$ *and* $T = T_1 \cup T_2$ *where each* T_i *is a maxterm of* t_i.
- $t = t_1 \wedge t_2$ *and* T *is a maxterm of* t_1 *or of* t_2.

Proof. This follows straightforwardly from Definition 4.2 structural induction on t. □

We can now characterize minterms and maxterms as maximal cliques in relation webs:

Theorem 4.6. *A set of variables is a minterm (resp. maxterm) of a negation-free constant-free linear term t if and only if it is a maximal $\wedge$-clique (resp. maximal $\vee$-clique) in $\mathcal{W}(t)$.*

Proof. This follows from structural induction on t and Proposition 4.5. □

Definition 4.7 (Read-once functions)**.** A Boolean function is called *read-once* if it is computed by some linear term (of propositional logic).

It is not exactly clear when the following result first appeared, although we refer to a discussion in [CH11] where it is stated that results directly implying this were first mentioned in [Kuz58]. The result also occurs in [Gur77], and is generalised to certain other bases in [HNW94] and [HK90].

Theorem 4.8. *Constant-free negation-free linear terms compute the same (read-once) Boolean function if and only if they are equivalent modulo AC.*

Proof. This follows immediately from Proposition 3.5, Theorem 4.6, and Observation 4.3. □

The following consequence of Theorem 4.8 appears in [Das11], where a detailed proof may be found.

Corollary 4.9. *Negation-free linear terms compute the same (read-once) Boolean function if and only if they are equivalent modulo ACU'.*

Proof idea. The result essentially follows from the observation that every negation-free term is ACU'-equivalent to $\bot$, $\top$ or a unique constant-free term. □

Let us end this section with the following classical result, characterising the read-once functions over $\wedge$ and $\vee$, due to Gurvich in [Gur77]. This has appeared in various presentations and, in particular, the proof appearing in [CH11] uses 'cooccurrence' graphs that correspond to our relation webs.

Theorem 4.10 (Gurvich)**.** *A monotone Boolean function f is read-once if and only if*

$$\forall S \in MIN(f).\, \forall T \in MAX(f).\, |S \cap T| = 1 \quad .$$

In this paper we will actually only need one direction of this theorem: that for monotone read-once functions, minterms and maxterms have singleton intersection. Using the different settings we have introduced, we arrive at a remarkably simple proof of this direction:

Proof of left-right direction of Theorem 4.10. A minterm and maxterm of f must intersect since, otherwise, we could simultaneously force f to evaluate to 0 and 1. On the other hand, by Theorem 4.6, a minterm is a $\wedge$-maxclique of $\mathcal{W}(t)$ and a maxterm is a $\vee$-maxclique of $\mathcal{W}(t)$, and cliques with different labels can intersect at most once. □

This simple proof exemplifies the usefulness of considering both the graph theoretic viewpoint and the Boolean function viewpoint. Such interplays will prove to be very useful in the remainder of this work.

5. Linear inferences, triviality and a polynomial bound on length

In the previous section we considered the semantics of linear terms via Boolean functions. In this section we study sound rewriting steps between linear terms and prove our main result about the length of such rewriting paths.

Definition 5.1 (Soundness)**.** We say that a rewrite rule $s \to t$ is *sound* if s and t compute Boolean functions f and g, respectively, such that $f \leq g$. We say that a TRS is sound if all its rules are sound. A *linear inference* is a sound linear rewrite rule.

Notation 5.2. To switch conveniently between the settings of terms and Boolean functions, we freely interchange notations, e.g. writing $s \leq t$ to denote that $s \to t$ is sound, and saying $f \to g$ is sound when $f \leq g$.

We immediately have the following, which can also be found in [Das13].

Proposition 5.3. *Any sound negation-free linear TRS, modulo ACU', is terminating in exponential-time.*

Proof. The result follows by Boolean semantics and Corollary 4.9: each consequent term must compute a distinct Boolean function that is strictly bigger, under $\leq$, and the graph of $\leq$ has length 2^n, where n is the number of variables in the input term. □

The purpose of this section is now to put a polynomial bound on the length of certain linear derivations. For this, the fundamental concept we use is that of "triviality", first introduced in [Das13] as "semantic triviality". The idea behind triviality of a variable in some linear inference is that the inference is "independent" of the behaviour of that variable.

Definition 5.4 (Triviality)**.** Let f and g be Boolean functions on a set of variables X, and let $x \in X$. We say $f \to g$ is *trivial* at x if for all $Y \subseteq X$, we have $f(Y \cup \{x\}) \leq g(Y \setminus \{x\})$. We say simply that $f \to g$ is *trivial* if it is trivial at one of its variables.

For example the following linear inference,

$$x \wedge y \quad \to \quad x \vee y$$

is trivial at x or y, whereas the linear inference,

$$x \wedge (y_1 \vee \cdots \vee y_n) \quad \to \quad x \vee (y_1 \wedge \cdots \vee y_n)$$

is trivial at all y_i simultaneously.

Remark 5.5 (Hereditariness of triviality)**.** Notice that the triviality property is somehow hereditary: if a sound sequence $f_0 \to f_1 \to \ldots \to f_l$ of Boolean functions is trivial at some point $f_i \to f_{i+1}$ for $0 \leq i < l$ then $f_1 \to f_n$ is trivial. However the converse does not hold: if the first and last function of a sound sequence constitutes a trivial pair it may be that there is no local triviality in the sequence. For example the endpoints of the derivation,

$$(w \wedge x) \vee (y \wedge z) \to [w \vee y] \wedge [x \vee z] \to w \vee x \vee (y \wedge z)$$

form a pair that is trivial at w (or trivial at x), but no local step witnesses this. In these cases we call the sequence *globally* trivial. This notion is fundamental later in Lemma 5.7, on which our main result crucially relies.

In a similar way to how we expressed soundness via minterms or maxterms in Proposition 4.4, we can also define triviality via minterms or maxterms.

Proposition 5.6. *The following are equivalent:*

(1) $f \to g$ *is trivial at* x.
(2) $\forall S \in \mathit{MIN}(f).\ \exists S' \in \mathit{MIN}(g).\ S' \subseteq S \setminus \{x\}$.
(3) $\forall T \in \mathit{MAX}(g).\ \exists T' \in \mathit{MAX}(f).\ T' \subseteq T \setminus \{x\}$.

Proof. We first show that $1 \implies 2$. Assume $f \to g$ is trivial at x, and let $S \in \mathit{MIN}(f)$. We have $f(S) = 1$, and hence also $f(S \cup \{x\}) = 1$. By way of contradiction assume there is no $S' \in \mathit{MIN}(g)$ with $S' \subseteq S \setminus \{x\}$. Therefore $g(S \setminus \{x\}) = 0$, contradicting triviality at x. Next, we show $2 \implies 1$. For this, let Y be such that $f(Y \cup \{x\}) = 1$. Then there is a minterm $S \in \mathit{MIN}(f)$ with $S \subseteq Y \cup \{x\}$. By 2, there is a minterm $S' \in \mathit{MIN}(g)$ with $S' \subseteq S \setminus \{x\}$. Hence $S' \subseteq Y \setminus \{x\}$. Therefore $g(Y \setminus \{x\}) = 1$, and thus $f \to g$ is trivial at x. To show $1 \implies 3$ and $3 \implies 1$ we proceed analogously. □

Let us now fix a sequence $f = f_0 < f_1 < \cdots < f_l = g$ of strictly increasing read-once Boolean functions on a variable set X. We are going to show that, unless $f \to g$ is trivial, for each variable $x \in X$ we must be able to associate a minterm S^x of f such that, for any $S \subseteq S^x$ that is a minterm of some f_i, it must be that $S \ni x$. This is visualized in Figure 1 together with the dual property for maxterms.

