A General View of Normalisation through Atomic Flows

Tom Gundersen

A General View of Normalisation through Atomic Flows

submitted by

Tom Erik Gundersen

for the degree of Doctor of Philosophy

of the

University of Bath

Department of Computer Science

August 2009

A General View of Normalisation through Atomic Flows

Tom Erik Gundersen

ABSTRACT

Atomic flows are a geometric invariant of classical propositional proofs in deep inference. In this thesis we use atomic flows to describe new normal forms of proofs, of which the traditional normal forms are special cases, we also give several normalisation procedures for obtaining the normal forms. We define, and use to present our results, a new deep-inference formalism called *the functorial calculus*, which is more flexible than the traditional calculus of structures. To our surprise we are able to 1) normalise proofs without looking at their logical connectives or logical rules; and 2) normalise proofs in less than exponential time.

ACKNOWLEDGEMENTS

I was first introduced to proof theory by my supervisor, Alessio Guglielmi, and for this I suppose I ought to be grateful. Should anyone reading this be considering a PhD in proof theory, then I urge you to give Alessio a call, as he truly is the most wonderful supervisor.

Furthermore, I wish to thank Alessio and Paola Bruscoli for looking after me when I moved to France. You have both been like mothers to me.

I would also like to express my gratitude to all my colleagues, too many to name, who have been influential in the shaping of this thesis. Two people deserve special mention: Alessio, for teaching me what I know, and my examiner, François Lamarche, for his thorough and insightful comments.

Finally, I wish to thank my friends and family for providing me with an endless source of entertainment throughout the course of my studies. In particular, I want to thank my husband John for his love and support. If some of my sanity remains, it is thanks to you.

Contents

Chapter 1

Introduction

Structural proof theory is the subdiscipline of logic that studies formal representation and manipulation of mathematical proofs.

A language for representing proofs is called a *formalism*. Traditionally, formalisms are variations of Gentzen's *natural deduction* and *sequent calculus* [Gen69]. Essentially, a formalism following Gentzen's methodology represents a proof as a tree, obtained by recursively breaking formulae apart at their main connective.

The rules by which proofs are constructed are called *inference rules*. A logic is represented in a given formalism by a set of inference rules, called a *logical system*.

Deep inference [Gug07] is a methodology that allows generalisations of Gentzen's formalisms. The standard deep-inference formalism, the *calculus of structures*, generalises the sequent calculus by allowing deduction at any place in a formula, rather than restricting it to the main connective. As a consequence, it is possible for all inference rules to be unary. In other words, proofs are represented as lists of formulae rather than as trees of sequents.

In this thesis, a new deep-inference formalism, named the *functorial calculus*, is presented. While, in the sequent calculus, the juxtaposition of two proofs denotes that they are composed by a conjunction, in the functorial calculus, this horizontal composition is generalised to allow both disjunctions and conjunctions. In other words, proofs are represented as directed acyclic graphs of formulae rather than as trees of sequents.

The calculus of structures and the functorial calculus are closely related and translations between the two are given. The relationship between the two formalisms is explored further in [GGP10], where a generalisation, called *open deduction*, is presented. It is shown there that a functorial calculus proof corresponds to an equivalence class of calculus of structures proofs.

The functorial calculus was chosen for this thesis, rather than the calculus of structures, for two reasons. Firstly, the smaller proofs and fewer arbitrary choices required by the functorial calculus simplifies the presentation of the results. Secondly, some of the results of this thesis have been presented elsewhere in terms of the calculus of structures [GG08, BGGP10,

GGS10], so using the functorial calculus illustrates the fact that the results are not tightly coupled to a specific formalism.

The focus of this thesis is propositional classical logic. By exploiting the symmetry available in deep-inference, it is possible to represent propositional classical logic in a system where every inference rule belongs to one of two kinds: *atomic* or *linear* [BT01].

An inference rule is *linear* if, for every instance of the rule, there is a one-to-one correspondence between the atom occurrences in the premiss and the atom occurrences in the conclusion. Linear inference rules increases the flexibility of proofs, as other inference rule instances can in most cases trivially be permuted 'through' the linear ones.

The *atomic* inference rules are rules where only a given atom or its dual occur in every instance. By replacing a generic inference rule with several atomic ones, the flexibility of the proof is increased as the different atomic rules can be permuted independently from each other.

The possibility, which is not present in the sequent calculus [Brü03b], of having only linear and atomic inference rules allows representations of proofs which are extremely 'malleable'.

The first part of this thesis will introduce classical logic in the functorial calculus, show the relationship between the functorial calculus and the calculus of structures, and present some standard deep-inference results.

A formalism usually comes with a *normalisation theory*, *i.e.* a notion of *normal form* of proofs as well as a procedure describing how to manipulate proofs in order to obtain their normal form. In natural deduction a proof is in normal form if no 'elimination rule' follows an 'introduction rule'; and in the sequent calculus a proof is in normal form if it does not contain the *cut rule*.

The cut rule, also known as *modus ponens*, is at the heart of proof theory. The cut rule allows an auxiliary result to be proven only once, but used many times. When viewing proofs as programs, the cut is the application of a function to an argument, and normalisation is computation.

As in the sequent calculus, the cut rule is admissible from deep-inference proofs without a premiss. In [Brü04], Brünnler presents a cut-elimination procedure for the calculus of structures and studies the connection between proofs with and without cut in the calculus of structures and in the sequent calculus.

The fact that the sequent calculus represent proofs as trees makes it inherently asymmetric in the horizontal axis. This asymmetry is not present in the calculus of structures or the functorial calculus. In fact, an asymmetry has to be enforced for the cut rule to be admissible.

The symmetry that is possible in deep-inference formalisms allows more notions of normal forms than just cut elimination. In particular, the dual of cut elimination also holds: axioms can be eliminated from proofs of falsehood.

In this thesis a new notion of normal form of propositional classical logic proofs, called

streamlining is introduced. Unlike cut or axiom elimination, streamlining applies to all deep-inference proofs, and in the asymmetric case where cut or axiom elimination is applicable, the notions coincide. Unlike normal forms based on the order of inference rule instances, streamlining is invariant under rule permutations. Furthermore, streamlining is a largely syntax independent notion, in the sense that it is not tied to a specific formalism, or a specific logical system.

In order to describe the notion of streamlining and the related normal forms, we introduce a proof invariant that we call *atomic flows*. Atomic flows are certain kinds of directed acyclic graphs that capture the structural information of proofs. Intuitively, an atomic flow is obtained from a proof by retaining the causal dependencies between creation, duplication and destruction of atoms and discarding all information about logical connectives, units and linear inference rules. A proof is streamlined if there is no path in its atomic flow from the creation to the destruction of an atom.

The second part of this thesis is devoted to atomic flows, their relationship with proofs and the definition of normal forms in terms of atomic flows.

Atomic flows were designed to describe normal form of proofs. However, it turns out that atomic flows are also a very convenient tool for designing and arguing about normalisation procedures. In the third part of this thesis two kinds of normalisation procedures are given. All the procedures are first presented in terms of atomic flows, before they are lifted to derivations.

The *global* procedures work by making several copies of an entire atomic flow, 'pruning' each copy and 'stitching' them together. Three different global procedures are presented, all producing derivations in the same normal form. It appears that there is great flexibility in the design of the global procedures and there is a lot of room for future investigations, especially with respect to complexity. We show that the global procedures can have less than exponential cost. However, they are all inherently non-confluent.

Whereas the global procedures consider the whole atomic flow, the *local* procedures work on one pair of adjacent vertices. These procedures are confluent, but their cost is inherently exponential.

It is expected that propositional classical logic normalisation is inherently exponential and non-confluent, and in fact we observe both these phenomena. However, they are separated into two distinct phases, which can be studied independently. It is worth noting that cut elimination is achieved with less than exponential cost.

The main contribution of this thesis is the use of atomic flows for arguing about normalisation. While it is true that all the results could be reformulated in terms of derivations, this would only serve to obfuscate what is going on.

It should be noted that all the important properties of normalisation can be proven in terms of atomic flows alone. In particular results about complexity, termination, confluence and correctness can be proven without reference to derivations. The challenge in designing normalisation procedures is finding the correct atomic flow transformation, verifying that a transformation can be lifted to derivations is always straight forward.

There are two reasons to consider flows to describe the essence of proofs from the point of view of normalisation: Firstly, the flow of a proof determines how the proof can be normalised. Secondly, isomorphisms between atomic flows are preserved by normalisation. That is, the results of normalising two proofs with isomorphic atomic flows have isomorphic atomic flows.

With respect to future work, two aspects of normalisation are especially relevant to this thesis: *bureaucracy* and *complexity*.

The complexity of cut elimination in the sequent calculus is known to be exponential [Sta78] and it is known that cut elimination has less than exponential cost in deep inference [Jeř09], however no lower bound exists. Furthermore, this thesis presents normal forms for which only exponential cost normalisation procedures are known. A possible direction of future work is to establish atomic flows as a tool for studying complexity, and to discover new normalisation procedures with lower complexity bounds.

The term *bureaucracy* was coined by Girard to denote arbitrary syntactic dependencies in proofs. The presence of bureaucracy means that proofs that are 'essentially the same' do not have a common canonical representation. Since all known formalisms have some degree of bureaucracy, an important aspect of any normalisation procedures is how it behaves with respect to bureaucracy. A desirable property is that, if two proofs are the same modulo bureaucracy, they have the same normal forms modulo bureaucracy. For the procedures presented in this thesis, this property always holds for notions of bureaucracy captured by atomic flows. Hence, another possible direction of future work is to show what notions of bureaucracy atomic flows capture and to adapt atomic flows to capture more notions of bureaucracy.

Part I

Derivations

Chapter 2

Propositional Classical Logic

The traditional formalism in deep inference is *the calculus of structures* [Gug07].

The idea of a new formalism, named *formalism A* based on the calculus of structures, but where derivations contain less bureaucracy, was proposed by Guglielmi in [Gug04], and later Brünnler and Lengrand developed a term calculus around these ideas [BL05].

In this chapter I define a formalism based on the ideas of formalism A and call it (as suggested by François Lamarche) *the functorial calculus*. The reason to introduce a new formalism is that it greatly simplifies the presentation of some of the more technical results in this thesis (in particular Section 6.4.1 on page 65).

After presenting the functorial calculus we compare it briefly with the calculus of structures before we introduce the standard deductive system for classical logic in deep inference and show some preliminary results.

We now define 'formulae' and 'inference rules', which are in common between both the functorial calculus and the calculus of structures. Definitions 2.0.1 to 2.0.4 on pages 6–7 are based on definitions given in [BG09]. The focus of this thesis is classical propositional logic, and the following definitions reflect this. However, it is worth noting that the definitions can be generalised to other units and connectives, if one wants to present other logics.

Definition 2.0.1. We define a set of *formulae*, denoted by α, β, γ, δ to be:

- *atoms*, denoted by a, b, c, d and $\bar{a}$, $\bar{b}$, $\bar{c}$, $\bar{d}$;
- *formula variables*, denoted by A, B, C, D;
- *units* f (false) and t (true); and
- the *disjunction* and *conjunction* of formulae α and β, denoted by $[\alpha \vee \beta]$ and $(\alpha \wedge \beta)$, respectively.

A formula is *ground* if it contains no variables. We usually omit external brackets of formulae, and sometimes we omit dispensable brackets under associativity. We use $\equiv$ to denote literal

equality of formulae. The *size* $|\alpha|$ of a formula α is the number of unit, atom and variable occurrences appearing in it. On the set of atoms there is an involution $\bar{\cdot}$, called *negation* (*i.e.*, $\bar{\cdot}$ is a bijection from the set of atoms to itself such that $\bar{\bar{a}} \equiv a$); we require that $\bar{a} \not\equiv a$ for every a; when both a and $\bar{a}$ appear in a formula, we mean that atom a is mapped to by $\bar{a}$ by $\bar{\cdot}$. A *context* is a formula where one *hole* $\{\ \}$ appears in the place of a subformula; for example, $a \vee (b \wedge \{\ \})$ is a context; the generic context is denoted by $\xi\{\ \}$. The hole can be filled with formulae; for example, if $\xi\{\ \} \equiv b \wedge [\{\ \} \vee c]$, then $\xi\{a\} \equiv b \wedge [a \vee c]$, $\xi\{b\} \equiv b \wedge [b \vee c]$ and $\xi\{a \wedge b\} \equiv b \wedge [(a \wedge b) \vee c]$. The *size* of $\xi\{\ \}$ is defined as $|\xi\{\ \}| = |\xi\{a\}| - 1$.

Definition 2.0.2. A *renaming* is a map from the set of atoms to itself, and it is denoted by $\{a_1/b_1, a_2/b_2, \dots\}$. A renaming of α by $\{a_1/b_1, a_2/b_2, \dots\}$ is indicated by $\alpha\{a_1/b_1, a_2/b_2, \dots\}$ and is obtained by simultaneously substituting every occurrence of a_i in α by b_i and every occurrence of $\bar{a}_i$ by $\bar{b}_i$; for example, if $\alpha \equiv a \wedge [b \vee (a \wedge [\bar{a} \vee c])]$ then $\alpha\{a/\bar{b}, \bar{b}/c\} \equiv \bar{b} \wedge \left[\bar{c} \vee \left(\bar{b} \wedge [b \vee c]\right)\right]$. A *substitution* is a map from the set of formula variables to the set of formulae, denoted by $\{A_1/\beta_1, A_2/\beta_2, \dots\}$. An *instance* of α by $\{A_1/\beta_1, A_2/\beta_2, \dots\}$ is indicated by $\alpha\{A_1/\beta_1, A_2/\beta_2, \dots\}$ and is obtained by simultaneously substituting every occurrence of variable A_i in α by formula β_i; for example if $\alpha \equiv A \vee (b \wedge c)$ then $\alpha\{A/\left(c \wedge \bar{b}\right)\} \equiv \left(c \wedge \bar{b}\right) \vee (b \wedge c)$.

Convention 2.0.3. By the above definition, formula variables will only be used to define inference rules, and will never appear in derivations. However, when we perform normalisation we will sometimes single out atom occurrences (by decorating them) and substitute on them as if they were formula variables.

Definition 2.0.4. An *inference rule* ρ is an expression $\rho\dfrac{\alpha}{\beta}$, where the formulae α and β are called *premiss* and *conclusion*, respectively. A (*deductive*) *system* is a finite set of inference rules. An *inference rule instance* $\rho\dfrac{\gamma}{\delta}$ of $\rho\dfrac{\alpha}{\beta}$ is such that γ and δ are ground, and $\gamma \equiv \alpha\{a_1/b_1, a_2/b_2, \dots\}\{A_1/\beta_1, A_2/\beta_2, \dots\}$ and $\delta \equiv \beta\{a_1/b_1, a_2/b_2, \dots\}\{A_1/\beta_1, A_2/\beta_2, \dots\}$, for some renaming $\{a_1/b_1, a_2/b_2, \dots\}$ and substiution $\{A_1/\beta_1, A_2/\beta_2, \dots\}$.

2.1 The Functorial Calculus

We now present the functorial calculus in the context of classical propositional logic and give some basic results.

The intuition behind derivations in the functorial calculus is that we can compose derivations by the same connectives we can compose formulae.

Definition 2.1.1. Given a deductive system $\mathscr{S}$, and formulae α and β; a (*functorial calculus*) *derivation* Ψ *in* $\mathscr{S}$ *from* α *to* β, denoted $\Psi \Vdash_{\mathscr{S}} \dfrac{\alpha}{\beta}$, is defined to be

1. a formula: $\Psi = \alpha \equiv \beta$;

2. a *vertical composition*:

$$\Psi = \begin{array}{c} \alpha \\ \Phi_1 \| \\ \rho \dfrac{\beta'}{\alpha'} \\ \Phi_2 \| \\ \beta \end{array} ,$$

where $\rho \dfrac{\beta'}{\alpha'}$ is an instance of an inference rule from $\mathscr{S}$, and $\begin{array}{c} \alpha \\ \Phi_1 \| \\ \beta' \end{array} \mathscr{S}$ and $\begin{array}{c} \alpha' \\ \Phi_2 \| \\ \beta \end{array} \mathscr{S}$ are derivations; or

3. a *horizontal composition*:

$$\Psi = \begin{array}{c} \alpha_1 \\ \Phi_1 \| \\ \beta_1 \end{array} \wedge \begin{array}{c} \alpha_2 \\ \Phi_2 \| \\ \beta_2 \end{array} \qquad \text{or} \qquad \Psi = \begin{array}{c} \alpha_1 \\ \Phi_1 \| \\ \beta_1 \end{array} \vee \begin{array}{c} \alpha_2 \\ \Phi_2 \| \\ \beta_2 \end{array} ,$$

where $\begin{array}{c} \alpha_1 \\ \Phi_1 \| \\ \beta_1 \end{array}$ and $\begin{array}{c} \alpha_2 \\ \Phi_2 \| \\ \beta_2 \end{array}$ are derivations, and $\alpha \equiv \alpha_1 \vee \alpha_2$ and $\beta \equiv \beta_1 \vee \beta_2$, or $\alpha \equiv \alpha_1 \wedge \alpha_2$ and $\beta \equiv \beta_1 \wedge \beta_2$, respectively.

A derivation with premiss t is, from now on, called a *proof*.

The size of a derivation Ψ, denoted $|\Psi|$, is defined to be the sum of the size of the formulae appearing in Ψ.

Convention 2.1.2. Given derivations $\begin{array}{c} \alpha_1 \\ \Phi_1 \| \\ \beta_1 \end{array}$, $\begin{array}{c} \alpha_2 \\ \Phi_2 \| \\ \beta_2 \end{array}$ and $\begin{array}{c} \alpha_3 \\ \Phi_3 \| \\ \beta_3 \end{array}$, and inference rule instances $\rho_1 \dfrac{\beta_1}{\alpha_2}$ and $\rho_2 \dfrac{\beta_2}{\alpha_3}$ we consider

$$\rho_2 \dfrac{\left(\begin{array}{c} \alpha_1 \\ \Phi_1 \| \\ \rho_1 \dfrac{\beta_1}{\alpha_2} \\ \Phi_2 \| \\ \beta_2 \end{array} \right)}{\begin{array}{c} \alpha_3 \\ \Phi_3 \| \\ \beta_3 \end{array}} \qquad \text{and} \qquad \rho_1 \dfrac{\begin{array}{c} \alpha_1 \\ \Phi_1 \| \\ \beta_1 \end{array}}{\left(\begin{array}{c} \alpha_2 \\ \Phi_2 \| \\ \rho_2 \dfrac{\beta_2}{\alpha_3} \\ \Phi_3 \| \\ \beta_3 \end{array} \right)}$$

to be equal, and we denote them both by

$$\begin{array}{l} \alpha_1 \\ \Phi_1 \| \\ \rho_1 \dfrac{\beta_1}{\alpha_2} \\ \Phi_2 \| \\ \rho_2 \dfrac{\beta_2}{\alpha_3} \\ \Phi_3 \| \\ \beta_3 \end{array} \quad .$$

Remark 2.1.3. If desireable, Convention 2.1.2 on the preceding page could be made redundant by forcing associativity of horizontal composition in Definition 2.1.1 on page 7. The only reason we did not do this was for the sake of brevity of the following results.

Lemma 2.1.4. *Given a derivation* $\begin{array}{l}\alpha\\ \Phi\|\\ \beta\end{array}$ *and a context* $\xi\{\ \}$, *a derivation* $\begin{array}{l}\xi\{\alpha\}\\ \Psi\|\\ \xi\{\beta\}\end{array}$, *with size* $|\Phi| + |\xi\{\ \}|$, *can be constructed.*

Proof. We proceed by structural induction on $\xi\{\ \}$. The base case, $\xi\{\ \} \equiv \{\ \}$, is trivial. For the inductive case, let

$$\xi\{\ \} \equiv \xi'\{\ \} \wedge \gamma \quad , \qquad \xi\{\ \} \equiv \gamma \wedge \xi'\{\ \} \quad ,$$

$$\xi\{\ \} \equiv \xi'\{\ \} \vee \gamma \qquad \text{or} \qquad \xi\{\ \} \equiv \gamma \vee \xi'\{\ \} \quad .$$

for some formula γ and a context $\xi'\{\ \}$. By the inductive hypothesis we can construct the derivation $\begin{array}{l}\xi'\{\alpha\}\\ \Psi'\|\\ \xi'\{\beta\}\end{array}$, so the result follows by case (3) of Definition 2.1.1 on page 7. □

Notation 2.1.5. Given a derivation $\begin{array}{l}\alpha\\ \Phi\|\\ \beta\end{array}$ and a context $\xi\{\ \}$, the derivation $\begin{array}{l}\xi\{\alpha\}\\ \|\\ \xi\{\beta\}\end{array}$ constructed in the proof of Lemma 2.1.4 is denoted $\xi\{\Phi\}$.

Lemma 2.1.6. *Given two derivations* $\begin{array}{l}\alpha\\ \Phi_1\|\\ \beta\end{array}$ *and* $\begin{array}{l}\beta\\ \Phi_2\|\\ \gamma\end{array}$, *a derivation* $\begin{array}{l}\alpha\\ \Psi\|\\ \gamma\end{array}$, *with size* $|\Phi_1| + |\Phi_2| - |\beta|$, *can be constructed.*

Proof. We argue by structural induction on Φ_1

1. if $\Phi_1 = \beta$ then $\Psi = \Phi_2$, with size $|\Phi_1| + |\Phi_2| - |\beta|$;

2. if

$$\Phi_1 = \rho\dfrac{\begin{array}{c}\alpha\\ \Phi_1'\|\end{array}}{\begin{array}{c}\beta'\\ \alpha'\end{array}}\dfrac{\begin{array}{c}\\ \Phi_1''\|\end{array}}{\beta},$$

then, by the inductive hypothesis, we can construct $\begin{array}{c}\alpha'\\ \Psi'\|\\ \gamma\end{array}$, with size $\left|\Phi_1''\right|+|\Phi_2|-|\beta|$, we can then build

$$\Psi = \rho\dfrac{\begin{array}{c}\alpha\\ \Phi_1'\|\end{array}}{\begin{array}{c}\beta'\\ \alpha'\end{array}}\dfrac{\begin{array}{c}\\ \Psi'\|\end{array}}{\gamma},$$

with size $\left|\Phi_1'\right|+\left|\Psi'\right|=\left|\Phi_1'\right|+\left|\Phi_1''\right|+|\Phi_2|-|\beta|=|\Phi_1|+|\Phi_2|-|\beta|$;

3. if

$$\Phi_1 = \begin{array}{c}\alpha_1\\ \Phi_{1,1}\|\\ \beta_1\end{array} \vee \begin{array}{c}\alpha_2\\ \Phi_{1,2}\|\\ \beta_2\end{array} \quad \text{or} \quad \Phi_1 = \begin{array}{c}\alpha_1\\ \Phi_{1,1}\|\\ \beta_1\end{array} \wedge \begin{array}{c}\alpha_2\\ \Phi_{1,2}\|\\ \beta_2\end{array}$$

we argue by structural induction on Φ_2:

   (a) if Φ_2 is a formula (resp., a vertical composition), the result follow by a symmetric argument to case 1 (resp., 2) above.

   (b) if

$$\Phi_2 = \begin{array}{c}\beta_1\\ \Phi_{2,1}\|\\ \gamma_1\end{array} \vee \begin{array}{c}\beta_2\\ \Phi_{2,2}\|\\ \gamma_2\end{array} \quad \text{or} \quad \Phi_2 = \begin{array}{c}\beta_1\\ \Phi_{2,1}\|\\ \gamma_1\end{array} \wedge \begin{array}{c}\beta_2\\ \Phi_{2,2}\|\\ \gamma_2\end{array}$$

   then, by the first inductive hypothesis, we can construct

$$\begin{array}{c}\alpha_1\\ \Psi_1\|\\ \gamma_1\end{array} \quad \text{and} \quad \begin{array}{c}\alpha_2\\ \Psi_2\|\\ \gamma_2\end{array},$$

   with size $\left|\Phi_{1,1}\right|+\left|\Phi_{2,1}\right|-|\beta_1|$ and $\left|\Phi_{1,2}\right|+\left|\Phi_{2,2}\right|-|\beta_2|$, respectively, we can then build

$$\Psi = \begin{array}{c}\alpha_1\\ \Psi_1\|\\ \gamma_1\end{array} \vee \begin{array}{c}\alpha_2\\ \Psi_2\|\\ \gamma_2\end{array} \quad \text{or} \quad \Psi = \begin{array}{c}\alpha_1\\ \Psi_1\|\\ \gamma_1\end{array} \wedge \begin{array}{c}\alpha_2\\ \Psi_2\|\\ \gamma_2\end{array}$$

   with size $|\Psi_1|+|\Psi_2|=\left|\Phi_{1,1}\right|+\left|\Phi_{1,2}\right|+\left|\Phi_{2,1}\right|+\left|\Phi_{2,2}\right|-(|\beta_1|+|\beta_2|)=|\Phi_1|+|\Phi_2|-|\beta|$.

□

Notation 2.1.7. Given derivations $\begin{matrix}\alpha\\ \Phi_1\|\\ \beta\end{matrix}$ and $\begin{matrix}\beta\\ \Phi_2\|\\ \gamma\end{matrix}$, the derivation $\begin{matrix}\alpha\\ \Psi\|\\ \gamma\end{matrix}$ constructed in the proof of Lemma 2.1.6 on page 9 is denoted:

$$\begin{matrix}\alpha\\ \Phi_1\|\\ \beta\\ \Phi_2\|\\ \gamma\end{matrix} .$$

2.2 The Calculus of Structures

We now present the calculus of structures and in Theorem 2.2.2 and Theorem 2.2.6 on page 13 we show that the functorial calculus and the calculus of structures polynomially simulate each other.

The intuition behind derivations in the calculus of structures is that we rewrite formulae by applying inference rules inside a context.

Definition 2.2.1. Given a deductive system $\mathscr{S}$, a set of formulae, $\mathscr{F}$, and α and β from $\mathscr{F}$; a *calculus of structures derivation* Ψ *in* $\mathscr{S}$ *from* α *to* β, denoted $\begin{matrix}\alpha\\ \Psi\|_{\mathscr{S}}\\ \beta\end{matrix}$, is defined to be

1. a formula: $\Psi = \alpha \equiv \beta$; or

2. a *vertical composition*:

$$\Psi = \begin{matrix}\alpha\\ \Phi_1\|\\ \xi\{\beta'\}\end{matrix} \rho \frac{\xi\{\beta'\}}{\xi\{\alpha'\}} \begin{matrix}\xi\{\alpha'\}\\ \Phi_2\|\\ \beta\end{matrix} ,$$

   where $\rho\dfrac{\beta'}{\alpha'}$ is an instance of an inference rule from $\mathscr{S}$, and $\begin{matrix}\alpha\\ \Phi_1\|_{\mathscr{S}}\\ \xi\{\beta'\}\end{matrix}$ and $\begin{matrix}\xi\{\alpha'\}\\ \Phi_2\|_{\mathscr{S}}\\ \beta\end{matrix}$ are calculus of structures derivations.