Lemma 5.7 (Subset and intersection lemma)**.** *Suppose* $f \to g$ *is not trivial. For every variable* $x \in X$*, there is a minterm* S^x *of* f *and a maxterm* T^x *of* g *such that:*

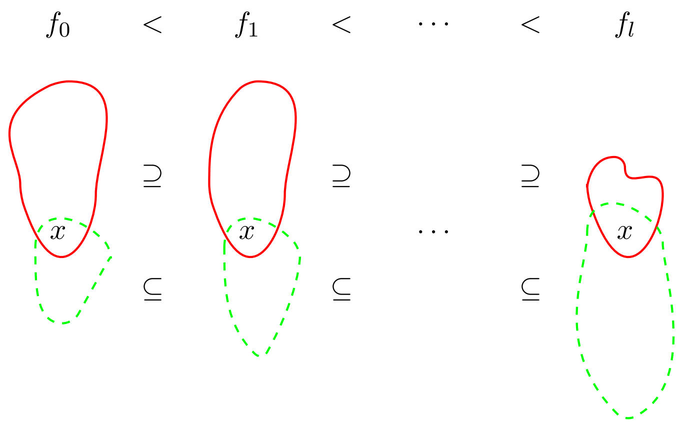


Figure 1: The critical minterms and maxterms of a sound sequence, cf. Lemma 5.7.

(1) $\forall S_i \in MIN(f_i).S_i \subseteq S^x \implies x \in S_i$.
(2) $\forall T_i \in MAX(g_i).T_i \subseteq T^x \implies x \in T_i$.
(3) $\forall S_i \in MIN(f_i), \forall T_i \in MAX(g_i).S_i \subseteq S^x, T_i \subseteq T^x \implies S_i \cap T_i = \{x\}$.

Proof. Suppose that, for some variable x no minterm of f has property 1. In other words, for every minterm S^x of f containing x there is some minterm S_i of some f_i that is a subset of S^x yet does not contain x. Since $f_i \to f_l$ is sound for every i we have that, by Proposition 4.4, for every minterm S^x of f containing x there is some minterm S_l of $f_l = g$ that is a subset of S^x not containing x. I.e. $f \to g$ is trivial, by Proposition 5.6, which is a contradiction. Property 2 is proved analogously. Finally, Property 3 is proved by appealing to read-onceness. Any such S_i and T_i must contain x by properties 1 and 2, yet their intersection must be a singleton by Theorem 4.10 since all f_i are read-once, whence the result follows. □

We notice that, since some S_i and T_i must exist for all i, by soundness, we can build a chain[2] of such minterms and maxterms preserving the intersection point. For a given derivation, let us call a choice of such minterms and maxterms *critical* (see Figure 1).

Lemma 5.7 will play a crucial role in the proof of the following theorem, which constitutes the main result of this paper.

We now state the main result of this section and the main technical contribution of this work, from which we can obtain our further results. While we state this result for terms, in order to access simultaneously the notions of relation webs and Boolean semantics, notice that this could equally be stated in the setting of read-once Boolean functions due to Gurvich's result, Thm. 4.10, but this is beyond the scope of this work.

Theorem 5.8. *For every sequence of negation-free constant-free linear terms $s = t_0 < t_1 < \cdots < t_l = t$ on variable set X of size n, such that $l > 0$ and such that $s \to t$ is not trivial, we have that $l = O(n^4)$.*[3]

The remainder of this section is devoted to the proof of Theorem 5.8. For this let us fix π to denote the sequence $s = t_0 < t_1 < \cdots < t_l = t$. Recall that $l > 0$, implies that we have

[2]More generally we can build lattices of these terms since the properties are universally quantified.
[3]More precisely, we have $l \leq n^4$.

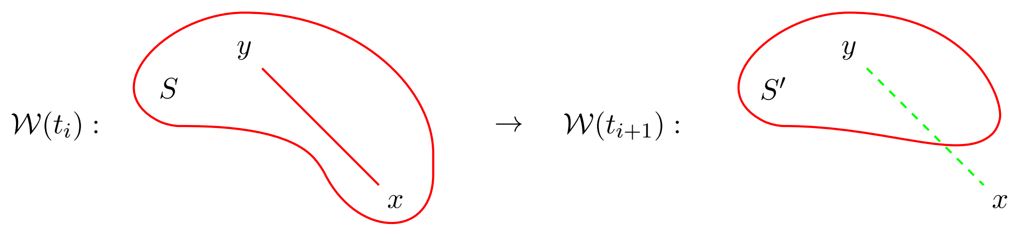


Figure 2: In the proof of Proposition 5.10, S' cannot contain both x and y, so we can assume without loss of generality that it does not contain x.

$s < t$, and therefore s and t are distinct modulo AC, which means they compute distinct Boolean functions by Theorem 4.8 and have distinct relation webs.

We now fix, for each $x \in X$ and $0 \leq i \leq l$, some choice of S_i^x and T_i^x as critical minterms and maxterms, respectively, of t_i, cf. Lemma 5.7. I.e. we have that, for each $x \in X$:

(1) $S_i^x \cap T_i^x = \{x\}$ for each $i \leq l$.
(2) $S_0^x \supseteq S_1^x \supseteq \cdots \supseteq S_l^x$.
(3) $T_0^x \subseteq T_1^x \subseteq \cdots \subseteq T_l^x$.

We denote the size of the critical minterms and maxterms of t_i by $|S_i^x|$ and $|T_i^x|$, respectively. Now we define:

$$\nu(t_i) := \sum_{x \in X} |S_i^x| \qquad \text{and} \qquad \mu(t_i) := \sum_{x \in X} |T_i^x| \tag{5.1}$$

Observation 5.9. Note that we always have $|S_i^x|, |T_i^x| \leq n$ because a minterm or maxterm has size at most n, and therefore we have $\nu(t_i), \mu(t_i) \leq n^2$ for all t_i in π.

The following two propositions now form the core of the argument. The first says that whenever an $\wedge$-edge becomes labelled $\vee$, some minterm strictly decreases in size, and the second one says that if a minterm strictly decreases in size then some critical maxterm must strictly increase in size. Thus the proof of Theorem 5.8 that follows again relies crucially on the interplay between the Boolean function setting and the graph-theoretic setting.

Proposition 5.10. *Suppose, for some $i < l$, we have that x —— y in $\mathcal{W}(t_i)$ and $x \cdots\cdots y$ in $\mathcal{W}(t_{i+1})$. Then there is a minterm S of t_i, and a minterm S' of t_{i+1} such that $S' \subsetneq S$.*

Proof. Take any maximal $\wedge$-clique in $\mathcal{W}(t_i)$ containing x and y, of which there must be at least one. This must have a $\wedge$-subclique which is maximal in $\mathcal{W}(t_{i+1})$, by Proposition 4.4 and Theorem 4.6. This subclique cannot contain both x and y, so the inclusion must be strict (see Figure 2). $\square$

Proposition 5.11. *Suppose for $j > i$ there is some minterm S_i of t_i and some minterm S_j of t_j such that $S_j \subsetneq S_i$. Then, for some variable $x \in X$, we have that $T_i^x \subsetneq T_j^x$.*

Proof. We let x be some variable in $x \in S_i \setminus S_j$, which must be nonempty by hypothesis. By Theorem 4.10 we have that $|T_i^x \cap S_i| = 1$, so it must be that $T_i^x \cap S_i = \{x\}$ by construction. On the other hand we also have that $|T_j^x \cap S_j| = 1$, and so there is some (unique) $y \in T_j^x \cap S_j$. Now, since $S_i \supsetneq S_j$ we must have $y \in S_i$. However we cannot have $y \in T_i^x$ since that would imply that $\{x, y\} \subseteq T_i^x \cap S_i$, contradicting the above. Since by soundness, we have $T_i^x \subseteq T_j^x$ we can now conclude that $T_i^x \subsetneq T_j^x$ as required, because $y \in T_j^x$ and $y \notin T_i^x$ (see Figure 3). $\square$