The size of a calculus of structures derivation Ψ, denoted $|\Psi|$, is defined to be the sum of the size of the formulae appearing in Ψ.

Theorem 2.2.2. *A calculus of structures derivation* $\Phi \begin{Vmatrix} \alpha \\ \beta \end{Vmatrix}$ *can be transformed into a functorial calculus derivation* $\Psi \begin{Vmatrix} \alpha \\ \beta \end{Vmatrix}$ *such that* $|\Psi| \leqslant |\Phi|$.

Proof. We argue by structural induction on Φ. The base case is trivial; $\Phi = \alpha \equiv \beta = \Psi$. For the inductive case, consider the following calculus of structures derivation:

$$\Phi = \rho \dfrac{\begin{matrix} \Phi_1 \| \alpha \\ \xi\{\beta'\} \end{matrix}}{\begin{matrix} \xi\{\alpha'\} \\ \Phi_2 \| \beta \end{matrix}} \quad .$$

By the inductive hypothesis, there are functorial calculus derivations $\Psi_1 \begin{Vmatrix} \alpha \\ \xi\{\beta'\} \end{Vmatrix}$ and $\Psi_2 \begin{Vmatrix} \xi\{\alpha'\} \\ \beta \end{Vmatrix}$, such that $|\Psi_1| \leqslant |\Phi_1|$ and $|\Psi_2| \leqslant |\Phi_2|$. By Lemma 2.1.4 on page 9, there is a functorial calculus derivation $\xi \left\{ \rho \dfrac{\beta'}{\alpha'} \right\}$, with size $|\xi\{\ \}| + |\alpha'| + |\beta'|$. By Lemma 2.1.6 on page 9, we can combine the three functorial calculus derivations to create $\Psi \begin{Vmatrix} \alpha \\ \beta \end{Vmatrix}$, with size $|\Psi_1| + |\Psi_2| + |\xi\{\ \}| + |\beta'| + |\alpha'| - |\xi\{\ \}| - |\beta'| - |\xi\{\ \}| - |\alpha'| = |\Psi_1| + |\Psi_2| - |\xi\{\ \}| \leqslant |\Phi_1| + |\Phi_2| = |\Phi|$. □

Example 2.2.3. Figure 4-1 on page 30 has three examples of calculus of structures derivations transformed into functorial calculus derivations.

Lemma 2.2.4. *Given a calculus of structures derivation* $\Phi \begin{Vmatrix} \alpha \\ \beta \end{Vmatrix}$ *and a context* $\xi\{\ \}$*, a calculus of structures derivation* $\Psi \begin{Vmatrix} \xi\{\alpha\} \\ \xi\{\beta\} \end{Vmatrix}$ *can be constructed, such that the number of inference rule instances in* Ψ *is the same as the number of inference rule instances in* Φ*, and the size of the largest formula in* Ψ *is the sum of the largest formula in* Φ *and* $|\xi\{\ \}|$.

Proof. The statements follows by structural induction on Φ. □

Lemma 2.2.5. *Given two calculus of structures derivations* $\Phi_1 \begin{Vmatrix} \alpha \\ \beta \end{Vmatrix}$ *and* $\Phi_2 \begin{Vmatrix} \beta \\ \gamma \end{Vmatrix}$*, a calculus of structures derivation* $\Psi \begin{Vmatrix} \alpha \\ \gamma \end{Vmatrix}$ *can be constructed, such that the number of inference rule instances in* Ψ *is the sum*

of the number of inference rule instances in Φ_1 and Φ_2 combined, and the largest formula in Ψ is the largest formula of Φ_1 or the largest formula of Φ_2.

Proof. The statement follows by structural induction on Φ_1. □

Theorem 2.2.6. *A functorial calculus derivation* $\begin{matrix}\alpha\\ \Phi\|\\ \beta\end{matrix}$ *can be transformed into a calculus of structures derivation* $\begin{matrix}\alpha\\ \Psi\|\\ \beta\end{matrix}$ *such that the size of Ψ depends at most quadratically on the size of Φ.*

Proof. We first show how to construct Ψ: The base cases, when Φ is a formula or a vertical composition, are trivial. For the inductive case, consider a conjunction of functorial calculus derivations (the argument for the disjunction is similar):

$$\Phi = \begin{matrix}\alpha_1\\ \Phi_1\|\\ \beta_1\end{matrix} \wedge \begin{matrix}\alpha_2\\ \Phi_2\|\\ \beta_2\end{matrix} .$$

By the inductive hypothesis and Lemma 2.2.4 on the preceding page there are calculus of structures derivations

$$\begin{matrix}\alpha_1 \wedge \alpha_1\\ \Psi_1\|\\ \beta_1 \wedge \alpha_1\end{matrix} \quad \text{and} \quad \begin{matrix}\beta_1 \wedge \alpha_2\\ \Psi_2\|\\ \beta_1 \wedge \beta_2\end{matrix} ,$$

and by Lemma 2.2.5 there exists a calculus of structures derivation $\begin{matrix}\alpha_1 \wedge \alpha_2\\ \Psi\|\\ \beta_1 \wedge \beta_2\end{matrix}$.

To find an upper bound on the size of Ψ, we observe that it depends at most linearly on the number of inference rule instances in Ψ multiplied by the size of the largest formula in Ψ. Furthermore, by the above Lemmata, the number of inference rules in Ψ is the same as the number of inference rules in Φ and the size of the largest inference rule depends at most linearly on the size of Φ, so the size of Ψ depends at most quadratically on the size of Φ. □

The calculus of structures is now well developed for classical [Brü03a, Brü06a, Brü06d, BT01, Brü06b], intuitionistic [Tiu06a], linear [Str02, Str03b], modal [Brü06c, GT07, Sto07] and commutative/non-commutative logics [Gug07, Tiu06b, Str03a, Bru02, DG04, GS01, GS02, GS09, Kah06, Kah07]. The basic proof complexity properties of the calculus of structures are known [BG09]. The calculus of structures promoted the discovery of a new class of proof nets for classical and linear logic [LS05a, LS05b, LS06, SL04] (see also [Gui06]). There exist implementations in Maude of deep-inference proof systems [Kah08].

2.3 System SKS

We now define the standard deductive system SKS for classical propositional logic in deep inference [Brü03a, Brü06a, Brü06d, BT01]. For an excellent reference to previous work on normalisation in SKS, see [Brü04]. Subsystems of SKS are used throughout this thesis.

The results presented in this section, with the exception of Theorem 2.3.14 on page 18, are standard results which can be found in the literature. We include the proofs for completeness and as means for giving examples of the functorial calculus.

Definition 2.3.1. System SKS is defined by the following *structural* inference rules:

$$\mathsf{ai}{\downarrow}\frac{\mathsf{t}}{a \vee \bar{a}} \qquad \mathsf{aw}{\downarrow}\frac{\mathsf{f}}{a} \qquad \mathsf{ac}{\downarrow}\frac{a \vee a}{a}$$

$$\mathsf{ai}{\uparrow}\frac{a \wedge \bar{a}}{\mathsf{f}} \qquad \mathsf{aw}{\uparrow}\frac{a}{\mathsf{t}} \qquad \mathsf{ac}{\uparrow}\frac{a}{a \wedge a} \quad ,$$

the *logical* inference rules:

$$\mathsf{s}\frac{A \wedge [B \vee C]}{(A \wedge B) \vee C} \qquad \mathsf{m}\frac{(A \wedge B) \vee (C \wedge D)}{[A \vee C] \wedge [B \vee D]} \quad ,$$

and the *invertible* (*logical*) rules:

$$=_{\vee\mathsf{c}}\frac{A \vee B}{B \vee A} \qquad =_{\wedge\mathsf{c}}\frac{A \wedge B}{B \wedge A} \qquad =_{\mathsf{a}}{\downarrow}\frac{A \vee [B \vee C]}{[A \vee B] \vee C} \qquad =_{\mathsf{a}}{\uparrow}\frac{(A \wedge B) \wedge C}{A \wedge (B \wedge C)}$$

$$=_{\mathsf{f}}{\downarrow}\frac{A}{A \vee \mathsf{f}} \qquad =_{\mathsf{t}}{\downarrow}\frac{A}{A \wedge \mathsf{t}} \qquad =_{\mathsf{f}}{\uparrow}\frac{\mathsf{t} \wedge A}{A} \qquad =_{\mathsf{t}}{\uparrow}\frac{\mathsf{f} \vee A}{A}$$

$$=_{\mathsf{f}\wedge}{\downarrow}\frac{\mathsf{f}}{\mathsf{f} \wedge \mathsf{f}} \qquad =_{\mathsf{t}\vee}{\downarrow}\frac{\mathsf{t}}{\mathsf{t} \vee \mathsf{t}} \qquad =_{\mathsf{f}\wedge}{\uparrow}\frac{\mathsf{t} \vee \mathsf{t}}{\mathsf{t}} \qquad =_{\mathsf{t}\vee}{\uparrow}\frac{\mathsf{f} \wedge \mathsf{f}}{\mathsf{f}} \quad .$$

The calculus of structures and system SKS were originally defined in terms of equivalence classes of formulae, called 'structures', and without the above invertible logical rules. However, we find it more convenient to use formulae instead, since it makes it simpler to 'trace atom occurrences', which we will see in Section 4.1 on page 28. We now show that the two approaches are morally the same.

Definition 2.3.2. We define the relation $=$ such that, given formulae α and β, $\alpha = \beta$ if there is a derivation $\Phi \Vdash \frac{\alpha}{\beta}$ in $\{=_{\vee\mathsf{c}}, =_{\wedge\mathsf{c}}, =_{\mathsf{a}}{\downarrow}, =_{\mathsf{a}}{\uparrow}, =_{\mathsf{f}}{\downarrow}, =_{\mathsf{t}}{\downarrow}, =_{\mathsf{f}}{\uparrow}, =_{\mathsf{t}}{\uparrow}, =_{\mathsf{f}\wedge}{\downarrow}, =_{\mathsf{t}\vee}{\downarrow}, =_{\mathsf{f}\wedge}{\uparrow}, =_{\mathsf{t}\vee}{\uparrow}\}$.

Notation 2.3.3. If $\alpha = \beta$, we often write $= \frac{\alpha}{\beta}$.

Remark 2.3.4. By Notation 2.3.3 on the facing page and Lemma 2.1.4 on page 9, for any formulae α and β and context $\xi\{\ \}$ we have that $\alpha = \beta$ implies $\xi\{\alpha\} = \xi\{\beta\}$.

Proposition 2.3.5. *The relation $=$ defined in Definition 2.3.2 on the facing page is an equivalence relation.*

It turns out that the equivalence class induced by $=$ is the same as the structures used in [Brü04].

Remark 2.3.6. If $\alpha = \beta$, then (as remarked in [BG09]) there exists a derivation

$$\Phi \Vdash \begin{matrix} \alpha \\ \beta \end{matrix} \{=_{\mathsf{Vc}}, =_{\wedge\mathsf{c}}, =_{\mathsf{a}}\downarrow, =_{\mathsf{a}}\uparrow, =_{\mathsf{f}}\downarrow, =_{\mathsf{t}}\downarrow, =_{\mathsf{f}}\uparrow, =_{\mathsf{t}}\uparrow, =_{\mathsf{f}\wedge}\downarrow, =_{\mathsf{t}\vee}\downarrow, =_{\mathsf{f}\wedge}\uparrow, =_{\mathsf{t}\vee}\uparrow\}$$

whose size depends at most quadratically on the sum of the sizes of α and β.

Notation 2.3.7. When we work in (subsystems of) SKS, we often omit mentioning the invertible rules. Given $\mathscr{S}$ be a subsystem of SKS, then, unless specified otherwise, when we write $\mathscr{S}$ we mean $\mathscr{S} \cup \{=_{\mathsf{Vc}}, =_{\wedge\mathsf{c}}, =_{\mathsf{a}}\downarrow, =_{\mathsf{a}}\uparrow, =_{\mathsf{f}}\downarrow, =_{\mathsf{t}}\downarrow, =_{\mathsf{f}}\uparrow, =_{\mathsf{t}}\uparrow, =_{\mathsf{f}\wedge}\downarrow, =_{\mathsf{t}\vee}\downarrow, =_{\mathsf{f}\wedge}\uparrow, =_{\mathsf{t}\vee}\uparrow\}$. Furthermore, if $\rho \in \mathsf{SKS}$, and there is a derivation

$$\dfrac{\dfrac{\alpha}{=\dfrac{\alpha'}{\rho\dfrac{\beta'}{=\dfrac{}{\beta}}}}}{}$$

we sometimes write

$$\dfrac{\alpha}{\rho\dfrac{}{\beta}} \quad .$$

E.g., instead of the derivation

$$\dfrac{\dfrac{\dfrac{\dfrac{\alpha \vee \beta}{=_{\mathsf{Vc}}\dfrac{}{\beta \vee \alpha}} \wedge \gamma}{=_{\wedge\mathsf{c}}\dfrac{}{\gamma \wedge [\beta \vee \alpha]}}}{\mathsf{s}\dfrac{}{(\gamma \wedge \beta) \vee \alpha}}}{=_{\mathsf{Vc}}\dfrac{}{\alpha \vee =_{\wedge\mathsf{c}}\dfrac{\gamma \wedge \beta}{\beta \wedge \gamma}}} \quad ,$$

we write

$$\mathsf{s}\dfrac{[\alpha \vee \beta] \wedge \gamma}{\alpha \vee (\beta \wedge \gamma)} \quad .$$

See the proofs of Theorems 6.3.2 to 6.4.4 on pages 53–61 for more examples of implicit equations.

We now give some standard results which will also serve as examples of system SKS and the functorial calculus.

Lemma 2.3.8. *Given a context $\xi\{\ \}$ and a formula α there exist derivations* $\begin{array}{c}\alpha\wedge\xi\{\mathsf{t}\}\\ \|\{\mathsf{s}\}\\ \xi\{\alpha\}\end{array}$ *and* $\begin{array}{c}\xi\{\alpha\}\\ \|\{\mathsf{s}\}\\ \xi\{\mathsf{f}\}\vee\alpha\end{array}$ *; both of whose size depend at most quadratically on the size of $\xi\{\alpha\}$.*

Proof. We show how to construct the first derivation, the second one can be done symmetrically. We argue by induction on the number of atom occurrences in $\xi\{\ \}$. The base case, $\xi\{\ \}=\{\ \}$, is trivial and the inductive cases are:

$$\mathsf{s}\dfrac{\alpha\ \wedge=\dfrac{\xi\{\mathsf{t}\}}{\xi'\{\mathsf{t}\}\vee\beta}}{=\dfrac{\left[\begin{array}{lll}\alpha\wedge\xi'\{\mathsf{t}\}\\ \quad\Psi\|\{\mathsf{s}\}\quad\vee\ \beta\\ \xi'\{\alpha\}\end{array}\right]}{\xi\{\alpha\}}}\qquad\text{and}\qquad\dfrac{=\dfrac{\alpha\wedge\xi\{\mathsf{t}\}}{\left(\begin{array}{lll}\alpha\wedge\xi'\{\mathsf{t}\}\\ \quad\Psi'\|\{\mathsf{s}\}\quad\wedge\ \beta\\ \xi'\{\alpha\}\end{array}\right)}}{=\dfrac{}{\xi\{\alpha\}}}\ ,$$

for some $\xi'\{\ \}$ and β where β is not a unit and Ψ and Ψ' exist by the inductive hypothesis. □

Notation 2.3.9. We often write $\mathsf{ss}\uparrow\dfrac{\alpha\wedge\xi\{\mathsf{t}\}}{\xi\{\alpha\}}$ and $\mathsf{ss}\downarrow\dfrac{\xi\{\alpha\}}{\xi\{\mathsf{f}\}\vee\alpha}$, instead of, respectively, the derivations $\begin{array}{c}\alpha\wedge\xi\{\mathsf{t}\}\\ \|\\ \xi\{\alpha\}\end{array}$ and $\begin{array}{c}\xi\{\alpha\}\\ \|\\ \xi\{\mathsf{f}\}\vee\alpha\end{array}$, as defined in the proof of Lemma 2.3.8. Instead of the derivation

$$\mathsf{s}\dfrac{\mathsf{ss}\downarrow\dfrac{\zeta\{\alpha\}}{\zeta\{\mathsf{f}\}\vee\alpha}\ \wedge\ \xi\{\mathsf{t}\}}{\zeta\{\mathsf{f}\}\ \vee\ \mathsf{ss}\uparrow\dfrac{\alpha\wedge\xi\{\mathsf{t}\}}{\xi\{\alpha\}}}$$

we write $\mathsf{ss}\dfrac{\zeta\{\alpha\}\wedge\xi\{\mathsf{t}\}}{\zeta\{\mathsf{f}\}\vee\xi\{\alpha\}}$.

We now show a consequenc of the previous Lemma, which will be very useful in Subsection 6.4.1 on page 65.

Lemma 2.3.10. *Given a formula α and an atom a, there exist derivations* $\begin{array}{c}a\wedge\alpha\{a/\mathsf{t}\}\\ \|\{\mathsf{ac}\uparrow,\mathsf{s}\}\\ \alpha\end{array}$ *and* $\begin{array}{c}\alpha\\ \|\{\mathsf{ac}\downarrow,\mathsf{s}\}\\ \alpha\{a/\mathsf{f}\}\vee a\end{array}$*; both of whose size depend at most quadratically on the size of α.*

Proof. We show how to construct the first derivation, the second one can be done symmetrically. The result follows by induction on the number of occurrences of a in α, and Lemma 2.3.8 on the preceding page. The base case is trivial. Let $\xi\{\ \}$ be some context such that, $\alpha = \xi\{a\}$, then the inductive case is:

$$\dfrac{\dfrac{a}{a \wedge a} \wedge (\xi\{a/\mathsf{t}\})\{\mathsf{t}\}}{a \wedge (\xi\{a/\mathsf{t}\})\{a\}}\,\Vert\{\mathsf{s}\} \quad .$$

□

For an example of the use of Lemma 2.3.10 on the facing page see Remark 2.3.16 on page 19.

Lemma 2.3.11. *Given a formula α, there exist derivations* $\begin{matrix}\mathsf{f}\\ \Vert\{\mathsf{aw}\downarrow,\mathsf{s}\}\\ \alpha\end{matrix}$ *and* $\begin{matrix}\alpha\\ \Vert\{\mathsf{aw}\uparrow,\mathsf{s}\}\\ \mathsf{t}\end{matrix}$*; both of whose size depend at most quadratically on the size of α.*

Proof. We show how to construct the first derivation, the second one can be done symmetrically. Let $a_1, \ldots, a_n$ be the atoms appearing in α, then there exists a derivation

$$\begin{matrix}\alpha\{a_1/\mathsf{f},\ldots,a_n/\mathsf{f}\}\\ \Vert\{\mathsf{aw}\downarrow\}\\ \alpha\end{matrix} \quad .$$

Since $\alpha\{a_1/\mathsf{f},\ldots,a_n/\mathsf{f}\}$ contains no atom occurrences, there exists a derivation

$$\begin{matrix}\mathsf{f}\\ \Vert\{=_{\mathsf{f}}\downarrow,=_{\mathsf{t}}\downarrow,=_{\mathsf{f}\wedge}\downarrow,=_{\mathsf{t}\vee}\downarrow\}\\ \alpha\{a_1/\mathsf{f},\ldots,a_n/\mathsf{f}\}\end{matrix} \quad \text{or} \quad \begin{matrix}\dfrac{\dfrac{\dfrac{\mathsf{f}}{\mathsf{f}\wedge[\mathsf{t}\vee\mathsf{f}]}\,=}{(\mathsf{f}\wedge\mathsf{t})\vee\mathsf{t}}\,\mathsf{s}}{\mathsf{t}}\,=\\ \Vert\{=_{\mathsf{f}}\downarrow,=_{\mathsf{t}}\downarrow,=_{\mathsf{f}\wedge}\downarrow,=_{\mathsf{t}\vee}\downarrow\}\\ \alpha\{a_1/\mathsf{f},\ldots,a_n/\mathsf{f}\}\end{matrix} \quad .$$

□

Lemma 2.3.12. *Given a formula α, there exist derivations* $\begin{matrix}\alpha\vee\alpha\\ \Vert\{\mathsf{ac}\downarrow,\mathsf{m}\}\\ \alpha\end{matrix}$ *and* $\begin{matrix}\alpha\\ \Vert\{\mathsf{ac}\uparrow,\mathsf{m}\}\\ \alpha\wedge\alpha\end{matrix}$*; both of whose size depend at most quadratically on the size of α.*

Proof. We show how to construct the first derivation, the second one can be done symmetrically. We argue by induction on the size of α. We have to consider the following three base cases

$$=_{\mathsf{f}\wedge}\uparrow\dfrac{\mathsf{t}\vee\mathsf{t}}{\mathsf{t}} \quad , \quad =_{\mathsf{t}}\uparrow\dfrac{\mathsf{f}\vee\mathsf{f}}{\mathsf{f}} \quad \text{and} \quad \dfrac{a\vee a}{a} \quad ,$$

and two inductive cases:

$$\mathsf{m}\dfrac{(\alpha\wedge\beta)\vee(\alpha\wedge\beta)}{\begin{array}{c}\alpha\vee\alpha\\ \|\{\mathsf{ac}\downarrow,\mathsf{m}\}\\ \alpha\end{array}\ \wedge\ \begin{array}{c}\beta\vee\beta\\ \|\{\mathsf{ac}\downarrow,\mathsf{m}\}\\ \beta\end{array}} \quad \text{and} \quad =\dfrac{[\alpha\vee\beta]\vee[\alpha\vee\beta]}{\begin{array}{c}\alpha\vee\alpha\\ \|\{\mathsf{ac}\downarrow,\mathsf{m}\}\\ \alpha\end{array}\ \vee\ \begin{array}{c}\beta\vee\beta\\ \|\{\mathsf{ac}\downarrow,\mathsf{m}\}\\ \beta\end{array}} \ .$$

□

Notation 2.3.13. In the non-atomic version of system SKS the derivations shown in the proofs of Lemma 2.3.11 on the preceding page and Lemma 2.3.12 on the previous page correspond to (co)weakening and (co)contractions, respectively. For this reason we sometimes write the inference rules $\mathsf{w}\downarrow\dfrac{\mathsf{f}}{\alpha}$, $\mathsf{w}\uparrow\dfrac{\alpha}{\mathsf{t}}$, $\mathsf{c}\downarrow\dfrac{\alpha\vee\alpha}{\alpha}$ and $\mathsf{c}\uparrow\dfrac{\alpha}{\alpha\wedge\alpha}$ instead of the derivations $\begin{array}{c}\mathsf{f}\\ \|\{\mathsf{aw}\downarrow,\mathsf{s}\}\\ \alpha\end{array}$, $\begin{array}{c}\alpha\\ \|\{\mathsf{aw}\uparrow,\mathsf{s}\}\\ \mathsf{t}\end{array}$, $\begin{array}{c}\alpha\vee\alpha\\ \|\{\mathsf{ac}\downarrow,\mathsf{m}\}\\ \alpha\end{array}$ and $\begin{array}{c}\alpha\\ \|\{\mathsf{ac}\uparrow,\mathsf{m}\}\\ \alpha\wedge\alpha\end{array}$, respectively.

To give an example of the notions defined so far, we now show a completeness proof of system SKS.

Theorem 2.3.14. *System* SKS *is complete for propositional classical logic.*

Proof. Consider a tautology α. We show by induction on the number of atoms appearing in α that there exists a proof of α in SKS. For the base case, let α consist only of units. Then, since α is a tautology, we can build

$$\begin{array}{c}\mathsf{t}\\ \|\{=_{\mathsf{f}}\downarrow,=_{\mathsf{t}}\downarrow,=_{\mathsf{f}\wedge}\downarrow,=_{\mathsf{t}\vee}\downarrow\}\\ \alpha\end{array} \ .$$

For the inductive case, let α be a tautology containing instances of the atom a. We consider two cases:

- if α does not contain an instance of $\bar{a}$, then $\alpha\{a/\mathsf{f}\}$ is a tautology, so by the inductive hypothesis we can build

$$\begin{array}{c}\mathsf{t}\\ \|\\ \alpha\{a/\mathsf{f}\}\\ \|\{\mathsf{aw}\downarrow\}\\ \alpha\end{array} \ ;$$

- otherwise, both $\alpha\{a/\mathsf{t},\bar{a}/\mathsf{f}\}$ and $\alpha\{a/\mathsf{f},\bar{a}/\mathsf{t}\}$ are tautologies, so by the inductive hypothesis we can build

$$\Phi = \begin{array}{c} \mathsf{t} \\ \| \\ \alpha\{a/\mathsf{t},\bar{a}/\mathsf{f}\} \\ \|\{\mathsf{ai}{\downarrow},\mathsf{aw}{\downarrow}\} \\ \alpha\{a/[a \vee \bar{a}]\} \\ \|\{\mathsf{ss}{\downarrow}\} \\ \left[\begin{array}{cc} & \bar{a} \vee \cdots \vee \bar{a} \\ \alpha \ \vee & \|\{\mathsf{ac}{\downarrow}\} \\ & \bar{a} \end{array}\right] \end{array} .$$

Using Φ and the inductive hypothesis we can build the desired derivation:

$$\begin{array}{c} \mathsf{t} \\ \| \\ \alpha\{a/\mathsf{f},\bar{a}/\mathsf{t}\} \\ \|\{\mathsf{aw}{\downarrow}\} \\ \alpha\{\bar{a}/\mathsf{t}\} \\ \| \\ \alpha\{\bar{a}/[\alpha \vee \bar{a}]\} \\ \|\{\mathsf{ss}{\downarrow}\} \\ \alpha \vee \cdots \vee \alpha \\ \|\mathsf{c}{\downarrow} \\ \alpha \end{array} .$$

□

Remark 2.3.15. Given any formulae α and β and any context $\xi\{\ \}$, then, by a construction similar to the one in the proof of Lemma 2.3.8 on page 16, we can build a derivation $\begin{array}{c}\xi\{\alpha \vee \beta\} \\ \|\{\mathsf{s},=_{\vee\mathsf{c}},=_{\wedge\mathsf{c}},=_{\mathsf{a}}{\downarrow}\} \\ \xi\{\alpha\} \vee \beta\end{array}$. If we use this derivation instead of the rule $\mathsf{ss}\downarrow$ in the proof of Theorem 2.3.14 on the preceding page, it follows that the system

$$\{\mathsf{ai}{\downarrow},\mathsf{ac}{\downarrow},\mathsf{aw}{\downarrow},\mathsf{s},\mathsf{m},=_{\vee\mathsf{c}},=_{\wedge\mathsf{c}},=_{\mathsf{a}}{\downarrow},=_{\mathsf{f}}{\downarrow},=_{\mathsf{t}}{\downarrow},=_{\mathsf{f}\wedge}{\downarrow},=_{\mathsf{t}\vee}{\downarrow}\}$$

is complete for classical logic. This justifies the naming of the invertible rules, as the tradition is in deep inference to label admissible rules with an $\uparrow$.