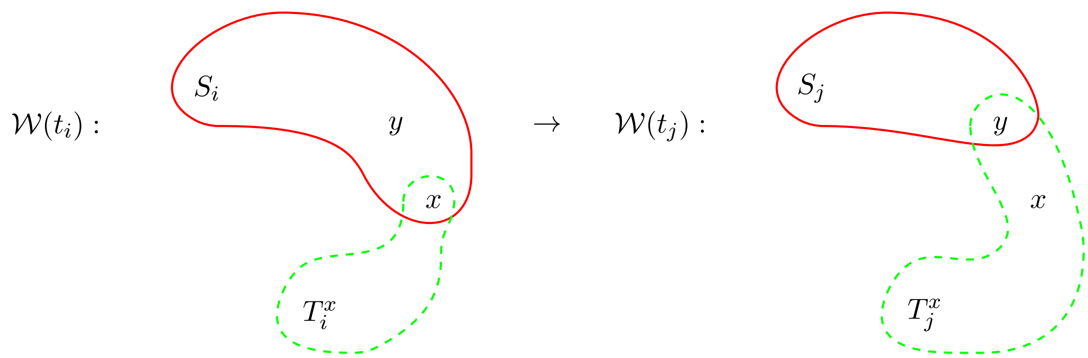


Figure 3: If some minterm becomes smaller then some critical maxterm must become bigger.

Lemma 5.12 (Increasing measure)**.** *The lexicographical product $\mu \times e_\wedge$ is strictly increasing at each step of π.*

Proof. Notice that, by Lemma 5.7.2, we have that $T_0^x \subseteq T_1^x \subseteq \cdots \subseteq T_l^x$, which means that μ is non-decreasing. So let us consider the case that $e_\wedge$ decreases at some step and show that μ must strictly increase. If $e_\wedge(t_i) > e_\wedge(t_{i+1})$ then we must have that some edge is labelled $\wedge$ in $\mathcal{W}(t_i)$ and labelled $\vee$ in $\mathcal{W}(t_{i+1})$. Hence, by Proposition 5.10 some minterm has strictly decreased in size and so by Proposition 5.11 some critical maxterm must have strictly increased in size. □

From here we can finally prove our main theorem:

Proof of Theorem 5.8. By Observation 5.9 and Proposition 3.3 we have that $\mu = O(n^2) = e_\wedge$ and so, since $s \to t$ is nontrivial, it must be that the length l of π is $O(n^4)$, as required. □

Notice that, while the various settings exhibit a symmetry between $\wedge$ and $\vee$, it is the property of soundness that induces the necessary asymmetry required to achieve this result.

6. No complete linear term rewriting system for propositional logic

Recall that a linear inference is a sound linear rewrite rule. We denote the set of all linear inferences by L. In this section we will show that there is no sound linear term rewriting system that is complete for L unless **coNP** = **NP**.

We start with the following observation made in [Str12]:

Proposition 6.1. L *is* **coNP**-*complete.*

This result is the reason, from the point of proof theory, one might restrict attention to only linear inferences at all: every Boolean tautology can be written as a linear inference. As we can see from the proof that follows, the translation is not very complicated, and it induces an at most quadratic blowup in size from an input tautology to a linear inference.

We include the proof here for completeness, and also since the statement here differs slightly from that in [Str12].

Proof of Proposition 6.1. That L is in **coNP** is due to the fact that checking soundness of a rewrite rule $s \to t$ can be reduced to checking validity of the formula $\bar{s} \vee t$. To prove **coNP**-hardness, we reduce validity of general tautologies to soundness of linear rewrite rules. Let t' be the term obtained from t (which is assumed to be in NNF) by doing the following for each positive variable x: let n be the number of occurrences of x in t, and let m be the number of occurrences of $\bar{x}$ in t. If $n = 0$ replace every occurence of $\bar{x}$ by $\perp$, and if $m = 0$ replace every occurrence of x by $\perp$. Otherwise, introduce $2mn$ fresh (positive) variables $x'_{i,j}, x''_{i,j}$ for $1 \leq i \leq n$ and $1 \leq j \leq m$. Now, for $1 \leq i \leq n$, replace the i^{th} occurrence of x by $x'_{i,1} \vee \ldots \vee x'_{i,m}$ and, for $1 \leq j \leq m$, replace the j^{th} occurrence of $\bar{x}$ by $x''_{1,j} \vee \ldots \vee x''_{n,j}$.

Now t' is a linear term (without negation), and its size is quadratic in the size of t. Let s' be the conjunction of all pairs $x' \vee x''$ of variables introduced in the construction of t'. Clearly $\mathit{Var}(s') = \mathit{Var}(t')$ and s' is also a linear term of the same size as t'. Furthermore, t is a tautology if and only if $s' \to t'$ is sound. To see this, let s'' and t'' be obtained from s' and t', respectively, by replacing each x'' by $\bar{x}'$. Then s'' always evaluates to 1, and t'' is a tautology if and only if t is a tautology. □

In the next step we extend the result of the previous section to all linear inferences, i.e., we have to deal with constants, negation, erasure, and trivialities. Some of the following results appeared already in [Das13], so we present only brief arguments here.

Definition 6.2. We define the following rules:

$$\mathsf{s} : x \wedge (y \vee z) \to (x \wedge y) \vee z \qquad \mathsf{m} : (w \wedge x) \vee (y \wedge z) \to (w \vee y) \wedge (x \vee z)$$

We call the former *switch* and the latter *medial* [BT01].

In what follows we implicitly assume that rewriting is conducted modulo ACU.

Lemma 6.3. *If s and t are negation-free linear terms on $x_1, \ldots, x_n$ and $s \leq t$, then there are linear terms s', t', u such that:*

(1) There are derivations $s \xrightarrow[\mathsf{s},\mathsf{m}]{*} s' \vee u$ and $t' \vee u \xrightarrow[\mathsf{s},\mathsf{m}]{*} t$ of length $O(n^2)$.

(2) $s' \to t'$ is sound and nontrivial.

Proof. See [Das13]. Briefly, the idea is that u is obtained by repeatedly 'moving aside' trivial variables, using s, m and ACU, until there are no trivialities remaining in $s' \to t'$. The bound of $O(n^2)$ is not explicitly mentioned in [Das13], but it is clear from direct inspection of that construction. □

Remark 6.4. Notice that, while the derivations from Lemma 6.3.(1) above are small in size, they are difficult to compute, due to the inherent complexity of detecting triviality. This problem is in fact already **coNP**-complete, since validity of an arbitrary linear inference $A \to B$ can be reduced to detecting triviality at x in $A \wedge x \to B \vee x$, where x is fresh. This is not an issue in what follows since we are only concerned with the existence of small derivations, and so the existence of an **NP**-algorithm, for various inferences.

A left- and right-linear rewrite rule may still erase or introduce variables, i.e. there may be variables on one side that do not occur on the other.[4] However, notice that any such situation must constitute a triviality at such a variable, since the soundness of the step is not dependent on the value of that variable.

Proposition 6.5. *Suppose $\rho : l \to r$ is linear, and there is some variable x occurring in only one of l and r. Then ρ is trivial at x.*

[4]Usually, term rewrite rules are required to not introduce new variables from left to right, but it does no harm to make this generalisation here.

If a (positive) variable x occurs negatively on both sides of a linear rule then $\bar{x}$ can be replaced soundly by x on both sides. Otherwise, if x occurs positively on one side and negatively on the other, it must be that we have a triviality at x.