Remark 2.3.16. If we do not restrict ourselves to the downfragment of SKS, we can build a more compact proof than what we do in Theorem 2.3.14 on the facing page, by using the

following as the inductive case:

$$\dfrac{\dfrac{\dfrac{\mathsf{t}}{\left(\dfrac{\dfrac{\mathsf{t}}{a \vee \bar{a}} \wedge \alpha\{a/\mathsf{t}, \bar{a}/\mathsf{f}\}}{(a \wedge \alpha\{a/\mathsf{t}, \bar{a}/\mathsf{f}\}) \vee \bar{a}}\,\mathsf{s} \wedge \alpha\{a/\mathsf{f}, \bar{a}/\mathsf{t}\}\right)}\,\mathsf{s}}{\left(a \wedge \begin{array}{c}\alpha\{a/\mathsf{t}, \bar{a}/\mathsf{f}\} \\ \|\{\mathsf{aw}\downarrow\} \\ \alpha\{a/\mathsf{t}\} \\ \|\{\mathsf{ac}\uparrow, \mathsf{s}\} \\ \alpha\end{array}\right) \vee \left(\bar{a} \wedge \begin{array}{c}\alpha\{a/\mathsf{f}, \bar{a}/\mathsf{t}\} \\ \|\{\mathsf{aw}\downarrow\} \\ \alpha\{\bar{a}/\mathsf{t}\} \\ \|\{\mathsf{ac}\uparrow, \mathsf{s}\} \\ \alpha\end{array}\right)}\,\mathsf{c}\downarrow}{\alpha}\,,$$

where we have used the derivations constructed in the proof of Lemma 2.3.10 on page 16.

Part II

Atomic Flows

Chapter 3

Atomic Flows

In this chapter we introduce the main tool used in this thesis, a geometric proof invariant called 'atomic flows'. An atomic flow is a directed graph obtained from a derivation by only retaining information about the creation and destruction of atom occurrences. Notably, the atomic flow of a derivation completely disregards all the logical relations and linear inference rule instances; so, an atomic flow is not a derivation.

Atomic flows can be seen as either specialised Buss flow graphs [Bus91, Car97], or a variation of the kind of proof nets developed in [Str05, Str09], based on work done in [LS05b]. The only difference between atomic flows and these proof nets is that the proof nets implement (co)associativity of (co)contraction and dinaturality of interaction and cut, while atomic flows do not. For a more detailed comparison see [Str09]. Despite their similarities, the motivation and use of atomic flows differ from that of proof nets.

We can think of atomic flows as composite diagrams that are freely generated from a set of six elementary diagrams. Technically, atomic flows are special kinds of labelled directed acyclic graphs, and the properties of their vertices are dictated by their labels, which we define as follows.

Definition 3.0.1. We call the following six diagrams (*atomic-flow*) *labels*:

ai↓ or *interaction* aw↓ or *weakening* ac↓ or *contraction*

ai↑ or *cut* aw↑ or *coweakening* ac↑ or *cocontraction*

Definition 3.0.2. An (*atomic*) *flow* is a tuple $(V, E, \eta, \mathrm{up}, \mathrm{lo})$, such that:

1. V is a finite set of *vertices*, denoted by v;
2. E is a finite set of *edges*, denoted by ϵ, ι or small numerals 1, 2, ...;

3. $\eta\colon V \to \{\mathsf{ai}{\downarrow}, \mathsf{ai}{\uparrow}, \mathsf{aw}{\downarrow}, \mathsf{aw}{\uparrow}, \mathsf{ac}{\downarrow}, \mathsf{ac}{\uparrow}\}$ maps vertices to their *labels*;

4. $\mathrm{up}\colon E \to V \cup \{\top\}$ and $\mathrm{lo}\colon E \to V \cup \{\bot\}$ are, respectively, the *upper* and *lower* maps, and $\top$ and $\bot$ are special vertices not belonging to V; we define, for every $v \in V \cup \{\top, \bot\}$, the set $L_v = \{\epsilon \mid \mathrm{up}(\epsilon) = v\}$ of *lower edges of* v, the set $U_v = \{\epsilon \mid \mathrm{lo}(\epsilon) = v\}$ of *upper edges of* v, and the set $E_v = L_v \cup U_v$ of *edges of* v;

5. if $|S|$ denotes the cardinality of set S, we have that

$$\begin{aligned}
\text{if } \eta(v) = \mathsf{ai}{\downarrow} \text{ then } |L_v| = 2 \text{ and } |U_v| = 0,\\
\text{if } \eta(v) = \mathsf{ai}{\uparrow} \text{ then } |L_v| = 0 \text{ and } |U_v| = 2,\\
\text{if } \eta(v) = \mathsf{aw}{\downarrow} \text{ then } |L_v| = 1 \text{ and } |U_v| = 0,\\
\text{if } \eta(v) = \mathsf{aw}{\uparrow} \text{ then } |L_v| = 0 \text{ and } |U_v| = 1,\\
\text{if } \eta(v) = \mathsf{ac}{\downarrow} \text{ then } |L_v| = 1 \text{ and } |U_v| = 2,\\
\text{if } \eta(v) = \mathsf{ac}{\uparrow} \text{ then } |L_v| = 2 \text{ and } |U_v| = 1;
\end{aligned}$$

6. there is no sequence $\epsilon_1, \ldots, \epsilon_h$ of edges of V such that $\mathrm{up}(\epsilon_i) = \mathrm{lo}(\epsilon_{i+1 \pmod h})$, for $1 \leqslant i \leqslant h$;

7. there is a *polarity assignment* $\pi\colon E \to \{-, +\}$ such that, for every $v \in V$,

   (a) if $\eta(v) \in \{\mathsf{ac}{\downarrow}, \mathsf{ac}{\uparrow}\}$ then $\pi(E_v) = \{-\}$ or $\pi(E_v) = \{+\}$;

   (b) if $\eta(v) \in \{\mathsf{ai}{\downarrow}, \mathsf{ai}{\uparrow}\}$ then $\pi(E_v) = \{-, +\}$.

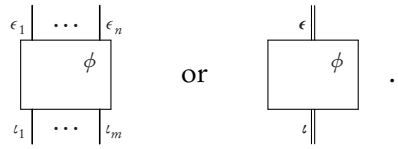
Given an atomic flow ϕ, we say that the sets $L_\top = \{\epsilon_1, \ldots, \epsilon_h\}$ and $U_\bot = \{\iota_1, \ldots, \iota_k\}$ contain, respectively, the *upper* and *lower edges of* ϕ.

Notation 3.0.3. We will use the letters ϕ and ψ, sometimes with standard additional decorations, to denote atomic flows.

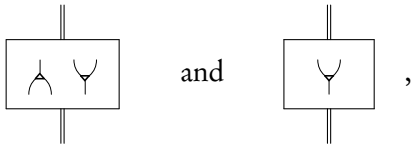
An atomic flow is a directed graph, whose edges are associated to atom occurrences in derivations, and the direction of the edges corresponds to the up-down direction in a derivation. Vertices are associated to points in the derivation where atom occurrences are created or destroyed, and the nature of each vertex is described by its label. Naturally, these graphs are acyclic (condition 6). The two special vertices $\top$ and $\bot$ represent the top and bottom of a derivation: we can consider $\top$ the vertex that creates all the atom occurrences in the premiss and $\bot$ the vertex that destroys all atom occurrences in the conclusion.

The polarity assignment condition (7) ensures that atoms in (co)contractions have the same polarity, and those in interactions and cuts have dual polarities (as happens in derivations). Every atomic flow has 2^n polarity assignments, where n is the number of connected components in the graph. We should not be worried about the apparent complexity of the polarity assignment condition: in fact, we could equivalently consider two sorts of (co)contraction and (co)weakening labels, the negative and the positive ones, and ask for vertices to be joined by respecting their polarities. This is clearly a locally checkable property, much simpler than, for example, some global correctness criterion for proof nets.

Notation 3.0.4. Let ϕ be a flow with upper edges $\epsilon = \epsilon_1, \ldots, \epsilon_n$ and lower edges $\iota = \iota_1, \ldots, \iota_m$, we then represent it as



We sometimes use flow labels to indicate what kind of vertices a flow might contain. *E.g.*, the following flows



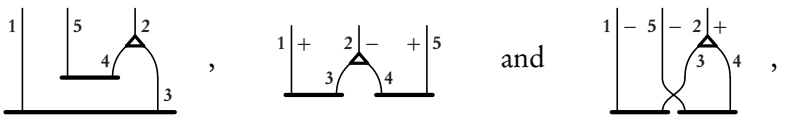
do not contain $\mathsf{ai}{\downarrow}$, $\mathsf{ai}{\uparrow}$, $\mathsf{aw}{\downarrow}$, $\mathsf{aw}{\uparrow}$ vertices, and in addition the flow to the right does not contain $\mathsf{ac}{\uparrow}$ vertices.

In general, we represent atomic flows as directed-graph diagrams, except that the special vertices $\top$ and $\bot$ are not shown, and the labels of the vertices are explicitly shown as graphical elements. When we refer to the vertices of an atomic flow, we do not include $\top$ and $\bot$. Sometimes we identify vertices with their labels.

Example 3.0.5. Consider the flow

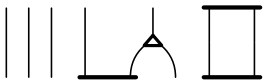
$$\begin{aligned} A = (&\{ \nu_1 , \nu_2 , \nu_3 \}, \\ &\{ 1 , 2 , 3 , 4 , 5 \}, \\ &\{ \nu_1 \mapsto \mathsf{ai}{\uparrow} , \nu_2 \mapsto \mathsf{ac}{\uparrow} , \nu_3 \mapsto \mathsf{ai}{\uparrow} \}, \\ &\{ 1 \mapsto \top , 2 \mapsto \top , 3 \mapsto \nu_2 , 4 \mapsto \nu_2 , 5 \mapsto \top \}, \\ &\{ 1 \mapsto \nu_1 , 2 \mapsto \nu_2 , 3 \mapsto \nu_1 , 4 \mapsto \nu_3 , 5 \mapsto \nu_3 \}) \quad ; \end{aligned}$$

the following are three of its possible representations:



in the last two diagrams, we also indicated each of the two possible polarity assignments. This flow has one cocontraction and two cointeraction vertices; it has three upper edges, 1, 2 and 5, and no lower edges.

Example 3.0.6. The flow



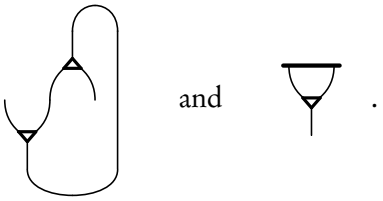
is obtained by juxtaposing (*i.e.*, taking the disjoint union of):

- three edges,

- a flow obtained by composing a cut vertex with a cocontraction vertex, and
- a flow obtained by composing an interaction vertex with a cut vertex.

Note that there are no cycles in the flow, and that we can find 32 different polarity assignments, *i.e.*, two for each of the five connected components of the flow.

Example 3.0.7. The following two diagrams are not atomic flows:



The left one is not a flow because it contains a cycle, and the right one because there is no possible polarity assignment.

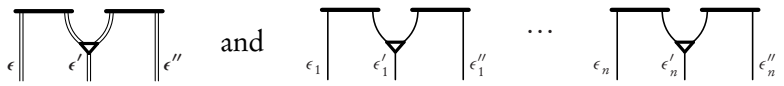
Definition 3.0.8. Given two flows $\phi_1 = (V_1, E_1, \eta_1, \mathrm{up}_1, \mathrm{lo}_1)$ and $\phi_2 = (V_2, E_2, \eta_2, \mathrm{up}_2, \mathrm{lo}_2)$, an *isomorphism* between ϕ_1 and ϕ_2 is a pair of functions (f_V, f_E), such that

- f_V is a bijection from V_1 to V_2; and
- f_E is a bijection from E_1 to E_2,

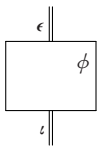
such that, for every ϵ in E_1,

- for every v in V_1, $\mathrm{up}_1(\epsilon) = v$ (resp., $\mathrm{lo}_1(\epsilon) = v$) if and only if $\mathrm{up}_2(f_E(\epsilon)) = f_V(v)$ (resp., $\mathrm{lo}_2(f_E(\epsilon)) = f_V(v)$); and
- $\mathrm{up}_1(\epsilon) = \top$ (resp., $\mathrm{lo}_1(\epsilon) = \bot$) if and only if $\mathrm{up}_2(f_E(\epsilon)) = \top$ (resp., $\mathrm{lo}_2(f_E(\epsilon)) = \bot$).

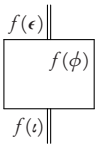
Notation 3.0.9. We extend the double-line notation to collections of isomorphic flows. For example, for $n \geqslant 0$; $\boldsymbol{\epsilon} = \epsilon_1, \ldots, \epsilon_n$; $\boldsymbol{\epsilon'} = \epsilon'_1, \ldots, \epsilon'_n$; and $\boldsymbol{\epsilon''} = \epsilon''_1, \ldots, \epsilon''_n$, the following diagrams represent the same flow:



Notation 3.0.10. Given a flow

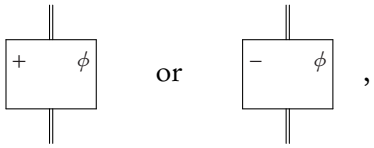


and a flow ψ which is isomorphic to ϕ, whenever we write



we mean that f is a given isomorphism between ϕ and ψ.

Notation 3.0.11. Given a flow ϕ and a polarity assignment π for ϕ, whenever we write



respectively, we mean that all the edges in ϕ have polarity assignment + or –, respectively. If we label a flow with a polarity assignment it can not contain any interaction or cut vertices duo to property 7 of Definition 3.0.2 on page 22.

Definition 3.0.12. Given a flow ϕ and a polarity assignment π for ϕ, the polarity assignment $\bar{\pi}$ for ϕ is defined to be, for every ϵ in ϕ:

$$\bar{\pi}(\epsilon) = \begin{cases} - & \text{if } \pi(\epsilon) = +, \\ + & \text{otherwise.} \end{cases}$$

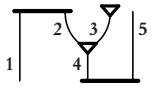
3.1 Paths and Cycles

We now define the notions of 'path', 'ai-path' and 'ai-cycle' in atomic flows. Paths are sequences of adjacent edges that only 'go down' or only 'go up'; ai-paths are formed by joining paths at interaction or cointeraction vertices; ai-cycles are circular ai-paths.

Definition 3.1.1. Given an atomic flow $(V, E, \eta, \mathrm{up}, \mathrm{lo})$ and $\epsilon_1, \ldots, \epsilon_h \in E$ such that, for $1 \leqslant i < h$, we have $\mathrm{lo}(\epsilon_i) = \mathrm{up}(\epsilon_{i+1})$, $\mathrm{up}(\epsilon_1) = v$ and $\mathrm{lo}(\epsilon_h) = v'$, we say that $\epsilon_1, \ldots, \epsilon_h$ is a *path from v to v'* and that $\epsilon_h, \ldots, \epsilon_1$ is a *path from v' to v*; both paths have *length h*.

An ai-*path from v to v'* of *length h* is either a path from v to v' of length h or a sequence of edges $\epsilon_1, \ldots, \epsilon_k, \epsilon_{k+1}, \ldots, \epsilon_h$ such that $\epsilon_k \neq \epsilon_{k+1}$ and, for some $v'' \in V$ with $\eta(v'') \in \{\mathsf{ai}{\downarrow}, \mathsf{ai}{\uparrow}\}$, we have that $\epsilon_1, \ldots, \epsilon_k$ is an ai-path from v to v'' and $\epsilon_{k+1}, \ldots, \epsilon_h$ is an ai-path from v'' to v'. An ai-path of length h is *maximal* if no ai-path containing its edges has length greater than h. An ai-path from (resp., to) v of length h is a *maximal* ai-*path from* (resp., *to*) v if no ai-path from (resp., to) v containing its edges has length greater than h.

Example 3.1.2. The flow on the left has the ai-paths on the right, and the paths are marked with an asterisk:



$$\begin{array}{llllll} \mathbf{1}^* & & & & & \\ \mathbf{1,2} & \mathbf{2}^* & \mathbf{3}^* & & & \\ \mathbf{1,2,4} & \mathbf{2,4}^* & \mathbf{3,4}^* & \mathbf{4}^* & & \\ \mathbf{1,2,4,5} & \mathbf{2,4,5} & \mathbf{3,4,5} & \mathbf{4,5} & \mathbf{5}^* & \end{array} .$$

In addition, the flow has the paths and ai-paths obtained from the shown ones by inverting the order of edges, for example 5, 4, 2, 1 is an ai-path. The ai-paths from the interaction vertex are 1 and 2 and 2, 4 and 2, 4, 5; the ai-paths to the contraction vertex are 1, 2 and 2 and 3 and 4 and 5, 4. The maximal ai-paths are 1, 2, 4, 5 and 3, 4, 5 and their inverses. The maximal ai-paths from the cointeraction vertex are 4, 2, 1 and 4, 3 and 5; the maximal ai-paths to the contraction vertex are 1, 2 and 3 and 5, 4.

3.2 Subflows

Definition 3.2.1. Given two flows $\phi_1 = (V_1, E_1, \eta_1, \mathrm{up}_1, \mathrm{lo}_1)$ and $\phi_2 = (V_2, E_2, \eta_2, \mathrm{up}_2, \mathrm{lo}_2)$, we say that ϕ_1 is a *subflow* of ϕ_2, if

- $V_1 \subset V_2$;
- $E_1 \subset E_2$;
- $\eta_1 = \eta_2|_{V_1}$;
- for every ϵ in E_1

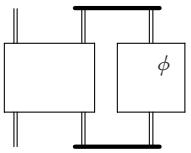
$$\mathrm{up}_1(\epsilon) = \begin{cases} \mathrm{up}_2(\epsilon) & \text{if } \mathrm{up}_2(\epsilon) \in V_1, \\ \top & \text{otherwise.} \end{cases} \quad \text{and}$$

$$\mathrm{lo}_1(\epsilon) = \begin{cases} \mathrm{lo}_2(\epsilon) & \text{if } \mathrm{lo}_2(\epsilon) \in V_1, \\ \bot & \text{otherwise.} \end{cases} \quad ; \text{and}$$

- if v_1 and v_2 are vertices in ϕ_1, and there is a vertex v' in ϕ_2, such that there are paths from v_1 to v' and from v' to v_2 in ϕ_2, then v' is a vertex in ϕ_1.

Definition 3.2.2. Given two flows ϕ and ψ, such that ϕ is a subflow of ψ, we say that ϕ *is an isolated subflow of* ψ if there is no path in ψ from a vertex in ϕ to $\top$ or $\bot$.

Example 3.2.3. In the following flow, ϕ is an isolated subflow of ψ:

$\psi =$ 

.

For other examples of isolated subflows see Definition 6.2.1 on page 47 and Definition 6.4.1 on page 59.

Definition 3.2.4. Given two flows ϕ and ψ, such that ϕ is a subflow of ψ, we say that ϕ *is a connected component of* ψ if, for any two polarity assignments π and π' for ψ and for any two edges ϵ and ϵ' in ϕ, $\pi(\epsilon) = \pi(\epsilon')$ if and only if $\pi'(\epsilon) = \pi'(\epsilon')$.

Chapter 4

Atomic Flows and Derivations

4.1 Extracting Flows from Derivations

We now define the mapping from derivations to flows. As we said, the idea is that structural rule instances map to the respective atomic-flow vertices, and the edges trace the atom occurrences between rule instances.

Definition 4.1.1. Given a derivation Φ, we define the flow ϕ *associated with* Φ:

- if Φ is a unit, then ϕ is the empty flow;
- if Φ is an atom, then ϕ is a flow containing only the edge ϵ and no vertices; we say that Φ *is mapped to* ϵ;
- if $\Phi = \Psi_1 \vee \Psi_2$ or $\Phi = \Psi_1 \wedge \Psi_2$, and ψ_1 and ψ_2 are the flow associated with Ψ_1 and Ψ_2, respectively, then ϕ is the disjoint union of ψ_1 and ψ_2; and
- if

$$\Phi = \rho \dfrac{\begin{array}{c} A \\ \Phi_1 \| \\ A' \end{array}}{\begin{array}{c} B' \\ \Phi_2 \| \\ B \end{array}} \quad ,$$

  where ψ_1 (resp., ψ_2) is the flow associated with Ψ_1 (resp., Ψ_2), then ϕ is obtained by modifying the disjoint union of ϕ_1 and ϕ_2 in the following way:

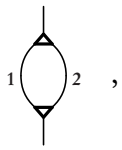
  - if ρ is a structural inference rule, ϕ also contains a new vertex v that is labelled with the name of ρ. Furthermore, the lower (resp., upper) map of ϕ maps each of the lower (resp., upper) edges of ϕ_1 (resp., ϕ_2) to v; we say that ρ *is mapped to* v, and

– if ρ is a linear inference rule, then the lower edges of ϕ_1 are pairwise identified with the upper edges of ϕ_2 in such a way that an atom occurrence in the premiss of ρ is mapped to the same edge as the corresponding atom occurrence in the conclusion of ρ.

Remark 4.1.2. Given a derivation Φ, one can associate several atomic flows with it, because we have to choose names for the vertices and edges. However, this is a rather trivial form of non-determinism, since the position of atom occurrences and inference rule instances can be located in a derivation without any ambiguity. Thus, given two atomic flows ϕ and ϕ' associated with the same derivation Φ, there is a unique flow isomorphism between them that makes the vertices correspond to their position in Φ. Furthermore, if ϕ is associated with Φ and if $\alpha : \phi' \to \phi$ is an atomic flow isomorphism, then one can immediately turn ϕ' into an associated flow for Φ in the following way: for every atom occurrence a (resp., structural inference rule instance ρ) in Φ and edge ϵ (resp., vertex v) in ϕ', we let a (resp., ρ) map to ϵ (resp., v) if and only if a (resp., ρ) maps to $\alpha(\epsilon)$ (resp., $\alpha(\rho)$).

Remark 4.1.3. It should be noted that the mapping from atom occurrences (resp., rule instances) in Φ to edges (resp., vertices) in ϕ is not uniquely defined. In other words, ϕ might have non-trivial automorphisms. However, this will not cause us any problems in this thesis, as in the cases where the mapping is ambiguous (Section 7.1), we only rely on its existence.

Example 4.1.4. The following flow has an automorphism that maps 1 to 2 and 2 to 1



,

it can therefore be associated with the following derivation in two different ways

$$\cfrac{\cfrac{\cfrac{\cfrac{a}{a \wedge a}\,\mathsf{ac}\uparrow}{(a \wedge a) \vee (\mathsf{t} \wedge \mathsf{f})}\,=}{[a \vee \mathsf{t}] \wedge [a \vee \mathsf{f}]}\,\mathsf{m}}{a \vee (\mathsf{t} \wedge [a \vee \mathsf{f}])}\,\mathsf{s}}{\cfrac{a \vee a}{a}\,\mathsf{ac}\downarrow}\,= \quad .$$

Example 4.1.5. Figure 4-1 on the next page has three examples of derivations and their associated flows, where colours are used to indicate the mapping from atom occurrences to edges.

Definition 4.1.6. Given a derivation Φ with flow ϕ, and an atom a, the *restriction of ϕ to a* is the largest subflow ψ of ϕ, such that every edge of ψ is mapped to from occurrences of a or $\bar{a}$.

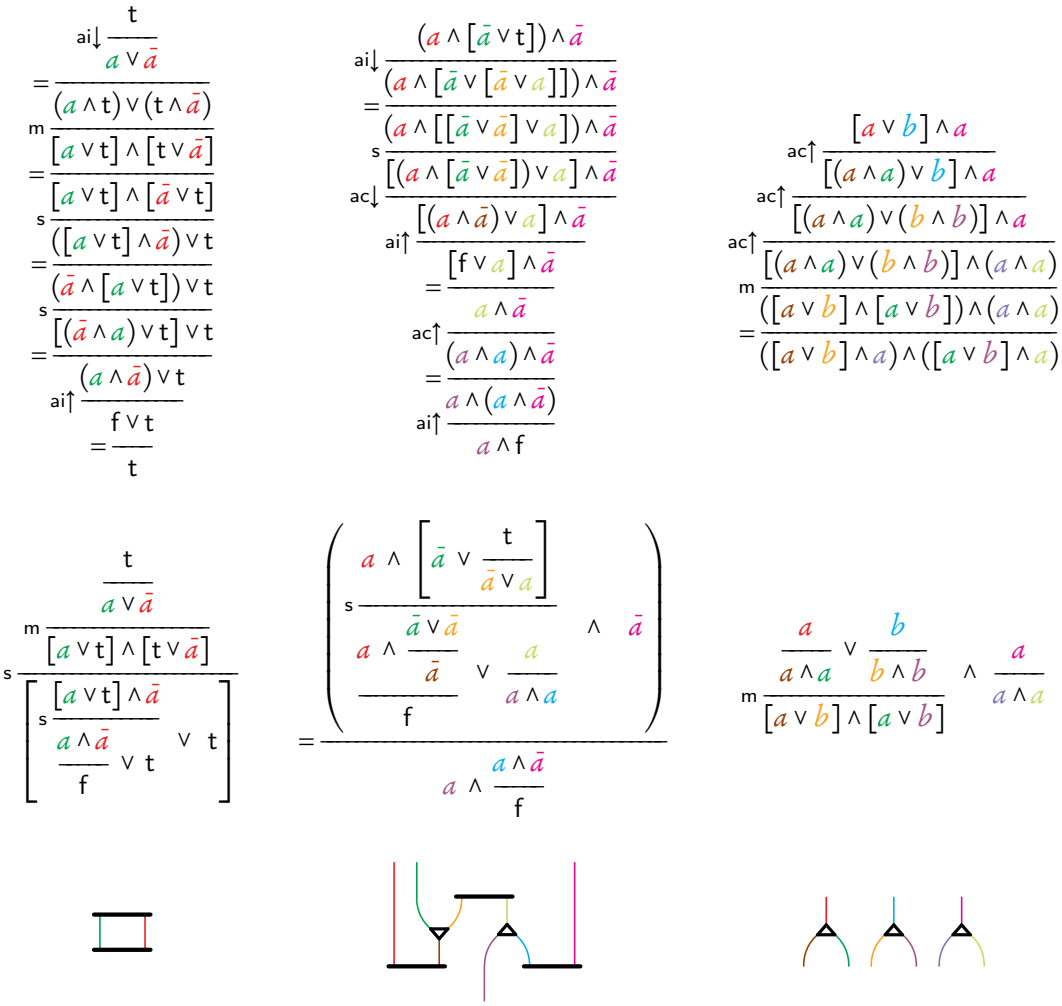
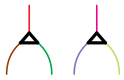


Figure 4-1: Examples of derivations in the calculus of structures (top row), their translation into the functorial calculus (middle row), and the flows associated with the latter (bottom row).

Example 4.1.7. Consider the rightmost derivation and its associated flow in Figure 4-1 on the facing page. The restriction of this flow to a is:



We now show that the process of association of a flow to a derivation is 'surjective modulo renaming', in the sense that every flow is associated with some derivation.

It should be noted that the following result relies on the fact that both the formula structures of the premiss and conclusion, as well as all units occuring in the derivation, are ignored when extracting a flow. In particular, the derivation we construct in the following proof is 'trivial', in the sense that it proves true from true. An example of this kind of construction can be seen in the first derivation of Figure 4-1 on the preceding page.

Theorem 4.1.8. *Every atomic flow is associated with some derivation.*

Proof. First, we show that, for any atom a and formula contexts $\xi\{\ \}$ and $\zeta\{\ \}$, there exists a derivation

$$\begin{array}{c}(\xi\{\mathsf{t}\}\wedge\zeta\{a\})\vee\mathsf{t}\\ \|_{\{\mathsf{s},\mathsf{m}\}}\\ (\xi\{a\}\wedge\zeta\{\mathsf{f}\})\vee\mathsf{t}\end{array},$$

in other words we can 'move' the atom a from the context $\xi\{\ \}$ to the context $\zeta\{\ \}$ by using a derivation whose flow contains no vertices:

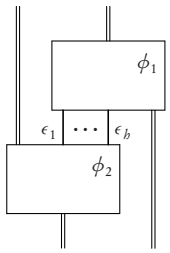
$$= \frac{\left[\begin{array}{c} \mathsf{ss}\dfrac{\xi\{\mathsf{t}\}\wedge\zeta\{a\}}{\xi\{a\}\vee\zeta\{\mathsf{f}\}} \\ = \dfrac{}{(\xi\{a\}\wedge\mathsf{t})\vee(\mathsf{t}\wedge\zeta\{\mathsf{f}\})} \\ \mathsf{m}\dfrac{}{[\xi\{a\}\vee\mathsf{t}]\wedge[\mathsf{t}\vee\zeta\{\mathsf{f}\}]} \\ = \dfrac{}{[\xi\{a\}\vee\mathsf{t}]\wedge[\zeta\{\mathsf{f}\}\vee\mathsf{t}]} \\ \mathsf{s}\dfrac{}{([\xi\{a\}\vee\mathsf{t}]\wedge\zeta\{\mathsf{f}\})\vee\mathsf{t}} \end{array}\right] \vee \quad \mathsf{t}}{= \dfrac{\left[\mathsf{s}\dfrac{\zeta\{\mathsf{f}\}\wedge[\xi\{a\}\vee\mathsf{t}]}{(\zeta\{\mathsf{f}\}\wedge\xi\{a\})\vee\mathsf{t}}\ \vee\ \mathsf{t}\right]}{(\xi\{a\}\wedge\zeta\{\mathsf{f}\})\vee\mathsf{t}}}.$$

This construction can be used repeatedly to build the derivation Ψ, for $h \geqslant 0$:

$$\begin{array}{c}(\xi\{\mathsf{t}\}\cdots\{\mathsf{t}\}\wedge\zeta\{a_1\}\cdots\{a_h\})\vee\mathsf{t}\\ \Psi\|_{\{\mathsf{s},\mathsf{m}\}}\\ (\xi\{a_1\}\cdots\{a_h\}\wedge\zeta\{\mathsf{f}\}\cdots\{\mathsf{f}\})\vee\mathsf{t}\end{array}.$$

We can now prove the theorem by induction on the number of vertices of a given flow ϕ. The cases where ϕ only has zero or one vertex are trivial. Let us then suppose that ϕ has

more than one vertex; then ϕ can be considered as composed of two flows ϕ_1 and ϕ_2, each with fewer vertices than ϕ, as follows:

$\phi =$  ,

where $h \geqslant 0$ (this can possibly be done in many different ways). By the inductive hypothesis, there exist derivations $\begin{matrix}\gamma\\ \Phi_1\|\\ \zeta\{a_1^{\epsilon_1}\}\cdots\{a_h^{\epsilon_h}\}\end{matrix}$ and $\begin{matrix}\xi\{a_1^{\epsilon_1}\}\cdots\{a_h^{\epsilon_h}\}\\ \Phi_2\|\\ \delta\end{matrix}$ whose flows are, respectively, ϕ_1 and ϕ_2. Using these, we can build

$$\left[\begin{matrix}\left(\xi\{\mathsf{t}\}\cdots\{\mathsf{t}\} \;\wedge\; \begin{matrix}\gamma\\ \Phi_1\|\\ \zeta\{a_1^{\epsilon_1}\}\cdots\{a_h^{\epsilon_h}\}\end{matrix}\right) \vee \mathsf{t}\\ \Psi\|\\ \left(\begin{matrix}\xi\{a_1^{\epsilon_1}\}\cdots\{a_h^{\epsilon_h}\}\\ \Phi_2\|\\ \delta\end{matrix} \;\wedge\; \zeta\{\mathsf{f}\}\cdots\{\mathsf{f}\}\right) \vee \mathsf{t}\end{matrix}\right] ,$$

whose flow is ϕ. □

Remark 4.1.9. From Proposition 4.1.2 on page 29 and Theorem 4.1.8 on the previous page we can conclude that: Given a derivation Φ and a flow ϕ, deciding if ϕ is associated with Φ, is equivalent to deciding if two flows are isomorphic. This will never be an issue in this thesis as we all the flows we will consider are associated with the relevant derivations by construction.

Notation 4.1.10. Given a derivation Φ, an atom occurrence a in Φ and the flow ϕ of Φ, then, whenever we write a^{ϵ} or a^{ψ}, we mean that there is a subflow ψ of ϕ containing the edge ϵ, such that a is mapped to ϵ.

We will now see how this notation might be useful when selectively substituting for atom occurrences. For example, let us suppose that we are given the following associated derivation and flow:

$$\Phi = \left[\begin{matrix}(a \wedge \mathsf{f}) \vee \left(a \wedge \dfrac{\mathsf{f}}{\bar{a}}\right)\\ \mathsf{m}\dfrac{\dfrac{a \vee a}{a} \wedge = \dfrac{\mathsf{f} \vee \bar{a}}{\bar{a}}}{\mathsf{f}} \quad \vee \quad \bar{a}\end{matrix}\right] \qquad \text{and} \qquad  .$$

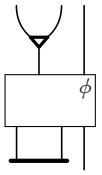
We can then distinguish between the three occurrences of $\bar{a}$ that are mapped to edge 1 and the one that is not, as in

$$\Phi = \left[\mathsf{m} \dfrac{\dfrac{(a \wedge \mathsf{f}) \vee \left(a \wedge \dfrac{\mathsf{f}}{\bar{a}^1}\right)}{\dfrac{a \vee a}{a} \wedge = \dfrac{\mathsf{f} \vee \bar{a}^1}{\bar{a}^1}} \quad \vee \quad \bar{a}}{\mathsf{f}} \right] \quad ;$$

we can also substitute for these occurrences, for example by $\{\bar{a}^1/\mathsf{f}\}$; such a situation occurs in the proof of Theorem 6.2.3 on page 48. Note that simply substituting f for $\bar{a}^1$ would invalidate this derivation because it would break the cut and weakening instances; however, the proof of Theorem 6.2.3 specifies how to fix the broken cut instance and Proposition 4.1.11 specifies how to fix the broken weakening.

We generalise this labelling mechanism to boxes. For example, we can use a different representation of the flow of Φ to individuate two classes a^ϕ and $\bar{a}^\phi$ of atom occurrences, as follows:

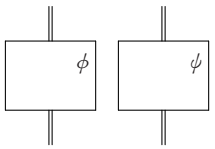
$$\Phi = \left[\mathsf{m} \dfrac{\dfrac{(a \wedge \mathsf{f}) \vee \left(a \wedge \dfrac{\mathsf{f}}{\bar{a}^\phi}\right)}{\dfrac{a \vee a}{a^\phi} \wedge = \dfrac{\mathsf{f} \vee \bar{a}^\phi}{\bar{a}^\phi}} \quad \vee \quad \bar{a}^\phi}{\mathsf{f}} \right] \qquad \text{and} \qquad$$



.

This notation is used in Proposition 4.1.11, where we define how we can, in certain cases, substitute formulae in place of atom occurrences. This technique is used in Theorem 6.1.3 on page 44, Theorem 6.2.3 on page 48 and Theorem 6.4.4 on page 61.

Proposition 4.1.11. *Given a derivation* $\begin{matrix}\alpha\\ \Phi\\ \beta\end{matrix}\!\!\parallel\mathsf{SKS}$, *let its associated flow have shape*

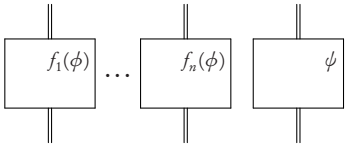


,

such that ϕ is a connected component whose edges are each associated with occurrences of the atom a; then, for any formula γ, there exists a derivation

$$\begin{matrix}\alpha\{a^\phi/\gamma\}\\ \Psi\\ \beta\{a^\phi/\gamma\}\end{matrix}\!\!\parallel\mathsf{SKS}$$

whose associated flow is



where n is the number of atom occurrences in γ; moreover, the size of Ψ depends linearly on the size of Φ and quadratically on the size of γ.

Proof. We can proceed by structural induction on Φ. For every formula in Φ we substitute a^ϕ with γ. Since all the edges in ϕ are mapped to from a (and not $\bar{a}$), we know that all the vertices in ϕ are mapped to from instances of $\mathsf{ac}\downarrow$, $\mathsf{ac}\uparrow$, $\mathsf{aw}\downarrow$ and $\mathsf{aw}\uparrow$. We substitute every instance of $\mathsf{ac}\downarrow$, $\mathsf{ac}\uparrow$, $\mathsf{aw}\downarrow$ and $\mathsf{aw}\uparrow$ where a^ϕ appears, by $\mathsf{c}\downarrow$, $\mathsf{c}\uparrow$, $\mathsf{w}\downarrow$, $\mathsf{w}\uparrow$, respectively, with γ in the place of a^ϕ. The result then follows by Lemma 2.3.11 on page 17 and Lemma 2.3.12 on page 17. □

Notation 4.1.12. The derivation Ψ obtained in the proof of Proposition 4.1.11 on the preceding page is denoted $\Phi\{a^\phi/\gamma\}$.

Remark 4.1.13. The notion of substitution can be extended to allow ϕ to contain interaction and cut vertices, but we shall not need that in this thesis.

4.2 A Normal Form of Derivation

In this section we introduce the ai-*decomposed form* of a derivation. The reason for introducing this normal form is that we will often find it convenient to assume that interaction instances appear at the top and cut instances appear at the bottom of a derivation. The important features of this normal form is that a derivation can be transformed into ai-decomposed form without changing its atomic flow, and without significantly changing its size.

Definition 4.2.1. Given two derivations

$$\begin{array}{c}\alpha\\ \Phi\|\\ \beta\end{array} \quad \text{and} \quad \Psi = \begin{array}{c}\left(\dfrac{\mathsf{t}}{a_1 \vee \bar{a}_1} \wedge \cdots \wedge \dfrac{\mathsf{t}}{a_n \vee \bar{a}_n} \wedge \alpha\right)\\ \|_{\mathsf{SKS}\setminus\{\mathsf{ai}\downarrow,\mathsf{ai}\uparrow\}}\\ \left[\beta \vee \dfrac{b_m \wedge \bar{b}_m}{\mathsf{f}} \vee \cdots \vee \dfrac{b_1 \wedge \bar{b}_1}{\mathsf{f}}\right]\end{array},$$

for some atoms $a_1, \ldots, a_n, b_1, \ldots, b_m$, such that Φ and Ψ have isomorphic flows, we say that Ψ is an ai-*decomposed form of* Φ.

Convention 4.2.2. Given a derivation Φ and an ai-decomposed form of Φ:

$$\begin{array}{c}\left(\dfrac{\mathsf{t}}{a_1 \vee \bar{a}_1} \wedge \cdots \wedge \dfrac{\mathsf{t}}{a_n \vee \bar{a}_n} \wedge \dfrac{\mathsf{t}}{c_1 \vee \bar{c}_1} \wedge \cdots \wedge \dfrac{\mathsf{t}}{c_k \vee \bar{c}_k} \wedge \alpha\right)\\ \|_{\mathsf{SKS}\setminus\{\mathsf{ai}\downarrow,\mathsf{ai}\uparrow\}}\\ \left[\beta \vee \dfrac{d_l \wedge \bar{d}_l}{\mathsf{f}} \vee \cdots \vee \dfrac{d_1 \wedge \bar{d}_1}{\mathsf{f}} \vee \dfrac{b_m \wedge \bar{b}_m}{\mathsf{f}} \vee \cdots \vee \dfrac{b_1 \wedge \bar{b}_1}{\mathsf{f}}\right]\end{array},$$

we sometimes want to single out only some of the interaction or cut instances. We therefore also call the following derivation an ai-decomposed form of Φ:

$$= \frac{\left(\dfrac{\mathsf{t}}{a_1 \vee \bar{a}_1} \wedge \cdots \wedge \dfrac{\mathsf{t}}{a_n \vee \bar{a}_n} \wedge \alpha\right)}{\left(\left[a_1 \vee \bar{a}_1\right] \wedge \cdots \wedge \left[a_n \vee \bar{a}_n\right] \wedge \dfrac{\mathsf{t}}{c_1 \vee \bar{c}_1} \wedge \cdots \wedge \dfrac{\mathsf{t}}{c_k \vee \bar{c}_k} \wedge \alpha\right)}$$
$$\|_{\mathsf{SKS}\setminus\{\mathsf{ai}\downarrow,\mathsf{ai}\uparrow\}}$$
$$= \frac{\left[\beta \vee \dfrac{d_l \wedge \bar{d}_l}{\mathsf{f}} \vee \cdots \vee \dfrac{d_1 \wedge \bar{d}_1}{\mathsf{f}} \vee \left(b_m \wedge \bar{b}_m\right) \vee \cdots \vee \left(b_1 \wedge \bar{b}_1\right)\right]}{\left[\beta \vee \dfrac{b_m \wedge \bar{b}_m}{\mathsf{f}} \vee \cdots \vee \dfrac{b_1 \wedge \bar{b}_1}{\mathsf{f}}\right]}.$$

Theorem 4.2.3. *Given a derivation* Φ*, an* ai*-decomposed form of* Φ *whose size depends at most cubically on the size of* Φ *can be constructed.*

Proof. Using Lemma 2.3.8 on page 16 apply, from top-to-bottom and left-to-right, the following transformations to each of the interaction and cut instances in Φ:

$$\xi\left\{\begin{array}{c}\alpha\\ \Psi\|\\ \dfrac{\mathsf{t}}{a \vee \bar{a}}\\ \Psi'\|\\ \beta\end{array}\right\} \quad \rightarrow \quad {}_{\mathsf{ss}\uparrow}\frac{\left(\begin{array}{cc}\dfrac{\mathsf{t}}{a \vee \bar{a}} & \wedge \quad \begin{array}{c}\alpha\\ \Psi\|\\ \xi\{\mathsf{t}\}\end{array}\end{array}\right)}{\begin{array}{c}\xi\left[a \vee \bar{a}\right]\\ \Psi'\|\\ \beta\end{array}} \qquad \text{and} \qquad \xi\left\{\begin{array}{c}\alpha\\ \Psi\|\\ \dfrac{a \wedge \bar{a}}{\mathsf{f}}\\ \Psi'\|\\ \beta\end{array}\right\} \quad \rightarrow \quad {}_{\mathsf{ss}\downarrow}\frac{\begin{array}{c}\alpha\\ \Psi\|\\ \xi\left(a \wedge \bar{a}\right)\end{array}}{\left[\begin{array}{cc}\xi\{\mathsf{f}\} & \\ \Psi'\| & \vee \quad \dfrac{a \wedge \bar{a}}{\mathsf{f}}\\ \beta & \end{array}\right]}$$

to obtain an ai-decomposed form of Φ. The size of the ai-decomposed form obtained in this way depends at most cubically on the size of Φ, since, by Lemma 2.3.8 on page 16, each of the transformations increase the size of the derivation at most quadratically and the number of transformations is bounded by the size of Φ. □

Remark 4.2.4. The only reason to insist on performing the transformations in the proof of Theorem 4.2.3 in a certain order is to ensure that the resulting derivation is unique. The uniqueness is useful in the following definition.

Definition 4.2.5. Given a derivation Φ, the ai-decomposed form of Φ obtained as described in the proof of Theorem 4.2.3 is called *the* (*canonical*) ai*-decomposed form of* Φ.

Chapter 5

Normal Forms

In this chapter we see the first use of atomic flows, namely to define normal forms of derivations. Traditionally, in Gentzen-style formalisms, a derivation in normal form is a cut-free derivation. The notion of cut-freeness is a syntactic notion, which does not translate nicely to the more general deep-inference formalisms.

In both Gentzen-style formalisms and deep-inference formalisms, the cut can be considered horizontal composition of two proofs. We make two observations: 1) deep-inference formalisms are symmetric in the vertical axis, whereas Gentzen-style formalisms are not; and 2) in order for the cut to be admissible from deep-inference derivations the symmetry must be broken, to correspond to the asymmetry of Gentzen-style formalisms. In particular, the cut is only admissible from proofs and not derivations.

These observations prompted us to look for a generalisation of cut elimination that work for all deep-inference derivations. Furthermore, since we are in the business of designing new formalisms, we wanted normal forms based on geometric notions which would be as syntax independent as possible.

We defined normal forms based on the causal dependency between structural inference rule instances. Atomic flows contain (by design) exactly the information needed in order to define normal forms in this way.

We call our generalisation of cut elimination *streamlining* and we describe it in terms of atomic flows. Intuitively, if we consider identities and weakenings to be the 'creators' of atom occurrences, and cuts and coweakening as the 'destroyers' of atom occurrences, then an atomic flow is streamlined if no atom is first created and then destroyed. The shape of a streamlined atomic flow is given in case (4) of Definition 5.0.1 on the next page.

The most challenging aspect of streamlining is the elimination of paths from interaction to cut vertices. For this reason, we define the notion of a *weakly streamlined* atomic flow, in case (3) of Definition 5.0.1. An atomic flow is weakly streamlined if it contains no path from an interaction to a cut vertex. This is the topic of Chapter 6 on page 42.

A path can be eliminated by removing the edges that make up the path. However, we

might imagine a situation where an edge belongs to two paths, one we want to eliminate and one we want to keep. An atomic flow is in *simple form*, if this situation does not occur. One approach to eliminating paths from a flow is to transform it into simple form and then eliminating the edges connecting interaction and cut vertices.

Sometimes, the elimination of edges mapped to by an atom a might interfere with the elimination of edges mapped to from the atom $\bar{a}$. For this reason, we find it convenient to define special cases of simple form and weakly streamlined, where for every pair of dual atoms the edges mapped to from one of them are ignored. These are cases (1) and (2) of Definition 5.0.1.

In summary, the intuition behind each of the normal forms in Definition 5.0.1 is:

1. a flow is in simple form with respect to a given polarity assignment, if all the edes with a positive polarity assignment can be partitioned into two classes, the ones that belong to paths connecting interaction and cut vertices (the rightmost box markde with a + in the below figure) and the ones that do not (the four leftmost boxes marked with + in the below figure);

2. a flow is weakly streamlined with respect to a given polarity assignment, if there are no edges with a possitive polarity assignment in paths from interaction cut to vertices;

3. a flow is weakly streamlined if it contains no paths from interaction to cut vertices;

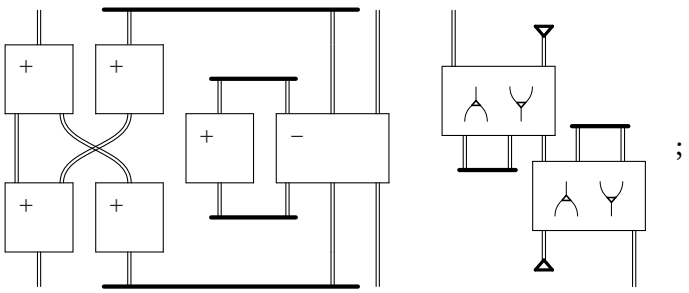
4. a weakly streamlined flow is streamlined if it contains no paths from interaction (resp., cut) to coweakening (resp., weakening) vertices, or from weakening to coweakening vertices;

5. a streamlined flow is *super streamlined* if it contains no paths from (co)weakening to (co)contraction vertices; and

6. a super streamlined flow is *hyper streamlined* if it contains no path whose first edge is an upper edge of a cocontraction vertex and last edge is the lower edge of a contraction vertex.

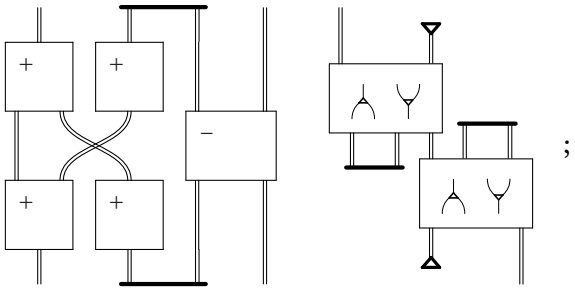
Definition 5.0.1. An atomic flow is

1. *in simple form with respect to the polarity assignment* π if it can be represented as



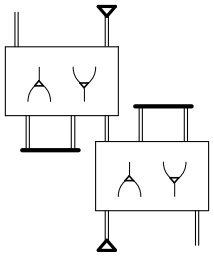
;

2. *weakly streamlined with respect to the polarity assignment* π if it can be represented as



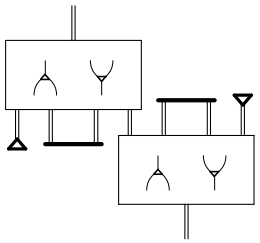
;

3. *weakly streamlined* if it can be represented as



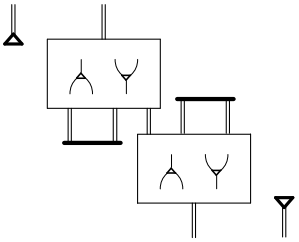
;

4. *streamlined* if it can be represented as



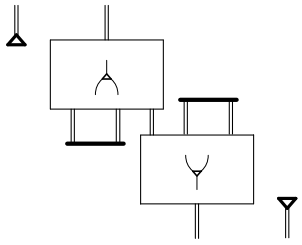
;

5. *super streamlined* if it can be represented as



; and

6. *hyper streamlined* if it can be represented as

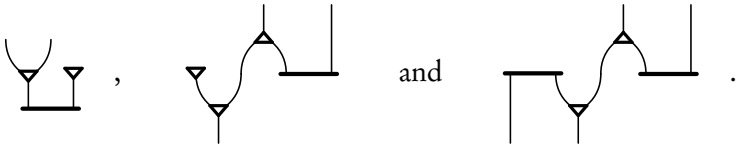


.

Definition 5.0.2. A derivation with associated flow ϕ is *in simple form with respect to* (*the atom*) a, if π is a polarity assignment for ϕ, such that the edges in ϕ mapped to from occurrences of a have a positive polarity, and the restriction of ϕ to a is in simple form with respect to π.

Definition 5.0.3. A derivation with associated flow ϕ is *weakly streamlined* (resp., *streamlined*, *super streamlined* and *hyper streamlined*) if ϕ is *weakly streamlined* (resp., *streamlined*, *super streamlined* and *hyper streamlined*). The derivation is *weakly streamlined with respect to* (*the atom*) a, if π is a polarity assignment for ϕ, such that the edges in ϕ mapped to from occurrences of a have a positive polarity, and the restriction of ϕ to a is weakly streamlined with respect to π.

Example 5.0.4. The first flow is weakly streamlined, the other two are hyper streamlined:



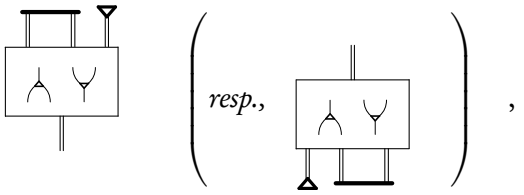
We now state some facts whose proofs are immediate from Definition 5.0.1 on page 37.

Proposition 5.0.5. *Given a polarity assignment π, a flow that is weakly streamlined with respect to both π and $\bar{\pi}$ is weakly streamlined.*

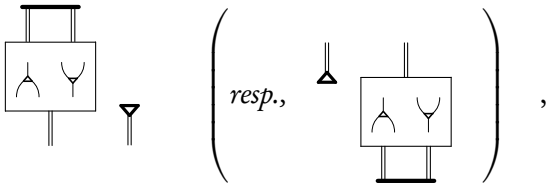
Proposition 5.0.6. *A streamlined flow with no pair of upper (resp., lower) edges such that there is an* ai*-path between them, contains no cut (resp., axiom) vertices.*

The following proposition makes the connection between cut elimination and streamlining. We consider the special case of atomic flows of proofs, *i.e.*, atomic flows without upper edges, and observe that a streamlined proof is cut free and a hyper streamlined proof is a proof in the system $\mathsf{SKS} \setminus \{\mathsf{ai}{\uparrow}, \mathsf{ac}{\uparrow}, \mathsf{aw}{\uparrow}\}$.

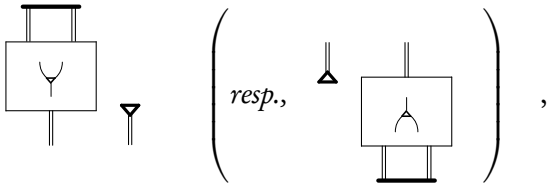
Proposition 5.0.7. *Given an atomic flow with no upper (resp., lower) edges, it can be represented as*

1. 

   if it is streamlined;

2. 

if it is super streamlined; and

3. 

,

if it is hyper streamlined.

Part III

Normalisation

Chapter 6

Global Reductions

In this and the next chapter we see the second use of atomic flows: Controlling normalisation of derivations. Conventional wisdom teaches us that normalisation is a delicate process, and that a careful design of inference rules is necessary in order to obtain it. Atomic flows were designed to describe normal forms, by removing a lot of information about the inference rules, it is therefore surprising that they contain enough information to design normalisation procedures.

There are two kinds of flow reductions: global and local ones. Global reductions rewrites the entire flow: normally, two or more slightly altered copies of a flow are connected together. Local reductions substitutes a bounded subflow in a flow by another subflow that fits in the context.

Alternatively, as suggested by François Lamarche, we could talk about external and internal instead of global and local reductions. This guides the intuition in the sense that the global reductions never 'look inside' the flows they work on. The size of the flows being copied is unbounded, however, the alterations to each of the copies are bounded, and it always happens at the 'outside' of the flow.

This chapter is dedicated to the most challenging part of normalisation: obtaining weakly streamlined derivations through global reductions. The process is non-confluent, and at first glance it increases the size of derivations exponentially. However, a second surprise was the fact that we are able to design procedures for weakly streamlining which only grow derivations quasipolynomially.

We will define several 'atomic flow reductions' which can be combined in different ways in order to obtain normalisation. Since we aim to produce derivations on normal forms, and not only their atomic flows, we find it convenient to define *operators* on derivations in terms of the flow reductions. It is important to note that we could have performed all the procedures purely in terms of atomic flows. The final results about derivations would follow from the 'soundness' of the flow reductions. We chose to be a bit more explicit and provide the derivations directly.