Proposition 6.6. *For each linear rule ρ either there is a negation-free linear rule that is equivalent to ρ (i.e. with the same reduction steps), or ρ is trivial.*

Recall that ACU' preserves the Boolean function computed by a term, and that every linear term is ACU'-equivalent to $\bot$, $\top$ or a unique constant-free linear term.

Proposition 6.7. *Let R be a complete linear system. Then any constant-free nontrivial linear inference $s \to t$ has a constant-free R/ACU'-derivation.*

Proof. By completeness there is an R-derivation of $s \to t$. Now reduce every line by ACU' to a constant-free term or $\bot$ or $\top$ (e.g. as shown in [Das11]). If some line reduces to $\bot$ or $\top$ and another does not, then $s \to t$ is trivial, and if every line reduces to $\bot$ or every line reduces to $\top$ then the derivation collapses and is no longer constant-free. □

We can combine the previous three propositions to obtain the following:

Proposition 6.8. *The following are equivalent:*

(1) *There is a sound linear system complete for sound constant-free negation-free nontrivial linear inferences.*
(2) *There is a sound constant-free negation-free nontrivial linear system, which is non-erasing and non-introducing, complete for the set of such inferences.*

Now, combining our results from Section 5 with the normal forms obtained above, we arrive at the main result of this work:

Theorem 6.9. *If there is a sound and complete linear system for L, then there is one that has a $O(n^4)$-length derivation for each linear inference on n variables.*

Proof. Assume we have a sound and complete linear system R for L, and let $s \to t$ be a linear inference on n variables. By Lemma 6.3 we have linear terms s', t' such that $|s'| \leq |s|$ and $s' \to t'$ is sound, linear, and nontrivial. By Propositions 6.5, 6.6 and 6.7 we can assume that s', t' have the same size and are free of negation and constants. By Proposition 6.8 there is a sound constant-free negation-free nontrivial linear system R', which is non-erasing and non-introducing, complete for the set of such inferences. Therefore we have a derivation $s' \xrightarrow[R']{*} t'$, and by Theorem 5.8, the length of this derivation is $O(n^4)$. And so, by Lemma 6.3.(1), we can construct a derivation for $s \to t$ with overall length $O(n^4)$ in $R' \cup \{\mathsf{s}, \mathsf{m}\}/ACU$. □

Corollary 6.10. *There is no sound linear system complete for L unless* $\mathbf{coNP} = \mathbf{NP}$.

Proof. By Proposition 6.1, L is **coNP**-complete, and Theorem 6.9 induces an **NP** decision procedure for L for any such system R, as we can guess a correct sequence of R-steps to derive $s \to t$. □

7. On the canonicity of switch and medial

In this section we investigate to what extent the two rules switch and medial from Definition 6.2, which play a crucial role in the proof theory of classical propositional logic, are "canonical". Recall that these rules are as follows:

$$\mathsf{s} : x \wedge (y \vee z) \to (x \wedge y) \vee z \qquad \mathsf{m} : (w \wedge x) \vee (y \wedge z) \to (w \vee y) \wedge (x \vee z)$$

First we observe that both rules are minimal in the following sense:

Definition 7.1. A sound linear rewrite rule $\rho \colon l \to r$ is *minimal* if there is no linear term t on the same variables as l and r such that $l < t < r$.

Proposition 7.2. *Switch and medial are minimal.*

Proof. By exhaustive search on all terms of size 3 (for switch) and 4 (for medial). □

Observe that, seen as an action on relation webs, switch and medial preserve $\vee$-edges and $\wedge$-edges, respectively. Formally, let us consider the following two properties of a linear inference ρ:

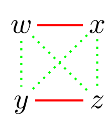
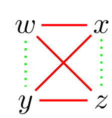
(*) If $s \underset{\rho}{\to} t$ then, whenever x ⋯⋯ y in $\mathcal{W}(s)$, we have that x ⋯⋯ y in $\mathcal{W}(t)$.

(**) If $s \underset{\rho}{\to} t$ then, whenever x —— y in $\mathcal{W}(s)$, we have that x —— y in $\mathcal{W}(t)$.

Our first canonicity result is that medial is the *only* sound linear inference that is minimal and satisfies (**). In fact, we will show the stronger property that any sound linear rule satisfying (**) is already derivable by medial. For this we will appeal to a characterisation of medial that appeared in [Str07a]:

Definition 7.3. Let s and t be linear terms on a set X of variables. We write $s \lhd\!\!\!\blacktriangleright t$ if:

1. Whenever x —— y in $\mathcal{W}(s)$ we have that x —— y in $\mathcal{W}(t)$.
2. Whenever x ⋯⋯ y in $\mathcal{W}(s)$ and x —— y in $\mathcal{W}(t)$, there are $w, z \in X$ such that,

 in $\mathcal{W}(s)$ and  in $\mathcal{W}(t)$.

The following result appeared in [Str07a], where a detailed proof may be found.

Proposition 7.4 (Medial criterion)**.** *$s \lhd\!\!\!\blacktriangleright t$ if and only if $s \overset{*}{\underset{\mathsf{m}}{\to}} t$.*

Using this result we can show that any sound linear rule satisfying (**) is already derivable by medial:

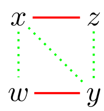
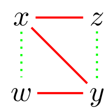
Theorem 7.5. *Let s and t be linear terms on a variable set X. The following are equivalent:*

1. *$s \leq t$ and for all $x, y \in X$ we have x —— y in $\mathcal{W}(s)$ implies x —— y in $\mathcal{W}(t)$.*
2. *$s \lhd\!\!\!\blacktriangleright t$.*
3. *$s \overset{*}{\underset{\mathsf{m}}{\to}} t$.*

For the proof let us say, if t is a linear term with $x, y, z \in \mathit{Var}(t)$, that y *separates* x *from* z in $\mathcal{W}(t)$ if x —— y in $\mathcal{W}(t)$ and y ⋯⋯ z in $\mathcal{W}(t)$.

Proof of Theorem 7.5. We have that $2 \Longrightarrow 3$ by Proposition 7.4 and $3 \Longrightarrow 1$ by inspection of medial, so it suffices to show $1 \Longrightarrow 2$. For this, assume 1 and suppose x ⋯⋯ y in $\mathcal{W}(s)$ and x —— y in $\mathcal{W}(t)$, and let S be a minterm of s containing x. We must have $S \supsetneq \{x\}$

since x —— y in $\mathcal{W}(t)$ and $s \to t$ is sound.[5] Similarly there must be a maxterm T of t containing y such that $T \supsetneq \{y\}$. Now, by 1, it must be that S (resp. T) is also a minterm (resp. maxterm) of t (resp. s),[6] and so, by Theorem 4.10, there is some (unique) $z \in S \cap T$ which, by definition, separates x from y in both $\mathcal{W}(s)$ and $\mathcal{W}(t)$. By a symmetric argument we obtain a w separating y from x in both $\mathcal{W}(s)$ and $\mathcal{W}(t)$. By construction, w and z must be distinct, so we have the following situation,

 in $\mathcal{W}(s)$ and  in $\mathcal{W}(t)$.

whence 2 follows by P_4-freeness. □

Corollary 7.6 (Canonicity of medial)**.** *Medial is the only sound linear inference that is minimal and has property (**).*

Proof. By Theorem 7.5, any linear inference satisfying (**) can be derived by medial. The result then follows by minimality of medial. □

Using these results, we are actually able to improve the length bound on nontrivial linear derivations that we proved earlier:

Corollary 7.7. *The bound in Theorem 5.8 can be improved to* $O(n^3)$.

For the proof, let us first define $\#_\wedge(t)$ to be the number of $\wedge$ symbols occurring in t.

Proof of Corollary 7.7. Instead of using $e_\wedge$ in Lemma 5.12, use $\#_\wedge$, which is linear in the size of the term. If no $\wedge$-edge becomes labelled $\vee$ in some step, it follows by Theorem 7.5 that the step is derivable using medial, and so $\#_\wedge$ must have strictly decreased, by inspection of medial. □

While we have just shown a fairly succinct form of canonicity for medial, it turns out that we cannot obtain an analogous result for switch: switch is *not* the only sound linear inference that is minimal and satisfies (*). To see this, simply recall the example of (1.7) from the introduction:

$$\begin{array}{rl} & (u \vee (v \wedge v')) \wedge ((w \wedge w') \vee (x \wedge x')) \wedge ((y \wedge y') \vee z) \\ \to & (u \wedge (w \vee y)) \vee (w' \wedge y') \vee (v' \wedge x') \vee ((v \vee x) \wedge z) \end{array}$$

Notice, however, that this inference does not preserve the number $\#_\wedge$ of conjunction symbols in a term. In fact, switch is the only nontrivial linear inference we know of that preserves $\#_\wedge$, although there are known trivial examples that even *increase* $\#_\wedge$, for instance the "supermix" rules from [Das13]:

$$x \wedge (y_1 \vee \cdots \vee y_n) \quad \to \quad x \vee (y_1 \wedge \cdots \wedge y_n) \tag{7.1}$$

This leads us to the following conjecture:

Conjecture 7.8. *If $s \to t$ is sound, nontrivial, satisfies (*) and $\#_\wedge(s) = \#_\wedge(t)$, then $s \xrightarrow{*}_{\mathsf{s}} t$.*

[5] Recall that, by Proposition 4.4 and Theorem 4.6, there must a subset of S which is a maximal $\wedge$-clique in $\mathcal{W}(t)$.

[6] Since by 1, $\wedge$-edges (resp. $\vee$-edges) are preserved left-to-right (resp. right-to-left) and so $\wedge$-cliques (resp. $\vee$-cliques) must be preserved (resp. reflected). Of course, these must be maximal by soundness.

Remark 7.9. This conjecture already implies our main result, Theorem 5.8, since $\#_\wedge \times e_\wedge$ would be a strictly decreasing measure. This measure can also be used for the usual proof of termination of $\{\mathsf{s}, \mathsf{m}\}$ (modulo AC) and also yields a cubic bound on termination. In this work we have matched that bound for *all* linear derivations in the case of weak normalisation, and in the case of strong normalisation for inferences that are not trivial. Some preliminary research shows that the length-bound for termination of $\{\mathsf{s}, \mathsf{m}\}$ can be improved to a quadratic. We conjecture that such an improvement is also possible in the case of (nontrivial) linear derivations in general.

The supermix rules (7.1) are also exemplary of linear inferences that satisfy neither (*) nor (**). However, again, we have not been able to identify any nontrivial examples of this, and we further conjecture the following:

Conjecture 7.10. *There is no nontrivial minimal sound linear inference that satisfies neither (*) nor (**).*

An interesting observation is that Conjecture 7.10 and Corollary 7.6 together entail that medial is the only linear inference that allows contraction to be reduced to atomic form. To see what this means, consider again (1.6) from the introduction. The steps marked c are instances of the contraction rule $x \vee x \to x$. If the contractum of such a step is simply a variable, then we call that instance of contraction *atomic*, denoted by $\mathsf{ac}\!\downarrow$ as in [BT01]. Dually, the atomic instances of 'cocontraction' $t \to t \wedge t$, when the redex is simply a variable, are denoted by $\mathsf{ac}\!\uparrow$. We say that a linear inference $\rho : l \to r$ *reduces contraction to atomic form* if, for every term t, we have $t \vee t \xrightarrow[\rho, \mathsf{ac}\downarrow]{*} t$ and $t \xrightarrow[\rho, \mathsf{ac}\uparrow]{*} t \wedge t$.

Conjecture 7.11. *Medial is the only minimal linear inference that reduces contraction to atomic form. More precisely, for every linear inference $\rho : l \to r$ that reduces contraction to atomic form we have $l \xrightarrow[\mathsf{m}]{*} r$.*

Proof using Conjecture 7.10. Assume $t \vee t \xrightarrow[\rho, \mathsf{ac}\downarrow]{*} t$ and $t \xrightarrow[\rho, \mathsf{ac}\uparrow]{*} t \wedge t$. Since t can contain $\vee$ and $\wedge$, it must be the case that ρ replaces $\vee$-edges in $\mathcal{W}(l)$ by $\wedge$-edges in $\mathcal{W}(r)$. By Conjecture 7.10 ρ does not replace $\wedge$-edges in $\mathcal{W}(l)$ by $\vee$-edges in $\mathcal{W}(r)$. By Theorem 7.5 we must have $l \xrightarrow[\mathsf{m}]{*} r$. □

8. On the normalisation of deep inference proofs

Another application of our results is to the normalisation of deep inference proofs. This is typically done via rewriting on certain graphs extracted from derivations, known as *atomic flows* [GG08, GGS10]. The main sources of complexity here are 'contraction loops', and so a lot of effort has gone into the question of whether such features can be eliminated. A consequence of our main result is that this is impossible for a large class of deep inference systems.

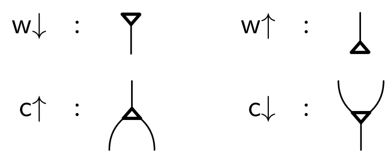
To show this, we will in this section only consider rewriting systems on positive terms, and then make some remarks about negative rules at the end of this section. We consider systems with the standard structural rules of deep inference, extended by an arbitrary (polynomial-time decidable) set of linear rules.

A formal definition of atomic flows can be found in [GG08], where they were first presented. And alternative definition can be found in [GGS10]. We give an informal definition below which is sufficient for our purposes.

Definition 8.1 (Structural rules and atomic flows)**.** We define the system cw as follows:

$$\begin{array}{lcl@{\qquad}lcl} \mathsf{w}\!\downarrow & : & x \to x \vee y & \mathsf{w}\!\uparrow & : & x \wedge y \to x \\ \mathsf{c}\!\uparrow & : & x \to x \wedge x & \mathsf{c}\!\downarrow & : & x \vee x \to x \end{array}$$

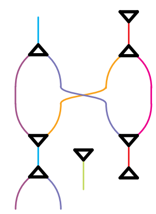
If S is the extension of cw by a set of linear rules and π is an S-derivation (written as a vertical list), then the *atomic flow* of π, denoted $\mathit{fl}(\pi)$, is the graph obtained by tracing the paths of each variable through the derivation, designating nodes at cw steps as follows:



Example 8.2. Consider the system MSKS obtained by extending cw by the rules switch and medial, from Definition 6.2, as well as rules ACU from Section 2 for associativity, commutativity and constants. This is equivalent to the monotone fragment of the common deep inference system SKS [BT01].