Definition 6.0.1. An (*atomic-flow*) *reduction* R is a binary relation on the set of atomic flows,

such that $\phi \; R \; \psi$ if

1. there is a one-to-one map, u, from the upper edges of ϕ to the upper edges of ψ;
2. there is a one-to-one map, l, from the lower edges of ϕ to the lower edges of ψ; and
3. for every polarity assignment π for ϕ, there is a polarity assignment π' for ψ such that $\pi'(u(\epsilon)) = \pi(\epsilon)$ and $\pi'(l(\iota)) = \pi(\iota)$, for any upper edge ϵ and any lower edge ι of ϕ'.

We call ϕ a *redex* and ψ a *contractum* of R.

Convention 6.0.2. Given a reduction R and two flows ϕ and ψ, such that $\phi \; R \; \psi$, we indicate the bijections u and l by labeling the upper (resp., lower) edge $u(\epsilon)$ (resp., $l(\epsilon)$) of ψ by ϵ, for every upper (resp., lower) edge ϵ of ϕ.

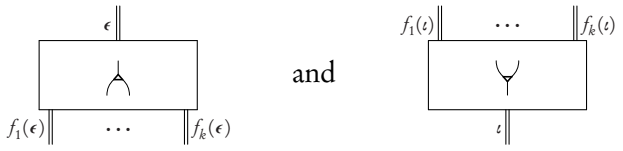
It is important to notice the difference in notation, between the bijections between edges belonging to isomorphic flows, and the bijections between upper and lower edges in a redex/contractum pair. For an example of these two conventions being used simultaneously, see Definition 6.1.1 on the next page.

Definition 6.0.3. A reduction R is *sound* if, for every ϕ and ψ, such that $\phi \; R \; \psi$, and for every derivation Φ with flow ϕ, there is a derivation Ψ with atomic flow ψ such that Φ and Ψ have the same premiss and conclusion; in this case we write $\Phi \; R \; \Psi$.

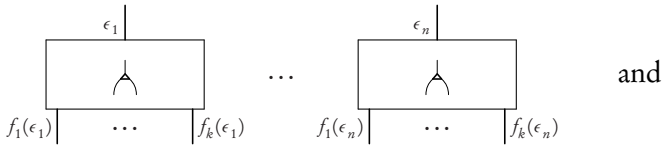
Convention 6.0.4. We provide constructive soundness proofs for every reduction in this chapter, so from now on, for any reduction R and derivation Φ, when we write $\Phi \; R \; \Psi$, we mean that Ψ is the derivation obtained form Φ in the soundness proof of R.

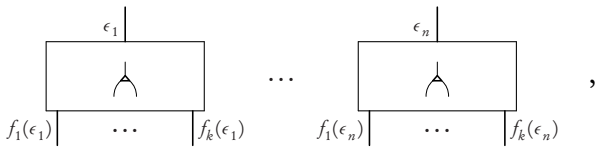
Remark 6.0.5. Alternatively, as suggested by François Lamarche, instead of saying that a reduction is sound, we could say that it is *liftable*. The constructive soundness proofs which we will see later on, then becomes *liftings*.

Convention 6.0.6. To avoid ambiguity in Definition 6.1.1 on the following page, Definition 6.2.1 on page 47, Definition 6.3.1 on page 53 and Definition 6.4.1 on page 59 we have established the following convention: Let $\epsilon = \epsilon_1, \ldots, \epsilon_n$, $\iota = \iota_1, \ldots, \iota_m$, $\epsilon' = \epsilon'_1, \ldots, \epsilon'_n$ and $\iota' = \iota'_1, \ldots, \iota'_m$, then, when we write



we mean



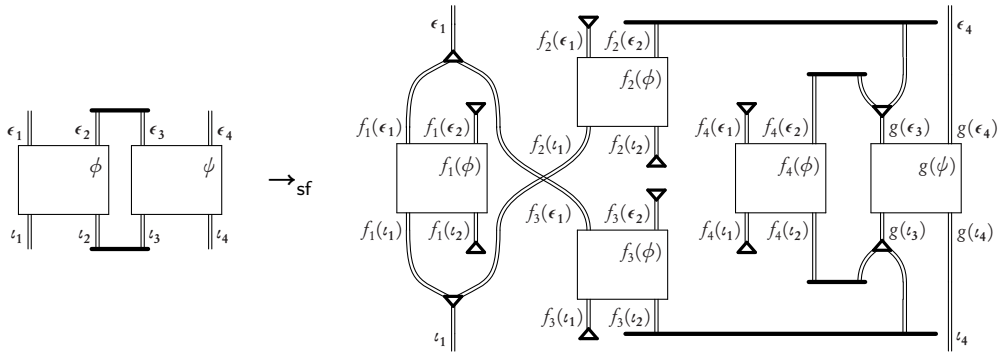


respectively. In other words, edges are not connected in unexpected ways.

6.1 Simplifier

Consider a flow ϕ' with polarity assignemnt π, such that ϕ is the subflow of ϕ' containing all the edges with a positive polarity assignment. We can observe that ϕ contains four types of paths: 1) paths from $\top$ to $\bot$; 2) paths from an interaction vertex to $\bot$; 3) paths from $\top$ to a cut vertex; and 4) paths from an interaction vertex to a cut vertex. We can turn ϕ' into simple form with respect to π if we can make sure that no edge belongs to both a path of type 1) and a path of type 4). In the following reduction, we achieve this by making four copies of ϕ each of which only contains one of the above types of paths.

Definition 6.1.1. We define the reduction $\to_{\mathsf{sf}}$ (where sf stands for *simple form*) as follows, for any two flows ϕ and ψ that do not contain any interaction or cut vertices:



Remark 6.1.2. The reduction $\to_{\mathsf{sf}}$ would still be sound if we removed the restriction on the flows ϕ and ψ in Definition 6.1.1. However, such a reduction would no longer correspond to the intuition described above.

Theorem 6.1.3. *Reduction $\to_{\mathsf{sf}}$ is sound; moreover if $\Phi \to_{\mathsf{sf}} \Psi$, then the size of Ψ depends at most polynomially on the size of Φ.*

Proof. Let Φ be a derivation with flow ϕ', such that $\phi' \to_{\mathsf{sf}} \psi'$. We show that there exists a derivation Ψ with flow ψ' and with the same premiss and conclusion as Φ. In the following, we refer to the figure in Definition 6.1.1.

Assume all the edges in ϕ are mapped to from occurrences of the atoms $a_1, \ldots, a_n$, and

let

$$\left(\frac{\mathsf{t}}{a_1^{\phi}\vee\bar{a}_1^{\psi}}\wedge\cdots\wedge\frac{\mathsf{t}}{a_n^{\phi}\vee\bar{a}_n^{\psi}}\wedge\alpha\right)$$
$$\Phi'\Vert\{\mathsf{ai}{\downarrow},\mathsf{ai}{\uparrow}\}$$
$$\left[\beta\vee\frac{a_n^{\phi}\wedge\bar{a}_n^{\psi}}{\mathsf{f}}\vee\cdots\vee\frac{a_1^{\phi}\wedge\bar{a}_1^{\psi}}{\mathsf{f}}\right]\quad,$$

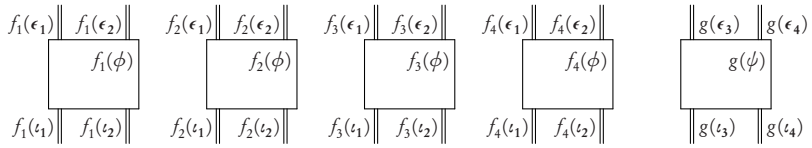
be the ai-decomposed form of Φ.

We show several intermediate derivations which will be used to build Ψ. To make it easier to verify the flow of Ψ, we will, through a slight misuse of notation, label the atom occurrences of the intermediate derivations to indicate what atomic flow each atom occurrence will map to, once the derivations are combined to create Ψ.

Consider the substitution

$$\sigma=\{a_1^{\phi}/([a_1^{f_1(\phi)}\vee a_1^{f_2(\phi)}]\wedge[a_1^{f_3(\phi)}\vee a_1^{f_4(\phi)}]),\ldots,a_n^{\phi}/([a_n^{f_1(\phi)}\vee a_n^{f_2(\phi)}]\wedge[a_n^{f_3(\phi)}\vee a_n^{f_4(\phi)}])\}\,.$$

We can then obtain, by Proposition 4.1.11 on page 33, the derivation $\Phi'\sigma$ with flow



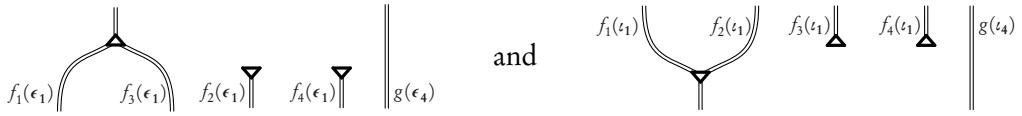
For every $1\leqslant i\leqslant n$, there exist derivations

$$\frac{a_i}{\left[a_i^{f_1(\phi)}\vee\frac{\mathsf{f}}{a_i^{f_2(\phi)}}\right]\wedge\left[a_i^{f_3(\phi)}\vee\frac{\mathsf{f}}{a_i^{f_4(\phi)}}\right]}\qquad\text{and}\qquad\frac{a_i^{f_1(\phi)}\vee a_i^{f_2(\phi)}}{a_i}\wedge\left[\frac{a_i^{f_3(\phi)}}{\mathsf{t}}\vee\frac{a_i^{f_4(\phi)}}{\mathsf{t}}\right]\quad,$$

which allow us to build

$$\begin{array}{c}\alpha\\ \Psi_{\top}\Vert\\ \alpha\sigma\end{array}\qquad\text{and}\qquad\begin{array}{c}\beta\sigma\\ \Psi_{\bot}\Vert\\ \beta\end{array}\quad,$$

with flows



respectively. Furthermore, for every $1\leqslant i\leqslant n$, there exist derivations

$$\Psi_{\mathsf{t},i}=\cfrac{\cfrac{\cfrac{\mathsf{t}}{a_i^{f_2(\phi)}\vee\bar{a}_i}\wedge\cfrac{\mathsf{t}}{a_i^{f_4(\phi)}\vee\bar{a}_i}}{\mathsf{s}\,a_i^{f_2(\phi)}\wedge[a_i^{f_4(\phi)}\vee\bar{a}_i]}}{\mathsf{s}\,(a_i^{f_2(\phi)}\wedge a_i^{f_4(\phi)})\vee\bar{a}_i}\vee\bar{a}_i}{=\left(\left[\frac{\mathsf{f}}{a_i^{f_1(\phi)}}\vee a_i^{f_2(\phi)}\right]\wedge\left[\frac{\mathsf{f}}{a_i^{f_3(\phi)}}\vee a_i^{f_4(\phi)}\right]\right)\vee\frac{\bar{a}_i\vee\bar{a}_i}{\bar{a}_i^{g(\psi)}}}$$

and

$$\Psi_{\mathsf{f},i} = \dfrac{\left(\left[\dfrac{a_i^{f_1(\phi)}}{\mathsf{t}} \vee \dfrac{a_i^{f_2(\phi)}}{\mathsf{t}}\right] \wedge \left[a_i^{f_3(\phi)} \vee a_i^{f_4(\phi)}\right]\right) \wedge \dfrac{\bar{a}_i^{g(\psi)}}{\bar{a}_i \wedge \bar{a}_i}}{\mathsf{s}\dfrac{\left[a_i^{f_3(\phi)} \vee a_i^{f_4(\phi)}\right] \wedge \bar{a}_i}{a_i^{f_3(\phi)} \vee \left(a_i^{f_4(\phi)} \wedge \bar{a}_i\right)} \wedge \bar{a}_i}{\mathsf{s}\dfrac{}{\dfrac{a_i^{f_3(\phi)} \wedge \bar{a}_i}{\mathsf{f}} \vee \dfrac{a_i^{f_4(\phi)} \wedge \bar{a}_i}{\mathsf{f}}}}$$

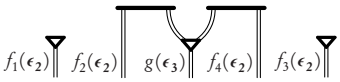
,

which allow us to build

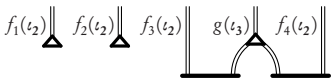
$$\Psi_{\mathsf{t}} = \begin{pmatrix} \mathsf{t} & & \mathsf{t} \\ \Psi_{\mathsf{t},1}\| & \wedge \cdots \wedge & \Psi_{\mathsf{t},n}\| \\ [a_1^{\phi} \vee \bar{a}_1^{\psi}]\sigma & & [a_n^{\phi} \vee \bar{a}_n^{\psi}]\sigma \end{pmatrix} \quad \text{and}$$

$$\Psi_{\mathsf{f}} = \begin{bmatrix} (a_n^{\phi} \wedge \bar{a}_n^{\psi})\sigma & & (a_1^{\phi} \wedge \bar{a}_1^{\psi})\sigma \\ \Psi_{\mathsf{f},n}\| & \vee \cdots \vee & \Psi_{\mathsf{f},1}\| \\ \mathsf{f} & & \mathsf{f} \end{bmatrix},$$

with flows



and



,

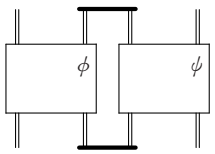
respectively. Combining these derivations we can build

$$\Psi = \begin{array}{c} \alpha \\ \Psi_{\mathsf{t}} \wedge \Psi_{\top}\| \\ \left(\left[a_1^{\phi} \vee \bar{a}_1^{\phi}\right] \wedge \cdots \wedge \left[a_n^{\phi} \vee \bar{a}_n^{\phi}\right] \wedge \alpha\right)\sigma \\ \Phi'\sigma\| \\ \left[\beta \vee \left(a_n^{\phi} \wedge \bar{a}_n^{\psi}\right) \vee \cdots \vee \left(a_1^{\phi} \wedge \bar{a}_1^{\psi}\right)\right]\sigma \\ \Psi_{\bot} \vee \Psi_{\mathsf{f}}\| \\ \beta \end{array},$$

with the desired flow.

We know that the size of $\Phi'\sigma$ depends at most polynomially on the size of Φ by Theorem 4.2.3 on page 35 and Proposition 4.1.11 on page 33, and it is straightforward to observe that the sizes of Ψ_{t}, $\Psi_{\top}$, Ψ_{f} and $\Psi_{\bot}$ depend at most linearly on the size of Φ, so the size of Ψ depends at most polynomially on the size of Φ. □

Definition 6.1.4. The *Simplifier*, Si, is an operator whose arguments are distinct and pairwise non-dual atoms $a_1, \ldots, a_n$ and a derivation Φ, with flow



,

such that all the edges in ϕ are mapped to from occurrences of $a_1, \ldots, a_n$ and no edges in ψ are mapped to from occurrences of $a_1, \ldots, a_n$. We then define $\mathsf{Si}(\Phi, a_1, \ldots, a_n)$ to be such that $\Phi \rightarrow_{\mathsf{sf}} \mathsf{Si}(\Phi, a_1, \ldots, a_n)$, where ϕ and ψ are the flows, by the same names, shown in Definition 6.1.1 on page 44.

Proposition 6.1.5. *Given distinct and pairwise non-dual atoms $a_1, \ldots, a_n$, and a derivation Φ,*

1. *$\mathsf{Si}(\Phi, a_1, \ldots, a_n)$ is in simple form with respect to $a_1, \ldots, a_n$;*

2. *for any atom b, if Φ is weakly streamlined with respect to b, then $\mathsf{Si}(\Phi, a_1, \ldots, a_n)$ is weakly streamlined with respect to b; and*

3. *the size of $\mathsf{Si}(\Phi, a_1, \ldots, a_n)$ depends at most polynomially on the size of Φ.*

Proof. In the following we refer to the figure in Definition 6.1.1 on page 44:

- by case (1) of Definition 5.0.1 on page 37;

- by studying the flows in Definition 6.1.1 we can observe that for every path from an interaction vertex to a cut vertex in the atomic flow of $\mathsf{Si}(\Phi, a_1, \ldots, a_n)$ whose edges are mapped to from occurrences of b, there is a path from an interaction vertex to a cut vertex in the flow of Φ whose edges are mapped to from occurrences of b; and

- by Theorem 6.1.3 on page 44.

□

6.2 Isolated Subflow Removal

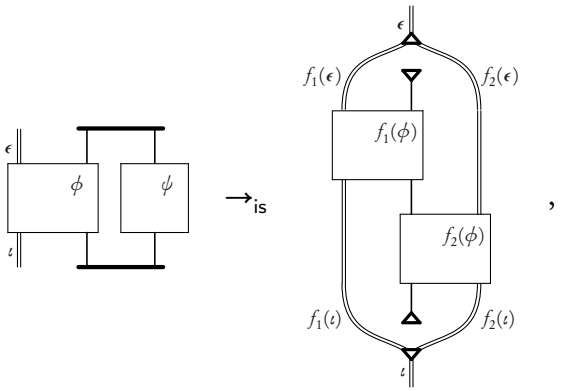
Given a derivation Φ in simple form with respect to an atom a, the operator, ISR, defined in this section produces a derivation with the same premiss and conclusion as Φ, which is weakly streamlined with respect to a.

We will see later how a derivation containing occurrences of n atoms can be weakly streamlined by two applications of Si and n applications of ISR. This is the most basic procedure for obtaining a weakly streamlined derivation, in particular it only deals with one atom at a time. In the following sections we will see how we can deal with several atoms in parallel.

The operator is defined in terms of the following flow reduction.

Definition 6.2.1. We define the reduction $\rightarrow_{\mathsf{is}}$ (where is stands for *isolated subflow*) as follows, for any flow ϕ and any connected component ψ that does not contain interaction or cut

vertices:



where we call the evidenced interaction (resp., cut) vertex $\nu_{\mathsf{ai}\downarrow}$ (resp., $\nu_{\mathsf{ai}\uparrow}$).

Remark 6.2.2. The condition on the flow ψ in Definition 6.2.1 on the previous page ensures that all the edges in ψ are mapped to from occurrences of the same atom. However, the reduction would still be sound if, at the expense of a slightly more verbose soundness proof, we relaxed the condition to say that there is a path from $\nu_{\mathsf{ai}\downarrow}$ to $\nu_{\mathsf{ai}\uparrow}$.

Theorem 6.2.3. *Reduction* $\rightarrow_{\mathsf{is}}$ *is sound; moreover, if* $\Phi \rightarrow_{\mathsf{is}} \Psi$*, then the size of* Ψ *depends polynomially on the size of* Φ*.*

Proof. Let Φ be a derivation with flow ϕ', such that $\phi' \rightarrow_{\mathsf{is}} \psi'$. We show that there exists a derivation Ψ with flow ψ' and with the same premiss and conclusion as Φ. In the following, we refer to the figure in Definition 6.2.1 on the preceding page.

Since ψ is connected, we assume, by Convention 4.2.2 on page 34, that the following derivation is an ai-decomposed form of Φ:

$$\begin{array}{c} \left(\dfrac{\mathsf{t}}{a^{\psi} \vee \bar{a}} \wedge \alpha\right) \\ \Phi' \| \\ \left[\beta \vee \dfrac{a^{\psi} \wedge \bar{a}}{\mathsf{f}}\right] \end{array} ,$$

for some atom a and formulae α and β.

We obtain the two derivations Φ_{t} and Φ_{f} from Φ' as follows:

$$\Phi_{\mathsf{t}} = \begin{array}{c} [\mathsf{t} \vee \bar{a}] \wedge \alpha \\ \Phi'\{a^{\psi}/\mathsf{t}\} \| \\ \beta \vee \bar{a} \end{array} \quad \text{and} \quad \Phi_{\mathsf{f}} = \begin{array}{c} \bar{a} \wedge \alpha \\ \Phi'\{a^{\psi}/\mathsf{f}\} \| \\ \beta \vee (\mathsf{f} \wedge \bar{a}) \end{array} .$$

Since ψ is connected and contains no interaction or cut vertices, the mapping from all the occurrences a^{ψ} to edges of ψ is surjective. Hence, we know that both derivation Φ_{t} and

Φ_f have a flow isomorphic to ϕ. We combine Φ_t and Φ_f to get the desired derivation Ψ with flow ψ' and the same premiss and conclusion as Φ:

$$\Psi = \dfrac{\dfrac{\dfrac{\alpha}{\left(\left[\mathsf{t} \vee \dfrac{\mathsf{f}}{\bar{a}}\right] \wedge \alpha\right) \wedge \alpha}\,\mathsf{c}\uparrow}{\beta \vee \bar{a}}\,\mathsf{s}}{\beta \vee \left[\beta \vee \left(\mathsf{f} \wedge \dfrac{\bar{a}}{\mathsf{t}}\right)\right]}\,\mathsf{c}\downarrow \quad .$$

We know that the size of Φ_t and the size of Φ_f depend polynomially on the size of Φ by Theorem 4.2.3 on page 35 and Proposition 4.1.11 on page 33, and that the size of Ψ depends at most quadratically on the size of α and β by Lemma 2.3.12 on page 17, so the size of Ψ depends polynomially on the size of Φ. □

We now show the basic properties of $\rightarrow_{\mathsf{is}}$. Namely, that the reduction does not create any 'new' interaction or cut vertices, and that it does not create any 'new' paths between interaction or $\top$ and cut or $\bot$ vertices.

Lemma 6.2.4. *In the following we refer to the names given in Definition 6.2.1 on page 47. Given two flows ϕ and ψ, such that $\phi \rightarrow_{\mathsf{is}} \psi$ then, given an interaction (resp., cut) vertex v in ψ, there is an interaction (resp., cut) vertex v' in ϕ, such that*

1. *$v = f_1(v')$ or $v = f_2(v')$;*
2. *if there is a path from v to $\bot$ (resp., $\top$), then there is a path from v' to $\bot$ (resp., $\top$); and*
3. *if there is a cut (resp., interaction) vertex $\hat{v}$ in ψ, such that there is a path from v to $\hat{v}$, then there is a cut (resp., interaction) vertex $\hat{v}'$ in ϕ, such that $\hat{v} = f_1(\hat{v}')$ or $\hat{v} = f_2(\hat{v}')$, or $\hat{v}' = v_{\mathsf{ai}\uparrow}$ (resp., $\hat{v}' = v_{\mathsf{ai}\downarrow}$); and there is a path from v' to $\hat{v}'$.*

Proof. We consider each case separately:

1. by definition;
2. any path from v to $\bot$ (resp., $\top$) must contain an edge ϵ, such that, for some lower (resp., upper) edge ϵ' of ϕ, $f_1(\epsilon') = \epsilon$ or $f_2(\epsilon') = \epsilon$. Hence, there is a path from v' to $\bot$ (resp., $\top$); and
3. we have to consider two cases:
   - $v = f_1(v')$ and $\hat{v} = f_1(\hat{v}')$, or $v = f_2(v')$ and $\hat{v} = f_2(\hat{v}')$, then there is a path from v' to $\hat{v}'$; or

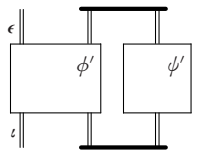
- $\nu = f_1(\nu')$ and $\hat{\nu} = f_2(\hat{\nu}')$ (resp., $\nu = f_2(\nu')$ and $\hat{\nu} = f_1(\hat{\nu}')$),then there is a path from ν' to $\nu_{\mathsf{ai}\uparrow}$ (resp., $\nu_{\mathsf{ai}\downarrow}$).

□

Definition 6.2.5. The *Isolated Subflow Remover*, ISR, is an operator whose arguments are an atom a and a derivation Φ that is in simple form with respect to a. If Φ is weakly streamlined with respect to a, then $\mathsf{ISR}(\Phi, a) = \Phi$; otherwise, consider the following ai-decomposed form of Φ:

$$\begin{array}{c} \left(\dfrac{\mathsf{t}}{a\psi' \vee \bar{a}} \wedge \cdots \wedge \dfrac{\mathsf{t}}{a\psi' \vee \bar{a}} \wedge \alpha \right) \\ \Phi' \| \\ \left[\beta \vee \dfrac{a\psi' \wedge \bar{a}}{\mathsf{f}} \vee \cdots \vee \dfrac{a\psi' \wedge \bar{a}}{\mathsf{f}} \right] \end{array},$$

with flow

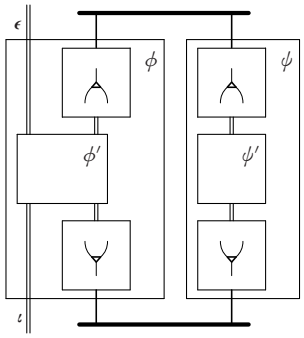


where ψ' is the juxtaposition of all the isolated subflows mapped to from occurrences of a in Φ. Consider the derivation

$$\Psi = \begin{array}{c} \left(\begin{array}{c} \dfrac{\mathsf{t}}{a \vee \bar{a}} \\ \|\{\mathsf{c}\uparrow\} \\ [a \vee \bar{a}] \wedge \cdots \wedge [a \vee \bar{a}] \end{array} \right) \wedge \alpha \\ \Phi' \| \\ \left[\begin{array}{c} (a \wedge \bar{a}) \vee \cdots \vee (a \wedge \bar{a}) \\ \|\{\mathsf{c}\downarrow\} \\ \beta \vee \dfrac{a \wedge \bar{a}}{\mathsf{f}} \end{array} \right] \end{array},$$

with flow

$$\psi'' = \quad$$



.

We then define $\mathsf{ISR}(\Phi, a)$ to be such that $\Psi \rightarrow_{\mathsf{is}} \mathsf{ISR}(\Phi, a)$, where ϕ and ψ are the flows, by the same names, shown in Definition 6.2.1 on page 47.

Proposition 6.2.6. *Given an atom a and a derivation Φ that is in simple form with respect to a,*

1. $\mathsf{ISR}(\Phi, a)$ *is weakly streamlined with respect to a;*
2. *for any atom b,*
   (a) *if Φ is weakly streamlined with respect to b, then $\mathsf{ISR}(\Phi, a)$ is weakly streamlined with respect to b, and*
   (b) *if b is not the dual of a and Φ is in simple form with respect to b, then $\mathsf{ISR}(\Phi, a)$ is in simple form with respect to b; and*
3. *the size of $\mathsf{ISR}(\Phi, a)$ depends polynomially on the size of Φ.*

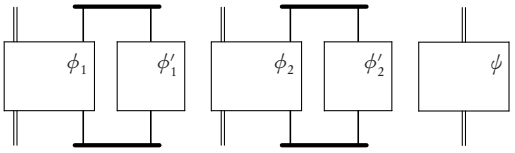
Proof. If Φ is weakly streamlined with respect to a, the result is trivial. Assume Φ is not weakly streamlined with respect to a, and let ϕ, ψ, ϕ', ψ' and ψ'' be the flows given in Definition 6.2.5 on the preceding page, then

1. by definition there is no path in ϕ from an interaction to a cut vertex whose edges are mapped to from instances of a. By Lemma 6.2.4 on page 49, we know that if there is a path from an interaction to a cut vertex in the flow of $\mathsf{ISR}(\Phi, a)$ whose edges are mapped to from instances of a, then there must be a path from an interaction to a cut vertex in ϕ whose edges are mapped to from instances a. Hence, the statement follows by contradiction;
2. (a) if the flow of $\mathsf{ISR}(\Phi, a)$ contains a path from an interaction vertex to a cut vertex whose edges are mapped to from instances of b, then, by Lemma 6.2.4 on page 49, there is a path from an interaction vertex to a cut vertex in ϕ, so also in ϕ', whose edges are mapped to from instances of b. Hence, the statement follows by contradiction; and
   (b) if there is an interaction (resp., cut) vertex v and a cut (resp., interaction) vertex $\hat{v}$ in the flow of $\mathsf{ISR}(\Phi, a)$ such that there is a path from v to $\hat{v}$ and a path from v to $\bot$ (resp., $\top$), both of whose edges are mapped to from instances of b, then, by Lemma 6.2.4 on page 49, there is an interaction (resp., cut) vertex v' and a cut (resp., interaction) vertex $\hat{v}'$ in ϕ such that there is a path from v to $\hat{v}$ and a path from v to $\bot$ (resp., $\top$), both of whose edges are mapped to from instances of b. Furthermore, since we can assume that b is not a or $\bar{a}$, ϕ restricted to b equals ϕ' restricted to b. Hence, the statement follows by contradiction.
3. the statement follows by Theorem 6.2.3 on page 48.