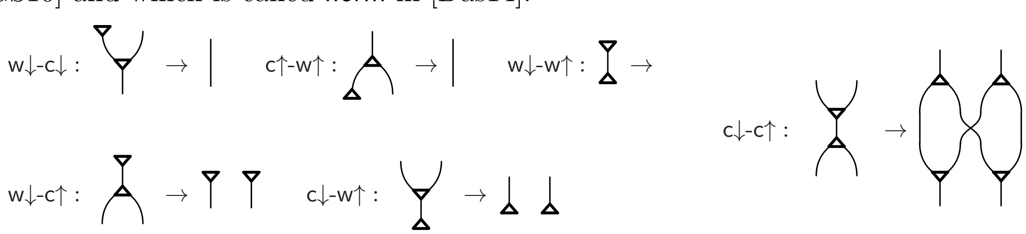
Here is an example of an MSKS rewrite derivation, with redexes underlined, and its atomic flow. The colours are used to help the reader associate edges with variable occurrences in the derivation.

$$\begin{array}{rc} & \underline{x} \\ \xrightarrow[\mathsf{w}\downarrow]{} & \underline{x} \vee x \\ \xrightarrow[\mathsf{c}\uparrow]{} & (x \wedge x) \vee \underline{x} \\ \xrightarrow[\mathsf{c}\uparrow]{} & \underline{(x \wedge x) \vee (x \wedge x)} \\ \xrightarrow[\mathsf{c}\downarrow]{} & \underline{x} \wedge x \\ \xrightarrow[\mathsf{w}\downarrow]{} & \underline{(x \vee y) \wedge x} \\ \xrightarrow[\mathsf{s}/AC]{} & \underline{x} \vee (y \wedge x) \\ \xrightarrow[\mathsf{c}\uparrow]{} & (x \wedge x) \vee \underline{(y \wedge x)} \\ \xrightarrow[\mathsf{w}\uparrow]{} & (x \wedge x) \vee y \end{array} \tag{8.1}$$



Definition 8.3 (Flow rewriting systems)**.** A *flow rewriting system* (FRS) is a set of graph rewriting rules on atomic flows. We say that a FRS R *lifts* to a TRS S if, for every S-derivation $\pi : s \xrightarrow[S]{*} t$ and reduction step $\mathit{fl}(\pi) \to \phi$ there is a S-derivation $\pi' : s \xrightarrow[S]{*} t$ with $\mathit{fl}(\pi') = \phi$.

Example 8.4. Consider the following FRS, which is a subset of rules occurring in [GG08, GGS10] and which is called norm in [Das14].



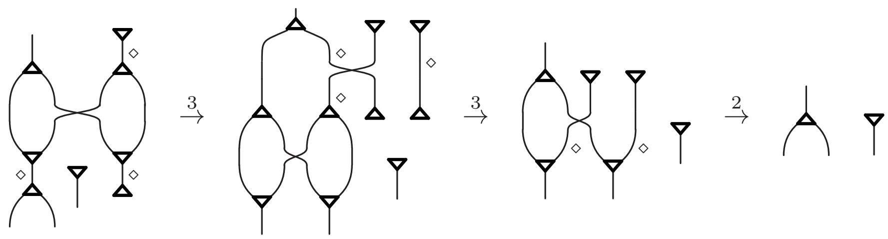
(8.2)

We have essentially the following result from [GG08]:

Proposition 8.5. norm *lifts to any extension of* MSKS *by linear rules.*

The proof of this is beyond the scope of this work, but crucially relies on the presence of switch, medial and *ACU* to make the w and c rules atomic, cf. 1.6, and thereby allow these steps to permute more freely in a derivation.

For example, here is a norm-derivation that normalises the flow from (8.1),



(8.3)

where redexes are marked by $\diamond$.

norm is strongly normalising, as implied by results in [GG08]. In the works [Das12] and [Das15] the main source of complexity of (weak) normalisation under norm is the presence of *contraction loops*. In their absence the time complexity of normalisation is polynomially bounded.

Definition 8.6 (Contraction loops, from [Das12])**.** Given a flow ϕ, a *contraction loop* is a pair of nodes (ν_1, ν_2) such that there two distinct paths from ν_1 to ν_2 in ϕ.

It turns out that our previous results imply that no deep inference system that extends MSKS by linear rules can admit a flow-rewriting normalisation procedure that eliminates contraction loops.

Theorem 8.7. *Let R be a FRS such that, for any flow ϕ, there is some flow ψ free of contraction loops such that $\phi \xrightarrow[R]{*} \psi$. Then R lifts to no sound extension of* MSKS *by linear rules unless* **coNP** $=$ **NP**.

Before giving the proof, let us first make the following observation:

Proposition 8.8. *If a flow ϕ is free of contraction loops and $\phi \xrightarrow[\mathsf{norm}]{*} \psi$, then ψ is also free of contraction loops.*

Proof sketch. By induction on the length of a norm-derivation under a careful analysis of the reduction steps in norm. □

We can now give a proof of the theorem above.

Proof of Theorem 8.7. Let us assume that R lifts to such a system S and show that **coNP** $=$ **NP**. Let $s \to t$ be an arbitrary linear inference and let s', t', u be linear terms obtained by Lemma 6.3. By completeness of S let $\pi : s' \xrightarrow[S]{*} t'$ and let $\pi' : s' \xrightarrow[S]{*} t'$ be obtained by first reducing $fl(\pi)$ under R to a flow free of contraction-loops and then to a normal form under norm, and finally lifting the resulting derivations to S by assumption and Proposition 8.5. Notice that $fl(\pi')$ is free of contraction loops by assumption and Proposition 8.8.

First we show that $fl(\pi')$ must be free of c↓ and c↑ nodes. Consider a topmost c↓ node and the maximal paths leading to its upper edges. Since $fl(\pi')$ is free of contraction loops we can assume these two paths are disjoint. If one of the paths begins with a w↓ node then there must be either a w↓-c↓ or w↓-c↑ redex in $fl(\pi')$, contradicting normality under norm. Therefore both paths must begin with variables from s', contradicting linearity of s'. The argument for c↑ is similar, by consideration of a bottommost such node.

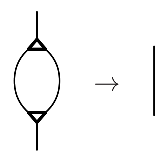
Now we show that $fl(\pi')$ is free of w↓ and w↑ nodes. Suppose there is a w↑ node and consider the maximal path leading to its edge. This cannot be connected to any other node since this would yield a redex. Therefore this path must begin from some variable x of s'. Consequently the occurrence of x in t' must originate from a w↓ node.[7] However this would imply that $s' \to t'$ is trivial at x, contradicting the fact that $s' \to t'$ is nontrivial.

Therefore $fl(\pi')$ is just a flow of simple edges, and so π' is linear. Since it also derives a nontrivial linear inference, it must have polynomial length by Theorem 5.8. Finally, by Lemma 6.3, this means that there is a polynomial-size S-derivation of $s \to t$. Since the choice of this linear inference was arbitrary, we thus have an **NP** algorithm for L. □

In particular we can conclude that a particularly natural FRS for eliminating contraction loops cannot be correct for a large class of deep inference systems, partially answering questions occurring in previous works and correspondences:

Corollary 8.9. *The following flow-rewriting rule*

c↑-c↓ :

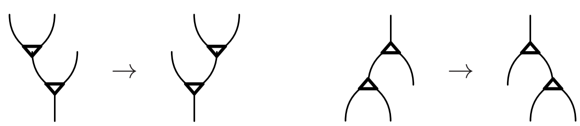


lifts to no sound extension of MSKS *by linear rules unless* **coNP** $=$ **NP**.

The proof follows immediately from Theorem 8.7 and the following observations:

Proposition 8.10. *We have the following:*

(1) *The FRS* assoc *given by the following rules,*



lifts to any extension of MSKS *by linear rules.*

(2) *Any flow can be reduced in* c↑-c↓ + c↓-c↑ + assoc *to one free of contraction loops.*

Proof sketch. 1 is routine, so we prove 2. For a c↑ node in a flow, let its *weight* be its distance from the top of the flow. We argue that c↓-c↑ + c↑-c↓/assoc is terminating, by noticing that the multiset of weights of c↑ nodes in a flow decreases[8] by any application of c↓-c↑ or c↑-c↓ and is preserved by assoc. Finally, we observe that there cannot be any contraction loop in a normal form of c↓-c↑ + c↑-c↓/assoc since it would contain either a c↓-c↑ or c↑-c↓ redex, modulo assoc. □

[7] Recall that we already have that there are no c↓ or c↑ nodes, so this follows immediately.

[8] Formally it suffices to associate a flow ϕ with the sum $\sum 2^{2w(\nu)}$, where ν ranges over c↑ nodes in ϕ and $w(\nu)$ is the weight of ν, and consider the usual order on natural numbers.

Remark 8.11. Here we only considered systems that extend the monotone fragment of the deep inference system SKS by arbitrary linear rules. To some extent the results above generalise to extensions by other rules, but there are certain interesting cases that could be points of further study.

First, of course, there could be rules that allow an interplay between positive and negative variables, most notably the identity and cut rules from SKS:

$$\top \quad \to \quad x \vee \bar{x} \qquad\qquad x \wedge \bar{x} \quad \to \quad \bot$$

Their normalisation behaviour is very different from that of the structural rules contraction and weakening, and so call for an independent analysis altogether. [9]

Another interesting case is when SKS is extended by nonlinear rules. In a particularly extreme case one can envisage rules that are 'multiplicative' but not linear. For instance, consider the following monotone formula, denoted $t(w,x,y,z)$:

$$(w \wedge x) \vee ((w \vee x) \wedge (y \vee z)) \vee (y \wedge z)$$

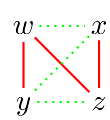
This computes the threshold function $TH^4_2(w,x,y,z)$, i.e. "at least two of w,x,y,z are true". Since this is a symmetric function, we can construct the following sound rule:[10]

$$t(w,x,y,z) \quad \to \quad t(w,y,x,z)$$

It can be considered 'multiplicative', in the sense that each variable occurs with the same multiplicity, 2, on each side, but it cannot be an instance of a linear rule, since we rely on the logical dependencies between variable occurrences for soundness.

9. Towards proof theory on arbitrary graphs

In this section we consider arbitrary complete graphs with edges labelled by $\wedge$ and $\vee$, i.e. graphs that are not necessarily P_4-free, and we consider their $\wedge$-maxcliques and $\vee$-maxcliques. Such graphs no longer correspond to terms, in fact they do not even correspond to Boolean functions since Theorem 4.6 breaks down by the example of (3.1):



The problem here is that there is a $\wedge$-maxclique $\{w,z\}$ and a $\vee$-maxclique $\{x,y\}$ which are disjoint, so under the association of $\wedge$- and $\vee$-maxcliques to minterms and maxterms respectively via Theorem 4.6, one would be able to force this graph to evaluate to 0 and 1 simultaneously by the assignment $\{w \mapsto 1, x \mapsto 0, y \mapsto 0, z \mapsto 1\}$.

On the other hand, the alternative definitions of entailment from Proposition 4.4 still remain meaningful in such a setting. Inspired by this, let us consider the following relations on graphs:

- $G \underset{\wedge}{\to} G'$ if, for any $\wedge$-maxclique C of G, there is a $\wedge$-maxclique C' of G' with $C' \subseteq C$.
- $G \underset{\vee}{\to} G'$ if, for any $\vee$-maxclique C of G', there is a $\vee$-maxclique C' of G with $C' \subseteq C$.

They have the following important properties, whose proofs are routine:

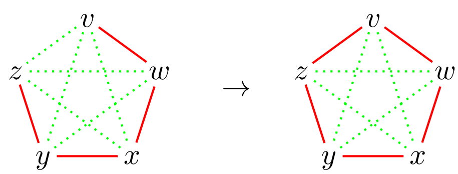
[9] We are aware that work studying linear systems extended by such rules is currently being pursued by Guglielmi, McCusker and Santamaria. This line of research is also related to [Lam07].

[10] In fact it would be sound for any permutation of variables, but this is the prototypical interesting case.

Proposition 9.1. *$\underset{\wedge}{\to}$ and $\underset{\vee}{\to}$ are reflexive and transitive.*

The point here is that, even though maximal cliques no longer correspond to minterms and maxterms, the notion of entailment induced by maximal cliques remains stable: if one starts with a P_4-free graph and applies one of the relations $\underset{\wedge}{\to}$ or $\underset{\vee}{\to}$ iteratively, and finishes with a P_4-free subgraph, then the underlying implication is sound, even if many of the intermediate graphs are not P_4-free, and so do not correspond to Boolean functions at all.

For instance, consider the following reduction:



This can easily be seen to be an instance of $\underset{\wedge}{\to}$, since only a new $\wedge$-maxclique, $\{v, z\}$, is added. On the other hand, its *inverse* is an instance of $\underset{\vee}{\to}$. Consequently the relations $\underset{\wedge}{\to}$ and $\underset{\vee}{\to}$ really are distinct, unlike their restrictions to P_4-free graphs.

Remark 9.2. Notice that there are alternative ways to define entailment for Boolean terms via their webs, but other intuitive choices do not satisfy Proposition 9.1 when generalised to arbitrary graphs in the natural way, and so do not induce any meaningful logic. For example, for linear terms s and t, we can show that $s \leq t$ if and only if every $\wedge$-maxclique of $\mathcal{W}(s)$ intersects every $\vee$-maxclique of $\mathcal{W}(t)$.[11] However, when generalised to arbitrary graphs, this relation is not even reflexive because of, again, the case of a P_4 configuration (3.1).

In further work we would like to study the logics induced by the relations $\underset{\wedge}{\to}$ and $\underset{\vee}{\to}$, and even systems where one may alternate between them any time a graph is, say, P_4-free. Such systems would be sound for Boolean logic when the source and target are P_4-free, under the association of a term to its web. They would also leave the world of Boolean functions altogether, as we previously mentioned, which bears semblance to algebraic proof systems for propositional logic such as Cutting Planes and Nullstellensatz (studied in, for example, [BPR97] and [BIK$^+$97]).

Furthermore, notice that our crucial Lemma 5.7 cannot immediately be generalised to the setting of arbitrary graphs due to the fact that $\wedge$-maxcliques no longer necessarily intersect $\vee$-maxcliques. It would be particularly interesting to examine the extent to which 'linear reasoning' can be recovered in this setting, sidestepping the shortcomings of P_4-free graphs (i.e. terms) we have studied in this work.

10. Final remarks

To some extent, this work can be seen as a justification for the approach of 'structural' proof theory: for any deductive system that can be embedded into a rewriting framework on Boolean terms, as we have considered here, completeness requires the inclusion of structural rules that introduce, destroy and duplicate formulae, unless **coNP** = **NP**. It is not difficult

[11] If s evaluates to 1, then one of its minterms must entirely be assigned to 1, and if this intersects every maxterm of t, then no maxterm of t is entirely assigned to 0, so t must also evaluate to 1. Conversely, if some minterm of s and some maxterm of t do not intersect, then we can simultaneously force s to evaluate to 1 and t to evaluate to 0.

to see that this covers a large class of proof systems, including essentially all the well-known systems based on formulae or related structures, e.g. Gentzen sequent calculi, Hilbert-Frege systems, Resolution, deep inference systems etc. On the other hand, as we mentioned in Section 9, proof systems based on other objects such as algebraic equations or graphs are not covered by our result. While the observation that structural behaviour is somewhat necessary for proof theory is perhaps not surprising, it is of natural theoretical interest and may also influence the construction of proof systems sensitive to space complexity.

There are clear thematic relationships between this line of work and linear logic. In some ways, we can see this work as contributing to the study of the 'multiplicative' fragment of Boolean logic. One particular connection we would like to point out is with Blass' model of linear logic in [Bla92], the first game semantics model of linear logic. The multiplicative fragment of this model in fact validates precisely the sound linear inferences of Boolean logic[12], which he calls 'binary tautologies'. Following from the paragraph above, it would seem that one drawback of this model is that it can admit no sound and complete proof system, unless **coNP** = **NP**, by virtue of our results.