□

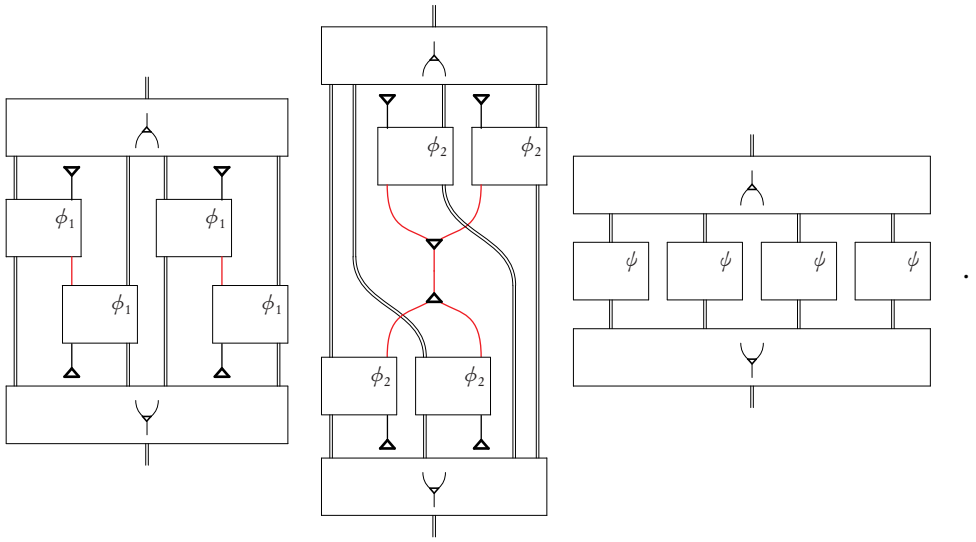
We now give an example of an application of ISR. In particular we want to show its inherent non-confluency.

Example 6.2.7. Given a derivation Φ where the atoms a_1 and a_2 occur, such that the flow associated with Φ is



,

and where all the edges in ϕ_1 (resp., ϕ'_1) are mapped to from a_1 (resp., $\bar{a}_1$) and all the edges in ϕ_2 (resp, ϕ'_2) are mapped to from a_2 (resp., $\bar{a}_2$), and there are no edges in ψ that are mapped to from a_1 or a_2, then the flow associated with $\mathsf{ISR}(\mathsf{ISR}(\Phi, a_1), a_2)$ is the following flow (where indications of the different isomorphisms are left out):



.

We marked some edges in red to point out the fundamental difference between the subflow containing ϕ_1 and the subflow containing ϕ_2. Note that, in order to improve readability, we have removed a contraction and a cocontraction vertex from the subflow containing ϕ_2, by using *weakening reductions*. Weakening reductions are defined in Definition 7.0.8 on page 75.

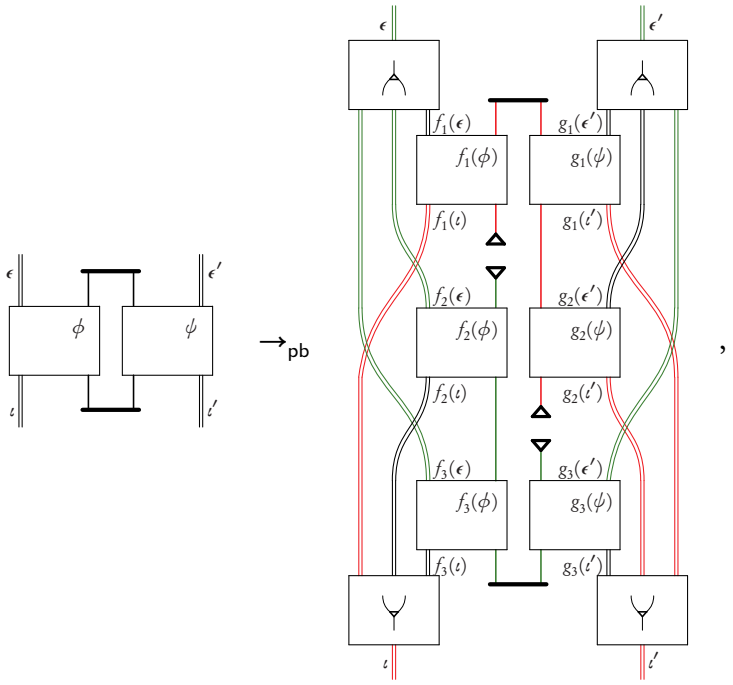
6.3 Path Breaker

Given a derivation Φ and an atom a, the operator, PB, defined in this section produces a derivation with the same premiss and conclusion as Φ, which is weakly streamlined with respect to both a and $\bar{a}$. This operator is a strict improvement over ISR, since it does not require the input derivation to be in simple form, and it deals with the dual atoms in parallel. We will see later how a derivation containing n atoms can be weakly streamlined by $n/2$ applications of PB.

A variation of the results in this section is also presented in the paper *Breaking Paths in Atomic Flows for Classical Logic* [GGS10], which was coauthored with Alessio Guglielmi and Lutz Straßburger.

The operator is defined in terms of the following flow reduction.

Definition 6.3.1. We define the reduction $\to_{\mathsf{pb}}$ (where pb stands for *path breaker*) as follows, for any two flows ϕ and ψ:



where we call the evidenced interaction (resp., cut) vertex in the redex $\nu'_{\mathsf{ai}\downarrow}$ (resp., $\nu'_{\mathsf{ai}\uparrow}$) and the evidenced interaction (resp., cut) vertex in the contractum $\nu_{\mathsf{ai}\downarrow}$ (resp., $\nu_{\mathsf{ai}\uparrow}$); and where there is a path from $\nu'_{\mathsf{ai}\downarrow}$ to $\nu'_{\mathsf{ai}\uparrow}$.

Theorem 6.3.2. *Reduction $\to_{\mathsf{pb}}$ is sound; moreover, if $\Phi \to_{\mathsf{pb}} \Psi$, then the size of Ψ depends polynomially on the size of Φ.*

Proof. Let Φ be a derivation with flow ϕ', such that $\phi' \to_{\mathsf{pb}} \psi'$. We show that there exists a derivation Ψ with flow ψ' and with the same premiss and conclusion as Φ. In the following, we refer to the figure in Definition 6.3.1.

Since the evidenced interaction and cut vertices belong to the same connected component, we assume, by Convention 4.2.2 on page 34, that the following derivation is an ai-decomposed form of Φ:

$$\begin{array}{c} \left(\dfrac{\mathsf{t}}{a^{\phi} \vee \bar{a}^{\psi}} \wedge \alpha\right) \\ \Phi' \| \\ \left[\beta \vee \dfrac{a^{\phi} \wedge \bar{a}^{\psi}}{\mathsf{f}}\right] \end{array} ,$$

for some atom a and formulae α and β.

We combine three copies of Φ' to obtain the desired derivation Ψ with flow ψ' and the same premiss and conclusion as Φ:

$$\Psi \quad = \quad \dfrac{\dfrac{\dfrac{\mathsf{c}\downarrow\dfrac{\alpha}{\left(\dfrac{\mathsf{t}}{a^{f_1(\phi)}\vee\bar{a}^{g_1(\psi)}}\wedge\alpha\right)}}{\left[\beta\vee\left(\dfrac{a^{f_1(\phi)}}{\mathsf{t}}\wedge\bar{a}^{g_1(\psi)}\right)\right]}\,\Phi'\parallel\,\wedge\;\mathsf{c}\uparrow\dfrac{\alpha}{\alpha\wedge\alpha}}{\mathsf{s}}}{\left[\begin{array}{c}\left(\left[\dfrac{\mathsf{f}}{a^{f_2(\phi)}}\vee\bar{a}^{g_2(\psi)}\right]\wedge\alpha\right)\\ \beta\;\vee\;\Phi'\parallel\\ \left[\beta\vee\left(a^{f_2(\phi)}\wedge\dfrac{\bar{a}^{g_2(\psi)}}{\mathsf{t}}\right)\right]\end{array}\right]}\,\wedge\;\alpha}{\mathsf{s}}}{\mathsf{c}\downarrow\dfrac{\beta\vee\beta}{\beta}\;\vee\;\dfrac{\left(\left[a^{f_3(\phi)}\vee\dfrac{\mathsf{f}}{\bar{a}^{g_3(\psi)}}\right]\wedge\alpha\right)}{\left[\beta\vee\dfrac{a^{f_3(\phi)}\wedge\bar{a}^{g_3(\psi)}}{\mathsf{f}}\right]}\,\Phi'\parallel}{\mathsf{c}\downarrow}}{\beta}\quad .$$

We know that the size of Φ' depends at most cubically on the size of Φ by Theorem 4.2.3 on page 35, and that the size of Ψ depends at most quadratically on the size of α and β by Lemma 2.3.12 on page 17, so Ψ depends polynomially on the size of Φ. □

We now show the basic properties of $\rightarrow_{\mathsf{pb}}$. Namely, that the reduction does not create any 'new' interaction or cut vertices, that it does not create any 'new' paths between interaction or $\top$ and cut or $\bot$ vertices, and that it breaks all the paths between the evidenced interaction and cut vertices.

Lemma 6.3.3. *In the following we refer to the names given in Definition 6.3.1 on the previous page. Given two flows ϕ and ψ, such that $\phi \rightarrow_{\mathsf{pb}} \psi$, then, given an interaction (resp., cut) vertex ν in ψ, there is an interaction (resp., cut) vertex ν' in ϕ, such that*

1. *for some $1 \leqslant i \leqslant 3$, $\nu = f_i(\nu')$ or $\nu = g_i(\nu')$, or $\nu = \nu_{\mathsf{ai}\downarrow}$ and $\nu' = \nu'_{\mathsf{ai}\downarrow}$ (resp., $\nu = \nu_{\mathsf{ai}\uparrow}$ and $\nu' = \nu'_{\mathsf{ai}\uparrow}$);*
2. *if there is a path from ν to $\bot$ (resp., $\top$) in ψ, then there is a path from ν' to $\bot$ (resp., $\top$) in ϕ;*
3. *if there is a cut (resp., interaction) vertex $\hat{\nu}$ in ψ, such that there is a path from ν to $\hat{\nu}$ in ψ, then there is a cut (resp., interaction) vertex $\hat{\nu}'$ in ϕ, such that, for some $1 \leqslant i \leqslant 3$, $\hat{\nu} = f_i(\hat{\nu}')$*

or $\hat{v} = g_i(\hat{v}')$*, or* $\hat{v} = v_{\mathsf{ai}\uparrow}$ *and* $\hat{v}' = v'_{\mathsf{ai}\uparrow}$ *(resp.,* $\hat{v} = v_{\mathsf{ai}\downarrow}$ *and* $\hat{v}' = v'_{\mathsf{ai}\downarrow}$*); and there is a path from* v' *to* $\hat{v}'$ *in* ϕ*; and*

4. *there is no path from* $v_{\mathsf{ai}\downarrow}$ *to* $v_{\mathsf{ai}\uparrow}$.

Proof. We consider each case separately:

1. by definition;

2. any path from v to $\bot$ (resp., $\top$) in ψ must contain an edge ϵ, such that, for some lower (resp., upper) edge ϵ' of ϕ and some $1 \leqslant i \leqslant 3$, $f_i(\epsilon') = \epsilon$ or $g_i(\epsilon') = \epsilon$. Hence, there is a path from v' to $\bot$ (resp., $\top$) in ϕ;

3. we have to consider two cases:

   (a) for some $1 \leqslant i \leqslant 3$, $v = f_i(v')$ and $\hat{v} = f_i(\hat{v}')$, or $v = g_i(v')$ and $\hat{v} = g_i(\hat{v}')$, then there is a path from v' to $\hat{v}'$ in ϕ, or

   (b) $v = g_1(v')$ and $\hat{v} = g_2(\hat{v}')$, or $v = f_2(v')$ and $\hat{v} = f_3(\hat{v}')$ (resp., $v = g_2(v')$ and $\hat{v} = g_1(\hat{v}')$, or $v = f_3(v')$ and $\hat{v} = f_2(\hat{v}')$), then there is a path from v' to $v'_{\mathsf{ai}\uparrow}$ (resp., $v'_{\mathsf{ai}\downarrow}$) in ϕ; and

4. in Definition 6.3.1 we have coloured the edges that might occur in paths from $v_{\mathsf{ai}\downarrow}$ in red and paths that might occur in path to $v_{\mathsf{ai}\uparrow}$ in green. Since the red and the green edges never coincide, there can be no paths from $v_{\mathsf{ai}\downarrow}$ to $v_{\mathsf{ai}\uparrow}$.

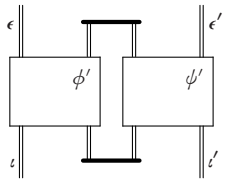
□

Definition 6.3.4. The *Path Breaker*, PB, is an operator whose arguments are an atom a and a derivation Φ. If Φ is weakly streamlined with respect to both a and $\bar{a}$, then $\mathsf{PB}(\Phi, a) = \Phi$; otherwise, consider the following ai-decomposed form of Φ:

$$\left(\frac{\mathsf{t}}{a^{\psi} \vee \bar{a}} \wedge \cdots \wedge \frac{\mathsf{t}}{a^{\psi} \vee \bar{a}} \wedge \alpha\right) \quad \Phi' \| \quad \left[\beta \vee \frac{a^{\psi} \wedge \bar{a}}{\mathsf{f}} \vee \cdots \vee \frac{a^{\psi} \wedge \bar{a}}{\mathsf{f}}\right],$$

with flow

$$\phi'' = \quad ,$$

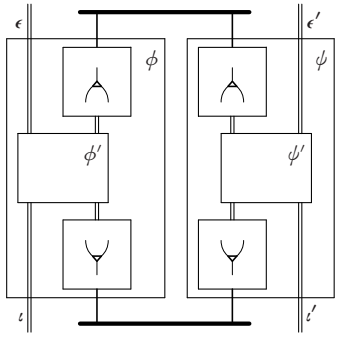


such that occurrences of a do not appear in an interaction or cut instance in Φ'. Consider the derivation

$$\Psi = \begin{bmatrix} \left(\begin{array}{c} \mathsf{t} \\ \hline a \vee \bar{a} \\ \| \{\mathsf{c}\uparrow\} \\ [a \vee \bar{a}] \wedge \cdots \wedge [a \vee \bar{a}] \end{array}\right) & \wedge & \alpha \\ \Phi' \| & & \\ \left[\begin{array}{cc} & (a \wedge \bar{a}) \vee \cdots \vee (a \wedge \bar{a}) \\ & \| \{\mathsf{c}\downarrow\} \\ \beta & \vee \quad a \wedge \bar{a} \\ & \hline & \mathsf{f} \end{array}\right] & \end{bmatrix},$$

with flow

$$\psi'' = \quad .$$



We then define $\mathsf{PB}(\Phi, a)$ to be such that $\Psi \rightarrow_{\mathsf{pb}} \mathsf{PB}(\Phi, a)$, where ϕ and ψ are the flows, by the same names, shown in Definition 6.3.1 on page 53.

Proposition 6.3.5. *Given an atom a and a derivation Φ,*

1. $\mathsf{PB}(\Phi, a)$ *is weakly streamlined with respect to both a and $\bar{a}$;*
2. *for any atom b, if Φ is weakly streamlined with respect to b, then* $\mathsf{PB}(\Phi, a)$ *is weakly streamlined with respect to b; and*
3. *the size of* $\mathsf{PB}(\Phi, a)$ *depends polynomially on the size of Φ.*

Proof. If Φ is weakly streamlined with respect to both a and $\bar{a}$, the result is trivial. Assume Φ is not weakly streamlined with respect to both a and $\bar{a}$, and let ϕ, ψ, ϕ', ψ', ϕ'' and ψ'' be the flows given in Definition 6.3.4 on the previous page and let $\nu_{\mathsf{ai}\downarrow}$ (resp., $\nu_{\mathsf{ai}\uparrow}$) be the evidenced interaction (resp., cut) vertex in ψ'', then

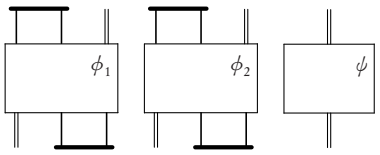
1. by Definition 6.3.4 all the paths from an interaction (resp., cut) vertex whose edges are mapped to from instances of a or $\bar{a}$ must start from $\nu_{\mathsf{ai}\downarrow}$ (resp., $\nu_{\mathsf{ai}\uparrow}$). The statement then follows by Lemma 6.3.3 on page 54;
2. if the flow of $\mathsf{PB}(\Phi, a)$ contains a path from an interaction vertex to a cut vertex whose edges are mapped to from instances of b, then, by Lemma 6.3.3, there is a path from an interaction to a cut vertex in ϕ or ψ, so also in ϕ' or ψ', whose edges are mapped to from instances of b. Hence, the statement follows by contradiction; and

3. the statement follows by Theorem 6.3.2 on page 53.

□

We now give an example of an application of PB. In particular we want to show its inherent non-confluency.

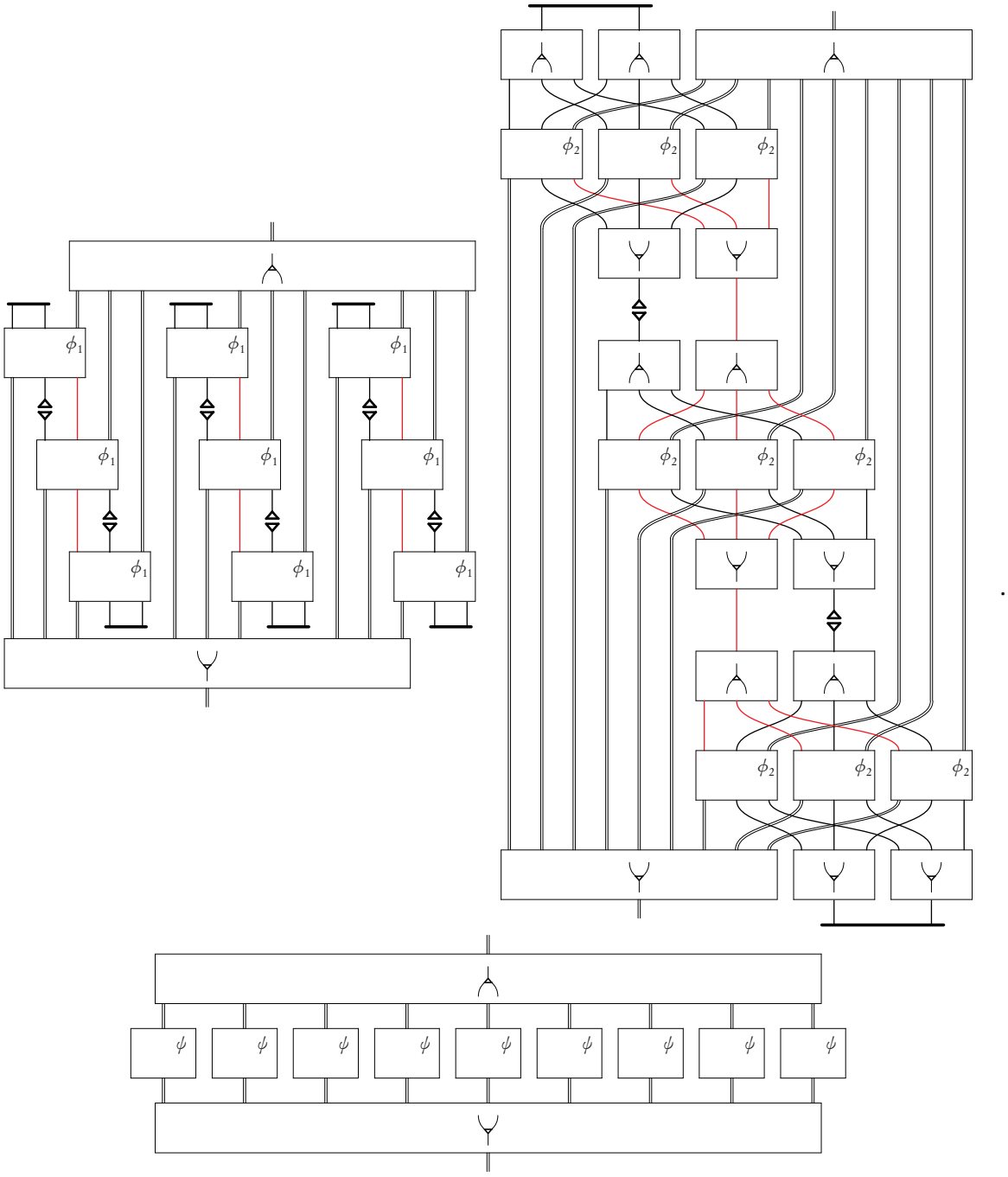
Example 6.3.6. Given a derivation Φ where the atoms a_1 and a_2 occur, such that the flow associated with Φ is



,

and where all the edges in ϕ_1 are mapped to from a_1 and $\bar{a}_1$, and all the edges in ϕ_2 are mapped to from a_2 and $\bar{a}_2$, and there are no edges in ψ that are mapped to from a_1 or a_2, then the flow associated with $\mathsf{PB}(\mathsf{PB}(\Phi,a_1),a_2)$ is the following flow (where indications of

the different isomorphisms are left out):



We marked some edges in red to point out the fundamental difference between the subflows containing ϕ_1 and the subflows containing ϕ_2.

6.4 Multiple Isolated Subflows Removal

With the operator ISR we can produce weakly streamilend derivations with respect to one atom at a time, with the operator PB we can produce weakly streamlined derivations with respect to two dual atoms in parallel. In this section we see an operator, MISR_n, for every $n > 0$, which is a generalisation of ISR, that can produce a weakly streamlined derivation with respect to n number of atoms in parallel, as long as they are pairwise non-dual.

We will see later how a derivation containing $2n$ atoms can be weakly streamlined by two applications of Si and two applications of MISR_n.

The results of this section, restricted to proofs, is also presented in the paper *A Quasipolynomial Cut-Elimination Procedure in Deep Inference via Atomic Flows and Threshold Formulae* [BGGP10], which was coauthored with Alessio Guglielmi, Paola Bruscoli and Michel Parigot.

The operator is defined in terms of the following flow reduction. Unlike the flow reductions of the preceding sections, we present here a reduction which depends on several parameters. It is important to note that these parameters are independent of the derivation to which we later apply the operator. In order to perform streamlining on an arbitrary number of atoms in parallel, we need find a class of atomic flows, η_k, which are used as a sort of sharing mechanism. We are at this stage not able to describe the flows η_k without relying on their corresponding derivations. For this reason, it might help the understanding of Definition 6.4.1 to refer to the derivation given in the proof of Theorem 6.4.4 on page 61.

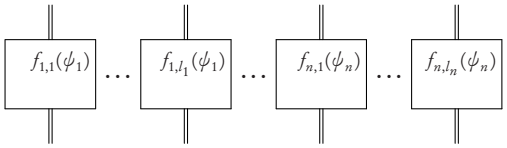
In Subsection 6.4.1 on page 65, we present one possible combination of valid parameters, which yields quasipolynomial (*i.e.* $n^{\mathsf{O}(\log n)}$) streamlining. We conjecture that by finding different parameters we will be able to obtain more efficient versions of this reduction. In particular, we hope to be able to obtain polynomial streamlining.

Definition 6.4.1. For every $n > 0$, given

- atoms $a_1, \ldots, a_n$;
- an $N > 0$;
- for $0 \leqslant k \leqslant N$, formulae $\gamma_{k,1}, \ldots, \gamma_{k,n}$, such that
  - $\gamma_{0,1} = \cdots = \gamma_{0,n} = \mathsf{t}$, and
  - $\gamma_{N,1} = \cdots = \gamma_{N,n} = \mathsf{f}$; and
- for $1 \leqslant k \leqslant N$, a derivation

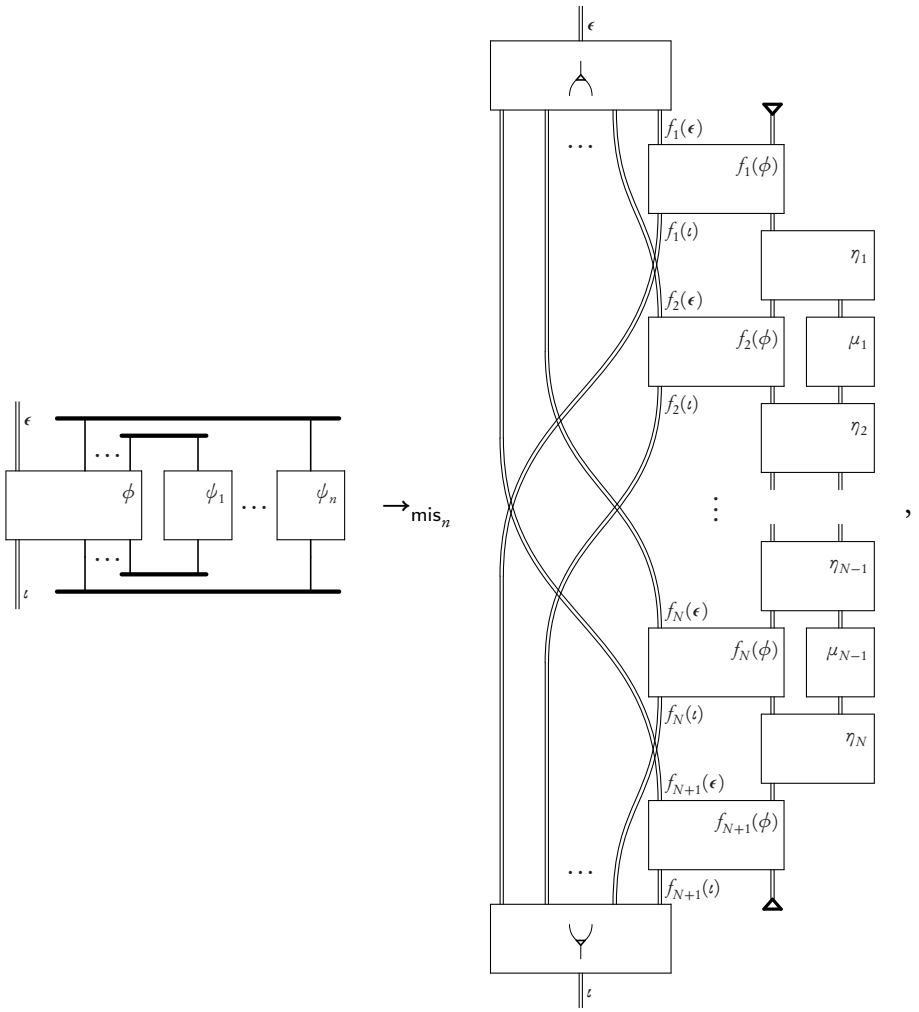
$$\Gamma_k \quad = \quad \begin{array}{c} \left(a_1 \wedge \gamma_{k-1,1}\right) \vee \cdots \vee \left(a_n \wedge \gamma_{k-1,n}\right) \\ \| \mathsf{SKS} \setminus \{\mathsf{ai}{\downarrow}, \mathsf{ai}{\uparrow}\} \\ \left[a_1 \vee \gamma_{k,1}\right] \wedge \cdots \wedge \left[a_n \vee \gamma_{k,n}\right] \end{array} ,$$

let, for $1 \leqslant k \leqslant N$, η_k be the flow of Γ_k, and let

$\mu_k =$ 

,

where, for $1 \leqslant i \leqslant n$, l_i is the number of atom occurrences in $\gamma_{k,i}$, we define the reduction $\rightarrow_{\mathsf{mis}_n}$ (where mis stands for *multiple isolated subflows*) as follows, for any flow ϕ and any connected components $\psi_1, \ldots, \psi_n$ that do not contain interaction or cut vertices:



,

where we call the evidenced interaction (resp., cut) vertices $\nu_{\mathsf{ai}\downarrow,1}, \ldots, \nu_{\mathsf{ai}\downarrow,n}$ (resp., $\nu_{\mathsf{ai}\uparrow,1}, \ldots, \nu_{\mathsf{ai}\uparrow,n}$).

Remark 6.4.2. The reduction $\rightarrow_{\mathsf{mis}_n}$ is denoted as if it only depends on n, this is a misuse of notation, and we will take it for granted that we also have the other parameters whenever we write $\rightarrow_{\mathsf{mis}_n}$.

Remark 6.4.3. If $N = 1$ and $\Gamma_1 = = \dfrac{a_1 \wedge \mathsf{t}}{a_1 \vee \mathsf{f}}$, then $\rightarrow_{\mathsf{mis}_1} = \rightarrow_{\mathsf{is}}$.

Theorem 6.4.4. *For every $n > 0$, reduction $\rightarrow_{\mathsf{mis}_n}$ is sound; moreover, if $\Phi \rightarrow_{\mathsf{mis}_n} \Psi$, then the size of Ψ depends linearly on N, polynomially on the size of Φ and at most polynomially on* $\max\{|\Gamma_1|, \ldots, |\Gamma_N|\}$.

Proof. Let Φ be a derivation with flow ϕ', such that $\phi' \rightarrow_{\mathsf{mis}_n} \psi'$. We show that there exists a derivation Ψ with flow ψ' and with the same premiss and conclusion as Φ. In the following, we refer to the figures in Definition 6.4.1 on page 59.

Since each of $\psi_1, \ldots, \psi_n$ is connected, we assume, by Convention 4.2.2 on page 34, that the following derivation is an ai-decomposed form of Φ:

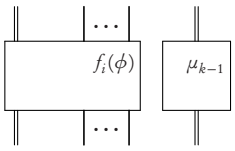
$$\dfrac{\dfrac{\mathsf{t}}{a_1 \vee \bar{a}_1^{\psi_1}} \wedge \dfrac{\mathsf{t}}{a_n \vee \bar{a}_n^{\psi_n}} \wedge \alpha}{\beta \vee \dfrac{a_1 \wedge \bar{a}_1^{\psi_1}}{\mathsf{f}} \vee \dfrac{a_n \wedge \bar{a}_n^{\psi_n}}{\mathsf{f}}}\ \Phi' \quad ,$$

for some atoms $a_1, \ldots, a_n$ (that, without loss of generality, we assume coincide with the atoms given in Definition 6.4.1 on page 59) and formulae α and β.

For every $0 \leqslant k \leqslant N$, we obtain the derivation Φ_k from Φ' as follows:

$$\Phi_k \quad = \quad \dfrac{\left[a_1 \vee \gamma_{k,1}\right] \wedge \cdots \wedge \left[a_n \vee \gamma_{k,n}\right] \wedge \alpha}{\Phi'\{\bar{a}_1^{\psi_1}/\gamma_{k,1}, \ldots, \bar{a}_n^{\psi_n}/\gamma_{k,n}\}} \\ \left[\beta \vee \left(a_1 \wedge \gamma_{k,1}\right) \vee \cdots \vee \left(a_n \wedge \gamma_{k,n}\right)\right]$$

Since each of $\psi_1, \ldots, \psi_n$ is a connected component and contains no interaction or cut vertices, the mapping from occurrences of $\bar{a}_i^{\psi_i}$ to edges of ψ_i is surjective. Hence, we know that Φ_k has flow



We combine $\Phi_0, \ldots, \Phi_N, \Gamma_1, \ldots, \Gamma_N$ to get the desired derivation Ψ with flow ψ' and the same

premiss and conclusion as Φ:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\alpha}{\left(\left[\dfrac{\mathsf{f}}{a_1}\vee\mathsf{t}\right]\wedge\cdots\wedge\left[\dfrac{\mathsf{f}}{a_n}\vee\mathsf{t}\right]\wedge\alpha\right)}\,\mathsf{c}\uparrow}}{\Phi_0\|}}{\begin{bmatrix}(a_1\wedge\mathsf{t})\vee\cdots\vee(a_n\wedge\mathsf{t})\\ \beta\ \vee\ \Gamma_1\|\\ [a_1\vee\gamma_{1,1}]\wedge\cdots\wedge[a_n\vee\gamma_{1,n}]\end{bmatrix}}\,\mathsf{s}}\ \wedge\ \mathsf{c}\uparrow\dfrac{\alpha}{\alpha\wedge\alpha}}{\begin{bmatrix}[a_1\vee\gamma_{1,1}]\wedge\cdots\wedge[a_n\vee\gamma_{1,n}]\wedge\alpha\\ \Phi_1\|\\ \beta\ \vee\ \begin{bmatrix}(a_1\wedge\gamma_{1,1})\vee\cdots\vee(a_n\wedge\gamma_{1,n})\\ \beta\ \vee\ \Gamma_2\|\\ [a_1\vee\gamma_{2,1}]\wedge\cdots\wedge[a_n\vee\gamma_{2,n}]\end{bmatrix}\end{bmatrix}}\,\mathsf{s}}\ \wedge\ \mathsf{c}\uparrow\dfrac{\alpha}{\alpha\wedge\alpha}}{\vdots}\,\mathsf{s}}{\begin{bmatrix}[a_1\vee\gamma_{N-1,1}]\wedge\cdots\wedge[a_n\vee\gamma_{N-1,n}]\wedge\alpha\\ \Phi_{N-1}\|\\ \mathsf{c}\downarrow\dfrac{\beta\vee\beta}{\beta}\ \vee\ \begin{bmatrix}(a_1\wedge\gamma_{N-1,1})\vee\cdots\vee(a_n\wedge\gamma_{N-1,n})\\ \beta\ \vee\ \Gamma_N\|\\ [a_1\vee\mathsf{f}]\wedge\cdots\wedge[a_n\vee\mathsf{f}]\end{bmatrix}\end{bmatrix}}\,\mathsf{s}}\ \wedge\ \alpha}{\begin{matrix}[a_1\vee\mathsf{f}]\wedge\cdots\wedge\left[a_n\vee\mathsf{f}\right]\wedge\alpha\\ \Phi_N\|\\ \mathsf{c}\downarrow\dfrac{\beta\vee\beta}{\beta}\ \vee\ \left[\beta\vee\left(\dfrac{a_1}{\mathsf{t}}\wedge\mathsf{f}\right)\vee\cdots\vee\left(\dfrac{a_n}{\mathsf{t}}\wedge\mathsf{f}\right)\right]\end{matrix}}\,\mathsf{c}\downarrow}{\beta}\ .$$

Since $\max\{\left|\gamma_{0,1}\right|,\ldots,\left|\gamma_{N,n}\right|\}$ is less than or equal to $\max\{|\Gamma_1|,\ldots,|\Gamma_N|\}$, we know that the size of Φ_0, …, Φ_N depend at most cubically on the size of Φ and at most quadratically on the size of $\max\{|\Gamma_1|,\ldots,|\Gamma_N|\}$ by Theorem 4.2.3 on page 35 and Proposition 4.1.11 on page 33, and that the size of Ψ depends at most cubically on the size of α and β by Lemma 2.3.12 on page 17, so the size of Ψ depends linearly on N, polynomially on the size of Φ and at most polynomially on the size of $\max\{|\Gamma_1|,\ldots,|\Gamma_N|\}$. □

We now show the basic properties of $\rightarrow_{\mathsf{mis}}$. Namely, that the reduction does not create any 'new' interaction or cut vertices, and that it does not create any 'new' paths between interaction or $\top$ and cut or $\bot$ vertices.

Lemma 6.4.5. *In the following we refer to the names given in Definition 6.4.1 on page 59. Given two flows ϕ and ψ and an $n > 0$, such that $\phi \rightarrow_{\mathsf{mis}_n} \psi$ then, given an interaction (resp., cut) vertex v in ψ, there is an interaction (resp., cut) vertex v' in ϕ, such that*

1. *for some $1 \leqslant i \leqslant N+1$, $v = f_i(v')$;*
2. *if there is a path from v to $\bot$ (resp., $\top$), then there is a path from v' to $\bot$ (resp., $\top$); and*

3. *if there is a cut (resp., interaction) vertex $\hat{v}$ in ψ, such that there is a path from v to $\hat{v}$, then there is a cut (resp., interaction) vertex $\hat{v}'$ in ϕ, such that, for some $1 \leqslant i \leqslant N+1$, $\hat{v} = f_i(\hat{v}')$, or, for some $1 \leqslant i \leqslant n$, $\hat{v}' = v_{\mathsf{ai}\uparrow,i}$ (resp., $\hat{v}' = v_{\mathsf{ai}\downarrow,i}$); and there is a path from v' to $\hat{v}'$.*

Proof. We consider each case separately:

1. the statement follows by definition;

2. any path from v to $\bot$ (resp., $\top$) must contain an edge ϵ, such that, for some lower (resp., upper) edge ϵ' of ϕ and some $1 \leqslant i \leqslant N+1$, $f_i(\epsilon') = \epsilon$. Hence, there is a path from v' to $\bot$ (resp., $\top$); and

3. we have to consider two cases:

   (a) for some $1 \leqslant i \leqslant N+1$, $v = f_i(v')$ and $\hat{v} = f_i(\hat{v}')$, then there is a path from v' to $\hat{v}'$; or

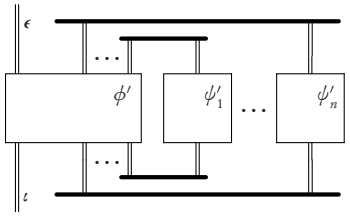
   (b) for some $1 \leqslant i < j \leqslant N+1$, $v = f_i(v')$ and $\hat{v} = f_j(\hat{v}')$ (resp., $v = f_j(v')$ and $\hat{v} = f_i(\hat{v}')$), then, for some $1 \leqslant i \leqslant n$, there is a path from v' to $v_{\mathsf{ai}\uparrow,i}$ (resp., $v_{\mathsf{ai}\uparrow,i}$).

□

Definition 6.4.6. For every $n > 0$, given the atoms, formulae and derivations described in Definition 6.4.1 on page 59, the *Multiple Isolated Subflow Remover*, MISR_n, is an operator whose arguments are atoms $a_1, \ldots, a_n$ (that, without loss of generality, we assume coincide with the atoms given in Definition 6.4.1), and a derivation Φ that is in simple form with respect to $a_1, \ldots, a_n$. If $n = 1$ and Φ is weakly streamlined with respect to a_1, then $\mathsf{MISR}_1(\Phi, a_1) = \Phi$; if $n > 1$ and, for some $1 \leqslant i \leqslant n$, Φ is weakly streamlined with respect to a_i, then $\mathsf{MISR}_n(\Phi, a_1, \ldots, a_n) = \mathsf{MISR}_{n-1}(\Phi, a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n)$; otherwise, consider the following ai-decomposed form of Φ:

$$\left(\frac{\mathsf{t}}{a_1^{\psi_1} \vee \bar{a}_1} \wedge \cdots \wedge \frac{\mathsf{t}}{a_1^{\psi_1} \vee \bar{a}_1} \wedge \cdots \wedge \frac{\mathsf{t}}{a_n^{\psi_n} \vee \bar{a}_n} \wedge \cdots \wedge \frac{\mathsf{t}}{a_n^{\psi_n} \vee \bar{a}_n} \wedge \alpha\right)$$
$$\Phi' \Big\| \qquad ,$$
$$\left[\beta \vee \frac{a_n^{\psi_n} \wedge \bar{a}_n}{\mathsf{f}} \vee \cdots \vee \frac{a_n^{\psi_n} \wedge \bar{a}_n}{\mathsf{f}} \vee \cdots \vee \frac{a_1^{\psi_1} \wedge \bar{a}_1}{\mathsf{f}} \vee \cdots \vee \frac{a_1^{\psi_1} \wedge \bar{a}_1}{\mathsf{f}}\right]$$

with flow

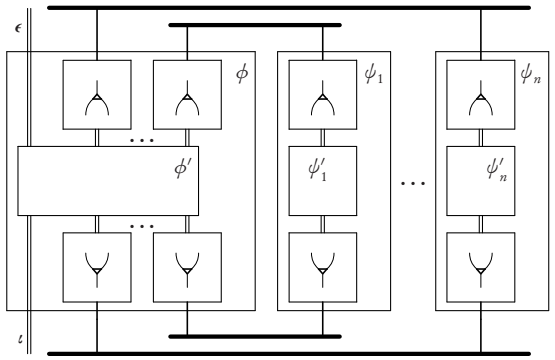


,

where, for $1 \leqslant i \leqslant n$, ψ_i is the juxtaposition of all the isolated subflows mapped to from occurrences of a_i in Φ. Consider the derivation

$$\Psi = \begin{array}{c} \left(\begin{array}{ccccc} \dfrac{\mathsf{t}}{a_1 \vee \bar{a}_1} & & \wedge \cdots \wedge & & \dfrac{\mathsf{t}}{a_n \vee \bar{a}_n} \\ \| \{\mathsf{c}\uparrow\} & & & & \| \{\mathsf{c}\uparrow\} \\ [a_1 \vee \bar{a}_1] \wedge \cdots \wedge [a_1 \vee \bar{a}_1] & & & & [a_n \vee \bar{a}_n] \wedge \cdots \wedge [a_n \vee \bar{a}_n] \end{array} \right) \\ \Phi' \| \\ \left[\begin{array}{ccccc} (a_n \wedge \bar{a}_n) \vee \cdots \vee (a_n \wedge \bar{a}_n) & & & & (a_1 \wedge \bar{a}_1) \vee \cdots \vee (a_1 \wedge \bar{a}_1) \\ \| \{\mathsf{c}\downarrow\} & & & & \| \{\mathsf{c}\downarrow\} \\ \beta \vee \dfrac{a_n \wedge \bar{a}_n}{\mathsf{f}} & & \vee \cdots \vee & & \dfrac{a_1 \wedge \bar{a}_1}{\mathsf{f}} \end{array}\right] \end{array},$$

with flow

$\psi'' =$ 

We then define $\mathsf{MISR}_n(\Phi, a_1, \ldots, a_n)$ to be such that $\Psi \rightarrow_{\mathsf{mis}} \mathsf{MISR}_n(\Phi, a_1, \ldots, a_n)$, where ϕ, $\psi_1, \ldots, \psi_n$ are the flows, by the same names, shown in Definition 6.4.1 on page 59.

Proposition 6.4.7. *Given the atoms, formulae and derivations described in Definition 6.4.1 on page 59, and atoms $a_1, \ldots, a_n$ and a derivation Φ that is in simple form with respect to $a_1, \ldots, a_n$,*

1. $\mathsf{MISR}_n(\Phi, a_1, \ldots, a_n)$ *is weakly streamlined with respect to $a_1, \ldots, a_n$;*
2. *for any atom b,*
   (a) *if Φ is weakly streamlined with respect to b, then $\mathsf{MISR}_n(\Phi, a_1, \ldots, a_n)$ is weakly streamlined with respect to b, and*
   (b) *if b is not the dual of any of $a_1, \ldots, a_n$ and Φ is in simple form with respect to b, then $\mathsf{MISR}_n(\Phi, a_1, \ldots, a_n)$ is in simple form with respect to b; and*
3. *the size of $\mathsf{MISR}_n(\Phi, a_1, \ldots, a_n)$ depends linearly on N, polynomially on the size of Φ, and at most polynomially on $\max\{|\Gamma_1|, \ldots, |\Gamma_N|\}$.*

Proof. If Φ is weakly streamlined with respect to some atom from $a_1, \ldots, a_n$, the result follows by induction. Assume Φ is not weakly streamlined with respect to any atom from $a_1, \ldots, a_n$, and let ϕ, $\psi_1, \ldots, \psi_n$, ϕ', $\psi'_1, \ldots, \psi'_n$ and ψ'' be the flows given in Definition 6.4.6 on the previous page, then

1. by definition there is no path in ϕ from an interaction to a cut vertex whose edges are mapped to from instances of one of a_1, …, a_n. By Lemma 6.4.5 on page 62, we know that if there is a path from an interaction to a cut vertex in the flow of $\mathsf{MISR}_n(\Phi, a_1, \ldots, a_n)$ whose edges are mapped to from instances of one of a_1, …, a_n, then there must be a path from an interaction to a cut vertex in ϕ whose edges are mapped to from instances of one of a_1, …, a_n. Hence, the statement follows by contradiction;

2. (a) if the flow of $\mathsf{MISR}_n(\Phi, a_1, \ldots, a_n)$ contains a path from an interaction vertex to a cut vertex whose edges are mapped to from b, then, by Lemma 6.4.5, there is a path from an interaction vertex to a cut vertex in ϕ, so also in ϕ', whose edges are mapped to from b. Hence, the statement follows by contradiction; and

   (b) if there is an interaction (reps., cut) vertex v and a cut (resp., interaction) vertex $\hat{v}$ in the flow of $\mathsf{MISR}_n(\Phi, a_1, \ldots, a_n)$ such that there is a path from v to $\hat{v}$ and a path from v to $\bot$ (resp., $\top$), both of whose edges are mapped to from b, then, by Lemma 6.4.5, there is an interaction (resp., cut) vertex v' and a cut (resp., interaction) vertex $\hat{v}'$ in ϕ such that there is a path from v to $\hat{v}$ and a path from v to $\bot$ (resp., $\top$), both of whose edges are mapped to from b. Furthermore, since we can assume that b is not any of a_1, …, a_n or their duals, ϕ restricted to b equals ϕ' restricted to b. Hence, the statement follows by contradiction.

3. the statement follows by Theorem 6.4.4 on page 61.

□

Remark 6.4.8. Given the atoms, formulae and derivations described in Definition 6.4.1 on page 59, we can prove by induction on k, that, for every $1 \leqslant i \leqslant n$ and every $0 \leqslant k \leqslant N$, the formula $\gamma_{k,i}$ is

- true if at least k of the atoms $a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n$ are true; and
- false if at least $N - k$ of the atoms $a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n$ are false.

It follows by contradiction that $N \geqslant n$. Furthermore, if $N = n$, we know that $\gamma_{k,i}$ is true if and only if at least k of the atoms $a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n$ are true. This makes $\gamma_{k,i}$ a *threshold formula*, as we will see in the next section.

6.4.1 Threshold Formulae

Recently, Jeřábek showed that cut-free SKS proofs can be constructed in quasipolynomial time from SKS proofs with cut [Jeř09]. This is a very surprising result because received wisdom suggests that cut elimination requires exponential-time normalisation, as is the case in Gentzen proof systems. Jeřábek obtained his result by relying on a construction over threshold functions by Atserias, Galesi and Pudlák, in the monotone sequent calculus [AGP02]. We note that the monotone sequent calculus specifies a weaker logic than propositional logic because negation is not freely applicable.

The technique that Jeřábek adopts is indirect because normalisation is performed over proofs in the sequent calculus, which are, in turn, related to deep-inference ones by polynomial simulations, originally studied in [Brü06b].

In [BGGP10], we demonstrated again Jeřábek's result, still by adopting, essentially, the Atserias-Galesi-Pudlák technique, and we improved on that as follows:

1. we significantly simplified the technicalities associated with the use of threshold functions, in particular the formulae and derivations that we adopted were simpler than those in [AGP02];
2. our cut-elimination procedure was direct, *i.e.*, it is internal to system SKS.

In this section I generalise those results in the following two ways:

1. they are extended from cut elimination to streamlining;
2. we observe, in Remark 6.4.8 on the previous page, a criterion on the kind of formulae we need to make the procedure work, which does not necessarily restrict us to threshold formulae.

As Atserias, Galesi and Pudlák argue, there is no apparent reason for this normalisation problem not to be polynomial. The difficulty in obtaining polynomiality resides in finding a suitable class of derivations as described in Remark 6.4.8 on the preceding page.

We present here the main construction of this section, *i.e.*, a class of derivations Γ that adhere to the condition of Definition 6.4.1 on page 59. The complexity of the Γ derivations dominates the complexity of the streamlined proof, and is due to the complexity of certain threshold formulae, on which the Γ derivations are based. The Γ derivations are constructed in Definition 6.4.16 on page 70; this directly leads to Theorem 6.4.17 on page 71, which states a crucial property of the Γ derivations and which is the main result of this section.

Threshold formulae realise boolean threshold functions, which are defined as boolean functions that are true if and only if at least k of n inputs are true (see [Weg87] for a thorough reference on threshold functions).

There are several ways of encoding threshold functions into formulae, and the problem is to find, among them, an encoding that allows us to obtain Theorem 6.4.17 on page 71. Efficiently obtaining the property stated in Theorem 6.4.17 crucially depends also on the proof system we adopt.

In the following, $\underline{n}$ (resp., $\overline{n}$) denotes the maximum (resp., minimum) integer x such that $x \leqslant n/2$ (resp., $x \geqslant n/2$). The reason for this notation will become clear in Definition 6.4.9 on the next page. We will need to split the n atoms $a_1, \ldots, a_n$ into the $\underline{n}$ atoms $a_1, \ldots, a_{\underline{n}}$ and the $\overline{n}$ atoms $a_{\underline{n}+1}, \ldots, a_n$. It is important to notice that, for any n, $\underline{n} + \overline{n} = n$.

The following class of threshold formulae, which we found to work for system SKS, is a simplification of the one adopted in [AGP02].

We now define a class of operators θ^n_k, which takes n atoms as arguments and returns a formula that is true if and only if at least k of the inputs are true.

Definition 6.4.9. For every $n > 0$ and $k \geqslant 0$, we define the operator θ^n_k inductively as follows:

$$\theta^n_k(a_1,\ldots,a_n) = \begin{cases} \mathsf{t} & \text{if } k = 0 \\ \mathsf{f} & \text{if } k > n \\ a_1 & \text{if } n = k = 1 \\ \bigvee_{\substack{i+j=k \\ 0\leqslant i\leqslant \underline{n} \\ 0\leqslant j\leqslant \overline{n}}} \left(\theta^{\underline{n}}_i(a_1,\ldots,a_{\underline{n}}) \wedge \theta^{\overline{n}}_j(a_{\underline{n}+1},\ldots,a_n)\right) & \text{otherwise.} \end{cases}$$

For any n atoms $a_1, \ldots, a_n$, we call $\theta^n_k(a_1,\ldots,a_n)$ the *threshold formula at level k (with respect to $a_1, \ldots, a_n$)*.

See, in Figure 6-1 on the following page, some examples of threshold formulae.

The formulae for threshold functions adopted in [AGP02] correspond, for each choice of k and n, to $\bigvee_{i\geqslant k} \theta^n_i(a_1,\ldots,a_n)$. We presume that [AGP02] employs these more complicated formulae because the formalism adopted there, the sequent calculus, is less flexible than deep inference, requiring more information in threshold formulae in order to construct suitable derivations.

The size of the threshold formulae dominates the cost of the normalisation procedure, so, we evaluate their size. We leave as an exercise the proof of the following proposition.