Finally, this work contributes to the study of term rewriting systems for Boolean Algebras. While complete axiomatisations have been known since the early 20th century by Whitehead, Huntington, Tarski and others, these are typically sets of equations, rather than 'directed' rewrite rules which are more related to proof theory. It has been known for some time, for example, that there is no convergent TRS for Boolean Algebras [Soc91]; our result, in the same vein, shows there is no *linear* TRS for the linear fragment of Boolean Algebras.

References

[BCST96] Richard Blute, Robin Cockett, Robert Seely, and Todd Trimble. Natural deduction and coherence for weakly distributive categories. *J. of Pure and Applied Algebra*, 113:229–296, 1996.

[BdGR97] Denis Bechet, Philippe de Groote, and Christian Retoré. A complete axiomatisation of the inclusion of series-parallel partial orders. In H. Common, editor, *Rewriting Techniques and Applications, RTA 1997*, volume 1232 of *LNCS*, pages 230–240. Springer, 1997.

[BIK+97] Samuel R. Buss, Russell Impagliazzo, Jan Krajícek, Pavel Pudlák, Alexander A. Razborov, and Jirí Sgall. Proof complexity in algebraic systems and bounded depth frege systems with modular counting. *Computational Complexity*, 6(3):256–298, 1997.

[Bla92] Andreas Blass. A game semantics for linear logic. *Ann. Pure Appl. Logic*, 56(1-3):183–220, 1992.

[BPR97] Maria Luisa Bonet, Toniann Pitassi, and Ran Raz. Lower bounds for cutting planes proofs with small coefficients. *J. Symb. Log.*, 62(3):708–728, 1997.

[BT01] Kai Brünnler and Alwen F. Tiu. A local system for classical logic. In R. Nieuwenhuis and A. Voronkov, editors, *LPAR 2001*, volume 2250 of *LNCS*, pages 347–361. Springer, 2001.

[CH11] Yves Crama and Peter L Hammer. *Boolean functions: Theory, algorithms, and applications*. Cambridge University Press, 2011.

[Che67] Michael Chein. Algorithmes d'écriture de fonctions booléennes croissantes en sommes et produits. *Revue Française d'Informatique et de Recherche Opérationnelle*, 1:97–105, 1967.

[CR74] Stephen Cook and Robert Reckhow. On the lengths of proofs in the propositional calculus (preliminary version). In *Proceedings of the 6th annual ACM Symposium on Theory of Computing*, pages 135–148. ACM Press, 1974.

[Das11] Anupam Das. On the proof complexity of cut-free bounded deep inference. In K. Brünnler and G. Metcalfe, editors, *Tableaux 2011*, volume 6793 of *LNAI*, pages 134–148, 2011.

[Das12] Anupam Das. Complexity of deep inference via atomic flows. In S. Barry Cooper, Anuj Dawar, and Benedikt Löwe, editors, *Computability in Europe*, volume 7318 of *Lecture Notes in Computer Science*, pages 139–150. Springer-Verlag, 2012.

[12]Under the assiociation of $\otimes$ with $\wedge$ and $⅋$ with $\vee$.

[Das13] Anupam Das. Rewriting with linear inferences in propositional logic. In Femke van Raamsdonk, editor, *RTA'13*, volume 21 of *LIPIcs*, pages 158–173, 2013.

[Das14] Anupam Das. On the pigeonhole and related principles in deep inference and monotone systems. In Thomas Henzinger and Dale Miller, editors, *Joint Meeting of the 23rd EACSL Annual Conference on Computer Science Logic (CSL) and the 29th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 36:1–10. ACM, 2014.

[Das15] Anupam Das. On the relative proof complexity of deep inference via atomic flows. *Logical Methods in Computer Science*, 11(1):4:1–27, 2015.

[DS15] Anupam Das and Lutz Straßburger. No complete linear term rewriting system for propositional logic. In Maribel Fernández, editor, *26th International Conference on Rewriting Techniques and Applications (RTA 2015)*, volume 36 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 127–142, Dagstuhl, Germany, 2015. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[DZ97] Moshe Dubiner and Uri Zwick. Amplification by read-once formulas. *SIAM Journal on Computing*, 26(1):15–38, 1997.

[GG08] Alessio Guglielmi and Tom Gundersen. Normalisation control in deep inference via atomic flows. *Logical Methods in Computer Science*, 4(1):9:1–36, 2008.

[GGS10] Alessio Guglielmi, Tom Gundersen, and Lutz Straßburger. Breaking paths in atomic flows for classical logic. In *LICS 2010*, 2010.

[Gir87] Jean-Yves Girard. Linear logic. *Theor. Comput. Sci.*, 50:1–102, 1987.

[GS01] A. Guglielmi and L. Straßburger. Non-commutativity and MELL in the calculus of structures. In L. Fribourg, editor, *CSL 2001*, volume 2142 of *LNCS*, pages 54–68, 2001.

[Gug07] Alessio Guglielmi. A system of interaction and structure. *ACM Transactions on Computational Logic*, 8(1):1–64, 2007.

[Gug11] Alessio Guglielmi. Question on a class of tautologies. *Proof Theory mailing list*, 2011. `http://article.gmane.org/gmane.science.mathematics.prooftheory/809`.

[Gur77] V. A. Gurvich. Repetition-free boolean functions. *Uspekhi Matematicheskikh Nauk*, 32(1):183–184, 1977.

[Gur82] V. A. Gurvich. On the normal form of positional games. In *Soviet math. dokl*, volume 25, pages 572–574, 1982.

[HK90] Lisa Hellerstein and Marek Karpinski. Computational complexity of learning read-once formulas over different bases. Technical report, University of Bonn, 1990.

[HNW94] Rafi Heiman, Ilan Newman, and Avi Wigderson. On read-once threshold formulae and their randomized decision tree complexity. In *Theoret. Comp. Science*, pages 78–87, 1994.

[Kuz58] Aleksandr Vasilevich Kuznetsov. Non-repeating contact schemes and non-repeating superpositions of functions of algebra of logic. *Trudy Matematicheskogo Instituta im. VA Steklova*, 51:186–225, 1958.

[Lam07] François Lamarche. Exploring the gap between linear and classical logic. *Theory and Applications of Categories*, 18(18):473–535, 2007.

[Möh89] Rolf H. Möhring. Computationally tractable classes of ordered sets. In I. Rival, editor, *Algorithms and Order*, pages 105–194. Kluwer Acad. Publ., 1989.

[Ret93] Christian Retoré. *Réseaux et Séquents Ordonnés*. PhD thesis, Université Paris VII, 1993.

[Soc91] Rolf Socher-Ambrosius. Boolean algebra admits no convergent term rewriting system. In *Rewriting Techniques and Applications, 4th International Conference, RTA-91, Como, Italy, April 10-12, 1991, Proceedings*, pages 264–274, 1991.

[Str07a] Lutz Straßburger. A characterisation of medial as rewriting rule. In Franz Baader, editor, *RTA 2007*, volume 4533 of *LNCS*, pages 344–358. Springer-Verlag, 2007.

[Str07b] Lutz Straßburger. On the axiomatisation of Boolean categories with and without medial. *Theory and Applications of Categories*, 18(18):536–601, 2007.

[Str12] Lutz Straßburger. Extension without cut. *Ann. Pure Appl. Logic*, 163(12):1995–2007, 2012.

[Ter03] Terese. *Term rewriting systems*. Cambridge University Press, 2003.

[Val84] L. G. Valiant. Short monotone formulae for the majority function. *Journal of Algorithms*, 5(3):363 – 366, 1984.