Proposition 6.4.10. *For any $n > 0$ and $k \geqslant 0$, $\left|\theta^n_k(a_1,\ldots,a_n)\right| \leqslant \left|\theta^n_{\underline{n}+1}(a_1,\ldots,a_n)\right|$.*

Lemma 6.4.11. *The size of $\theta^n_{\underline{n}+1}(a_1,\ldots,a_n)$ is $n^{\mathrm{O}(\log n)}$.*

Proof. Observe that $\left|\theta^n_k(a_1,\ldots,a_n)\right| \leqslant \left|\theta^{n+1}_k(a_1,\ldots,a_{n+1})\right|$. Consider:

$$\begin{aligned} \left|\theta^n_{\underline{n}+1}(a_1,\ldots,a_n)\right| &= \sum_{\substack{i+j=\underline{n}+1 \\ 0\leqslant i\leqslant \underline{n} \\ 0\leqslant j\leqslant \overline{n}}} \left(\left|\theta^{\underline{n}}_i(a_1,\ldots,a_{\underline{n}})\right| + \left|\theta^{\overline{n}}_j(a_{\underline{n}+1},\ldots,a_n)\right|\right) \\ &\leqslant \sum_{\substack{i+j=\underline{n}+1 \\ 0\leqslant i,j\leqslant \overline{n}}} \left(\left|\theta^{\overline{n}}_i(a_1,\ldots,a_{\overline{n}})\right| + \left|\theta^{\overline{n}}_j(a_1,\ldots,a_{\overline{n}})\right|\right) \qquad (6.1) \\ &\leqslant 2(\overline{n}+1)\left|\theta^{\overline{n}}_{\underline{(\overline{n})}+1}(a_1,\ldots,a_{\overline{n}})\right| \quad , \end{aligned}$$

where we use Proposition 6.4.10. Let $h = 2/\log\frac{3}{2}$, then we show that, for any $n > 0$, we have $\left|\theta^n_{\underline{n}+1}(a_1,\ldots,a_n)\right| \leqslant n^{h\log n}$. We reason by induction on n; the case $n = 1$ trivially holds. For $n > 1$, we have that $2(\overline{n}+1) \leqslant n^2$, $\overline{n} \leqslant n$ and $\overline{n} \leqslant \frac{2}{3}n$, so by the inequality (6.1), we have

$$\begin{aligned} \left|\theta^n_{\underline{n}+1}(a_1,\ldots,a_n)\right| &\leqslant 2(\overline{n}+1)\overline{n}^{h\log\overline{n}} \\ &\leqslant n^2 n^{h\log(\frac{2}{3}n)} = n^{h\log n - h\log\frac{3}{2}+2} = n^{h\log n} \quad . \end{aligned}$$

$$
\begin{aligned}
\theta^2_0(a,b) &\equiv \mathsf{t} \quad , \\
\theta^2_1(a,b) &\equiv (\theta^1_1(a) \wedge \theta^1_0(b)) \vee (\theta^1_0(a) \wedge \theta^1_1(b)) \equiv (a \wedge \mathsf{t}) \vee (\mathsf{t} \wedge b) \\
&= a \vee b \quad , \\
\theta^2_2(a,b) &\equiv \theta^1_1(a) \wedge \theta^1_1(b) \\
&\equiv a \wedge b \quad , \\
\theta^3_0(a,b,c) &\equiv \mathsf{t} \quad , \\
\theta^3_1(a,b,c) &\equiv (\theta^1_1(a) \wedge \theta^2_0(b,c)) \vee (\theta^1_0(a) \wedge \theta^2_1(b,c)) \equiv (a \wedge \mathsf{t}) \vee (\mathsf{t} \wedge [(b \wedge \mathsf{t}) \vee (\mathsf{t} \wedge c)]) \\
&= a \vee b \vee c \quad , \\
\theta^3_2(a,b,c) &\equiv (\theta^1_1(a) \wedge \theta^2_1(b,c)) \vee (\theta^1_0(a) \wedge \theta^2_2(b,c)) \\
&= (a \wedge [b \vee c]) \vee (b \wedge c) \quad , \\
\theta^3_3(a,b,c) &\equiv \theta^1_1(a) \wedge \theta^2_2(b,c) \equiv (a \wedge (b \wedge c)) \\
&= a \wedge b \wedge c \quad , \\
\theta^5_0(a,b,c,d,e) &\equiv \mathsf{t} \quad , \\
\theta^5_1(a,b,c,d,e) &\equiv (\theta^2_1(a,b) \wedge \theta^3_0(c,d,e)) \vee (\theta^2_0(a,b) \wedge \theta^3_1(c,d,e)) \\
&= a \vee b \vee c \vee d \vee e \quad , \\
\theta^5_2(a,b,c,d,e) &\equiv (\theta^2_2(a,b) \wedge \theta^3_0(c,d,e)) \vee (\theta^2_1(a,b) \wedge \theta^3_1(c,d,e)) \vee (\theta^2_0(a,b) \wedge \theta^3_2(c,d,e)) \\
&= (a \wedge b) \vee ([a \vee b] \wedge [c \vee d \vee e]) \vee (c \wedge [d \vee e]) \vee (d \wedge e) \quad , \\
\theta^5_3(a,b,c,d,e) &\equiv (\theta^2_2(a,b) \wedge \theta^3_1(c,d,e)) \vee (\theta^2_1(a,b) \wedge \theta^3_2(c,d,e)) \vee (\theta^2_0(a,b) \wedge \theta^3_3(c,d,e)) \\
&= (a \wedge b \wedge [c \vee d \vee e]) \vee ([a \vee b] \wedge [(c \wedge [d \vee e]) \vee (d \wedge e)]) \vee (c \wedge d \wedge e) \quad , \\
\theta^5_4(a,b,c,d,e) &\equiv (\theta^2_2(a,b) \wedge \theta^3_2(c,d,e)) \vee (\theta^2_1(a,b) \wedge \theta^3_3(c,d,e)) \\
&= (a \wedge b \wedge [(c \wedge [d \vee e]) \vee (d \wedge e)]) \vee ([a \vee b] \wedge c \wedge d \wedge e) \quad , \\
\theta^5_5(a,b,c,d,e) &\equiv \theta^2_2(a,b) \wedge \theta^3_3(c,d,e) \\
&= a \wedge b \wedge c \wedge d \wedge e \quad , \\
\theta^5_6(a,b,c,d,e) &\equiv \mathsf{f} \quad .
\end{aligned}
$$

Figure 6-1: Examples of threshold formulae.

□

Theorem 6.4.12. *For any $k \geqslant 0$ the size of $\theta^n_k(a_1,\ldots,a_n)$ is $n^{O(\log n)}$.*

Proof. It immediately follows from Proposition 6.4.10 on page 67 and Lemma 6.4.11 on page 67. □

Remark 6.4.13. Given $n > 1$ and distinct atoms $a_1, \ldots, a_n$. For $0 \leqslant k \leqslant \overline{n}$ and $1 \leqslant l \leqslant \underline{n}$, the following derivation is well defined:

$$\mathsf{w}\uparrow \frac{\theta^n_{\underline{n}}(a_1,\ldots,a_{\underline{n}})\{a_l/\mathsf{f}\} \wedge \theta^{\overline{n}}_k(a_{\underline{n}+1},\ldots,a_n)}{\mathsf{f}} = \mathsf{w}\uparrow \frac{a_1 \wedge \cdots \wedge a_{l-1} \wedge a_{l+1} \wedge \cdots \wedge a_{\underline{n}} \wedge \theta^{\overline{n}}_k(a_{\underline{n}+1},\ldots,a_n)}{\mathsf{t}} \wedge \mathsf{f}\,.$$

Analogously, for $0 \leqslant k \leqslant \underline{n}$ and $\underline{n}+1 \leqslant l \leqslant n$, we can define the following derivation:

$$\mathsf{w}\uparrow \frac{\theta^n_k(a_1,\ldots,a_{\underline{n}}) \wedge \theta^{\overline{n}}_{\overline{n}}(a_{\underline{n}+1},\ldots,a_n)\{a_l/\mathsf{f}\}}{\mathsf{f}} = \mathsf{w}\uparrow \frac{\theta^n_k(a_1,\ldots,a_{\underline{n}}) \wedge a_{\underline{n}+1} \wedge \cdots \wedge a_{l-1} \wedge a_{l+1} \wedge \cdots \wedge a_n}{\mathsf{t}} \wedge \mathsf{f}\,.$$

Both classes of derivations are used in Definition 6.4.14.

The only reason why we require atoms to be distinct is to avoid certain technical problems with substitutions. The same situation occurs in Definitions 6.4.14 and 6.4.16 on the following page.

Definition 6.4.14. Consider $n > 0$, distinct atoms $a_1, \ldots, a_n$.

- For $n > 1$ and $1 \leqslant l \leqslant n$, we define the derivations $\Upsilon^n_{k,l}(a_1,\ldots,a_n)$ and $\Delta^n_{k,l}(a_1,\ldots,a_n)$ as follows:

$$\Upsilon^n_{k,l}(a_1,\ldots,a_n) = \begin{cases} \mathsf{w}\uparrow \dfrac{(\theta^n_{\underline{n}}(a_1,\ldots,a_{\underline{n}}))\{a_l/\mathsf{f}\} \wedge \theta^{\overline{n}}_{k-\underline{n}}(a_{\underline{n}+1},\ldots,a_n)}{\mathsf{f}} & \text{if } \underline{n} \leqslant k \leqslant n \text{ and } l \leqslant \underline{n} \\ \mathsf{w}\uparrow \dfrac{\theta^n_{k-\overline{n}}(a_1,\ldots,a_{\underline{n}}) \wedge (\theta^{\overline{n}}_{\overline{n}}(a_{\underline{n}+1},\ldots,a_n))\{a_l/\mathsf{f}\}}{\mathsf{f}} & \text{if } \overline{n} \leqslant k \leqslant n \text{ and } \underline{n} < l \\ \mathsf{f} & \text{otherwise} \end{cases}$$

  and

$$\Delta^n_{k,l}(a_1,\ldots,a_n) = \begin{cases} \mathsf{w}\downarrow \dfrac{\mathsf{f}}{\theta^{\overline{n}}_k(a_{\underline{n}+1},\ldots,a_n)} & \text{if } 0 < k \leqslant \overline{n} \text{ and } l \leqslant \underline{n} \\ \mathsf{w}\downarrow \dfrac{\mathsf{f}}{\theta^n_k(a_1,\ldots,a_{\underline{n}})} & \text{if } 0 < k \leqslant \underline{n} \text{ and } \underline{n} < l \\ \mathsf{f} & \text{otherwise} \end{cases}.$$

- For $k \geqslant 0$ and $1 \leqslant l \leqslant n$, we define the derivations $\Gamma^n_{k,l}(a_1,\ldots,a_n)$, recursively on n, as follows:

- $\Gamma^1_{0,1}(a_1) = \mathsf{t}$;
- for $k > 0$, $\Gamma^1_{k,1}(a_1) = \mathsf{f}$;
- for $k > n$, $\Gamma^n_{k,l}(a_1,\dots,a_n) = \mathsf{f}$;
- for $n > 1$, $k \leqslant n$ and $l \leqslant \underline{n}$, let $\Gamma^n_{k,l}(a_1,\dots,a_n)$ be

$$\bigvee_{\substack{i+j=k\\ 0\leqslant i<\underline{n}\\ 0\leqslant j\leqslant\overline{n}}} \left(\Gamma^{\underline{n}}_{i,l}(a_1,\dots,a_{\underline{n}}) \wedge \theta^{\overline{n}}_j(a_{\underline{n}+1},\dots,a_n)\right) \vee \Upsilon^n_{k,l}(a_1,\dots,a_n) \vee \Delta^n_{k+1,l}(a_1,\dots,a_n)$$

- for $n > 1$, $k \leqslant n$ and $\underline{n} < l$, let $\Gamma^n_{k,l}(a_1,\dots,a_n)$ be

$$\bigvee_{\substack{i+j=k\\ 0\leqslant i\leqslant\underline{n}\\ 0\leqslant j<\overline{n}}} \left(\theta^{\underline{n}}_i(a_1,\dots,a_{\underline{n}}) \wedge \Gamma^{\overline{n}}_{j,l-\underline{n}}(a_{\underline{n}+1},\dots,a_n)\right) \vee \Upsilon^n_{k,l}(a_1,\dots,a_n) \vee \Delta^n_{k+1,l}(a_1,\dots,a_n)\,.$$

Theorem 6.4.15. *For any $n > 0$, $k \geqslant 0$ and $1 \leqslant l \leqslant n$, the derivation $\Gamma^n_{k,l}(a_1,\dots,a_n)$ has shape*

$$\begin{array}{c} \theta^n_k(a_1,\dots,a_n)\{a_l/\mathsf{f}\} \\ \Vert\{\mathsf{aw}\downarrow,\mathsf{aw}\uparrow\} \\ \theta^n_{k+1}(a_1,\dots,a_n)\{a_l/\mathsf{t}\} \end{array},$$

and $\left|\Gamma^n_{k,l}(a_1,\dots,a_n)\right|$ is $n^{\mathsf{O}(\log n)}$.

Proof. The shape of $\Gamma^n_{k,l}(a_1,\dots,a_n)$ can be verified by inspecting Definition 6.4.14 on the previous page. For example, this is the case when $n > 1$ and $l \leqslant \underline{n} \leqslant k < \overline{n}$:

$$\begin{array}{c} \theta^n_k(a_1,\dots,a_n)\{a_l/\mathsf{f}\} \\ \Gamma^n_{k,l}(a_1,\dots,a_n)\Vert \\ \theta^n_{k+1}(a_1,\dots,a_n)\{a_l/\mathsf{t}\} \end{array} = \bigvee_{\substack{i+j=k\\ 0\leqslant i<p\\ 0\leqslant j\leqslant q}} \left(\begin{array}{c} \theta^p_i(a_1,\dots,a_p)\{a_l/\mathsf{f}\} \\ \Gamma^p_{i,l}(a_1,\dots,a_p)\Vert \\ \theta^p_{i+1}(a_1,\dots,a_p)\{a_l/\mathsf{t}\} \end{array} \wedge\ \theta^q_j(a_{p+1},\dots,a_n)\right) \vee$$

$$\vee\, \mathsf{w}\uparrow \frac{(\theta^p_p(a_1,\dots,a_p))\{a_l/\mathsf{f}\} \wedge \theta^q_{k-p}(a_{p+1},\dots,a_n)}{\mathsf{f}} \vee\, \mathsf{w}\downarrow \frac{\mathsf{f}}{\theta^q_{k+1}(a_{p+1},\dots,a_n)}\quad.$$

General (co)weakening rule instances can be replaced by their atomic counterparts due to Lemma 2.3.11 on page 17. The size bound on $\Gamma^n_{k,l}(a_1,\dots,a_n)$ follows from Proposition 4.1.11 on page 33 and Theorem 6.4.12 on the preceding page. □

Definition 6.4.16. Consider $n > 0$, distinct atoms $a_1, \ldots, a_n$. For $k \geqslant 0$, we define the derivation $\Gamma^n_k(a_1,\ldots,a_n)$ to be:

$$\left[\begin{array}{c} \left(\begin{array}{ccc} & \theta^n_k(a_1,\ldots,a_n)\{a_1/\mathsf{f}\} & \\ a_1 \;\wedge & \Vdots\,\Gamma^n_{k,1}(a_1,\ldots,a_n) & \\ & \theta^n_{k+1}(a_1,\ldots,a_n)\{a_1/\mathsf{t}\} & \end{array}\right) \vee \cdots \vee \left(\begin{array}{ccc} & \theta^n_k(a_1,\ldots,a_n)\{a_n/\mathsf{f}\} & \\ a_n \;\wedge & \Vdots\,\Gamma^n_{k,n}(a_1,\ldots,a_n) & \\ & \theta^n_{k+1}(a_1,\ldots,a_n)\{a_n/\mathsf{t}\} & \end{array}\right) \\ \Vdots\,\{\mathsf{ac}\uparrow,\mathsf{s}\} \\ \theta^n_{k+1}(a_1,\ldots,a_n) \end{array}\right] \\ \Vdots\,\{\mathsf{c}\downarrow\} \\ \theta^n_{k+1}(a_1,\ldots,a_n) \\ \Vdots\,\{\mathsf{c}\uparrow\} \\ \left(\begin{array}{ccc} \theta^n_{k+1}(a_1,\ldots,a_n) & & \theta^n_{k+1}(a_1,\ldots,a_n) \\ \Vdots\,\{\mathsf{ac}\downarrow,\mathsf{s}\} & \wedge \cdots \wedge & \Vdots\,\{\mathsf{ac}\downarrow,\mathsf{s}\} \\ \left[a_1 \vee \theta^n_{k+1}(a_1,\ldots,a_n)\{a_1/\mathsf{f}\}\right] & & \left[a_n \vee \theta^n_{k+1}(a_1,\ldots,a_n)\{a_n/\mathsf{f}\}\right] \end{array}\right)$$

where we use the derivations constructed in the proof of Lemma 2.3.10 on page 16.

Theorem 6.4.17. *For any $n > 0$ and $k \geqslant 0$, the derivation $\Gamma^n_k(a_1,\ldots,a_n)$ has shape*

$$\begin{array}{c} \left(a_1 \wedge \theta^n_k(a_1,\ldots,a_n)\{a_1/\mathsf{f}\}\right) \vee \cdots \vee \left(a_n \wedge \theta^n_k(a_1,\ldots,a_n)\{a_n/\mathsf{f}\}\right) \\ \Vdots\,\mathsf{SKS}\setminus\{\mathsf{ai}\downarrow,\mathsf{ai}\uparrow\} \\ \left[a_1 \vee \theta^n_{k+1}(a_1,\ldots,a_n)\{a_1/\mathsf{f}\}\right] \wedge \cdots \wedge \left[a_n \vee \theta^n_{k+1}(a_1,\ldots,a_n)\{a_n/\mathsf{f}\}\right] \end{array},$$

and $\left|\Gamma^n_k(a_1,\ldots,a_n)\right|$ *is* $n^{\mathsf{O}(\log n)}$.

Definition 6.4.18. For every $n > 0$, we define

- the reduction $\rightarrow_{\mathsf{qmis}_n}$ (where qmis stands for *quasipolynomial multiple isolated subflows*); and
- and the operator the *Quasipolynomial Multiple Isolated Subflows Remover*, QMISR_n,

to be special cases of $\rightarrow_{\mathsf{mis}_n}$ and MISR_n, respectively, such that, given atoms $(a_1,\ldots,a_n)$,

- $N = n$;
- for $0 \leqslant k \leqslant n$ and $1 \leqslant i \leqslant n$, $\gamma_{k,i} = (\theta^n_k(a_1,\ldots,a_n))\{a_i/\mathsf{f}\}$; and
- for $1 \leqslant k \leqslant n$, $\Gamma_k = \Gamma^n_k(a_1,\ldots,a_n)$.

Theorem 6.4.19. *For every $n > 0$, $\rightarrow_{\mathsf{qmis}_n}$ is sound; moreover, if $\Phi \rightarrow_{\mathsf{qmis}_n} \Psi$, then the size of Ψ depends polynomially on the size of Φ and quasipolynomially on n.*

Proof. The result follows by Theorem 6.4.4 on page 61, Definition 6.4.9 on page 67 and Theorem 6.4.17. □

Proposition 6.4.20. *Given atoms $a_1, \ldots, a_n$ and a derivation Φ that is in simple form with respect to $a_1, \ldots, a_n$,*

1. $\mathsf{QMISR}_n(\Phi, a_1, \ldots, a_n)$ *is weakly streamlined with respect to $a_1, \ldots, a_n$;*
2. *for any atom b,*
   (a) *if Φ is weakly streamlined with respect to b, then $\mathsf{QMISR}_n(\Phi, a_1, \ldots, a_n)$ is weakly streamlined with respect to b, and*
   (b) *if b is not the dual of any of $a_1, \ldots, a_n$ and Φ is in simple form with respect to b, then $\mathsf{QMISR}_n(\Phi, a_1, \ldots, a_n)$ is in simple form with respect to b; and*
3. *the size of $\mathsf{QMISR}_n(\Phi, a_1, \ldots, a_n)$ depends polynomially on the size of Φ, and quasipolynomially on n.*

Proof. The statements follow by Proposition 6.4.7 on page 64 and Theorem 6.4.17 on the preceding page. □

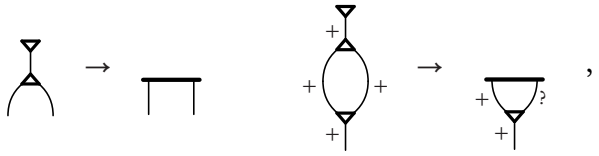
Chapter 7

Local Reductions

In this chapter, we see local transformations, which are based on reduction rules. It is convenient to classify reduction rules into those for weakening and those for contraction. After seeing flow reductions and tying them with derivations, in Section 7.1 on page 75, we explore some of their basic properties, in the two short Sections 7.2 on page 77 and 7.3 on page 78.

Definition 7.0.1. In Figure 7-1 on the following page, we define graphical expressions of the kind $r : \phi' \to \psi'$, where r is a name and ϕ' and ψ' are flows.

Example 7.0.2. The 'reduction' on the left, when used inside a larger flow, might create a situation as on the right:



where the graph at the right is not an atomic flow, for lack of a polarity assignment.

This prompts us to define reduction rules for atomic flows as follows.

Definition 7.0.3. An (*atomic-flow*) *reduction rule* r *from flow* ϕ' *to flow* ψ' is a quadruple (ϕ', ψ', f, g) such that:

1. f is a one-to-one map from the upper edges of ϕ' to the upper edges of ψ',
2. g is a one-to-one map from the lower edges of ϕ' to the lower edges of ψ',
3. for every polarity assignment π for ϕ', there is a polarity assignment π' for ψ' such that $\pi'(f(\epsilon)) = \pi(\epsilon)$ and $\pi'(g(\epsilon')) = \pi(\epsilon')$, for any upper edge ϵ and any lower edge ϵ' of ϕ';

we define reduction rules with graphical expressions $r : \phi' \to \psi'$, where f and g are indicated by labelling edges. For every reduction rule $r : \phi' \to \psi'$, the reduction $\to_r$ is defined, such

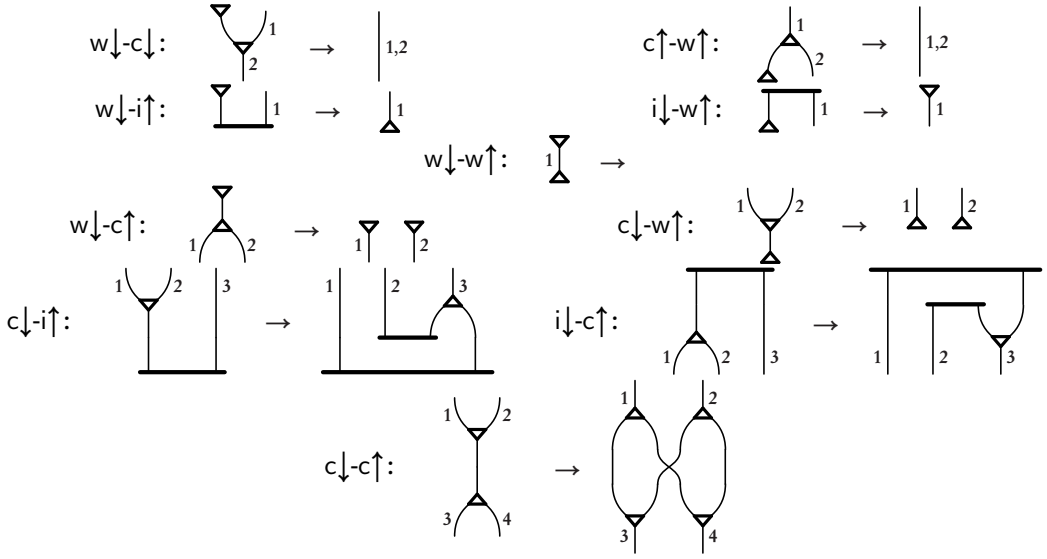
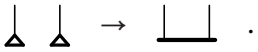


Figure 7-1: Atomic-flow reduction rules.

that $\phi \to_r \psi$ if and only if ϕ' appears as a subflow in ϕ and we obtain ψ by replacing ϕ' with ψ' in ϕ, while respecting the correspondence of edges; we call this operation a *reduction by* r.

Remark 7.0.4. The condition on polarity assignments for a reduction rule r guarantees that the ψ in $\phi \to_r \psi$ is a proper atomic flow, if ϕ is one.

Remark 7.0.5. Because of the condition on polarity assignments for reduction rules, two distinct connected components in a flow cannot be connected by a reduction. To see that this is impossible, consider the following 'reduction rule', which violates the condition on polarity assignments:



For this 'reduction rule' there exist both valid (left) and invalid (right) polarity assignments:

It is immediate to check:

Proposition 7.0.6. *The graphical expressions in Figure 7-1 are atomic-flow reduction rules.*

Definition 7.0.7. A finite set of reduction rules is a *flow rewriting system*. For every flow rewriting system $F = \{r_1, \ldots, r_h\}$ we define $\to_F = \to_{r_1} \cup \cdots \cup \to_{r_h}$. The reflexive transitive closure of $\to_F$ is denoted by $\to_F^\star$. Given a set of atomic flows S, we say that a flow rewriting system F is *terminating on* S if there is no infinite chain $\phi_1 \to_F \phi_2 \to_F \cdots$, for every $\phi_1 \in S$; if F is terminating on the set of atomic flows, we say that it is *terminating*. We say that the flow ϕ is *normal* for flow rewriting system F if there is no flow ψ such that $\phi \to_F \psi$.

Definition 7.0.8. The following flow rewriting system is called w:

$$\{\ \mathsf{w}{\downarrow}\text{-}\mathsf{c}{\downarrow}\ ,\ \mathsf{c}{\uparrow}\text{-}\mathsf{w}{\uparrow}\ ,\ \mathsf{w}{\downarrow}\text{-}\mathsf{i}{\uparrow}\ ,\ \mathsf{i}{\downarrow}\text{-}\mathsf{w}{\uparrow}\ ,\ \mathsf{w}{\downarrow}\text{-}\mathsf{w}{\uparrow}\ ,\ \mathsf{w}{\downarrow}\text{-}\mathsf{c}{\uparrow}\ ,\ \mathsf{c}{\downarrow}\text{-}\mathsf{w}{\uparrow}\ \}\quad .$$

Definition 7.0.9. The following flow rewriting system is called c:

$$\{\ \mathsf{c}{\downarrow}\text{-}\mathsf{i}{\uparrow}\ ,\ \mathsf{i}{\downarrow}\text{-}\mathsf{c}{\uparrow}\ ,\ \mathsf{c}{\downarrow}\text{-}\mathsf{c}{\uparrow}\ \}\quad .$$

Maximal ai-paths provide for a measure when dealing with the termination of c.

Remark 7.0.10. A simple inspection to the reduction rules of c convinces us that reducing by c does not change the number and length of the maximal ai-paths of a flow. The same holds for the maximal ai-paths to or from vertices that are not involved in a given reduction.

We now state two propositions whose proofs are immediate from the appropriate definitions:

Proposition 7.0.11. *Given a weakly-streamlined flow ϕ, if $\phi \to_{\mathsf{w}}^{\star} \psi$ and ψ is normal for* w*, then ψ is super streamlined.*

Proposition 7.0.12. *Given a super-streamlined flow ϕ, if $\phi \to_{\mathsf{c}}^{\star} \psi$ and ψ is normal for* c*, then ψ is hyper streamlined.*

7.1 Soundness

Definition 7.1.1. A reduction rule r is *sound* if $\to_r$ is sound.

The proof of the following theorem is essentially contained in Figure 7-2 on the following page and Figure 7-3 on page 77.

Theorem 7.1.2. *The reduction rules* w↓-c↓, w↓-i↑, w↓-w↑, w↓-c↑, c↓-i↑, c↓-c↑, c↑-w↑, i↓-w↑, c↓-w↑ *and* i↓-c↑ *are sound.*

Proof. For $r \in \{\mathsf{w}{\downarrow}\text{-}\mathsf{c}{\downarrow}, \mathsf{w}{\downarrow}\text{-}\mathsf{i}{\uparrow}, \mathsf{w}{\downarrow}\text{-}\mathsf{w}{\uparrow}, \mathsf{w}{\downarrow}\text{-}\mathsf{c}{\uparrow}, \mathsf{c}{\downarrow}\text{-}\mathsf{i}{\uparrow}, \mathsf{c}{\downarrow}\text{-}\mathsf{c}{\uparrow}\}$ and $r : \phi' \to \psi'$ as in the left columns of Figures 7-2 on the following page and 7-3 on page 77, for every ϕ and ψ such that $\phi \to_r \psi$ and for every Φ with flow ϕ, the right columns of the tables provide reductions $\Phi \to_r \Psi$, where Ψ has flow ψ, as follows. If $\Phi' \to_r \Psi'$ is the reduction provided by the table, then

$$\Phi = \begin{array}{r} \alpha \\ {\scriptstyle \Psi_1}\| \\ \alpha' \\ \Phi' \| \\ \beta' \\ {\scriptstyle \Psi_2}\| \\ \beta \end{array} \qquad \text{and} \qquad \Psi = \begin{array}{r} \alpha \\ {\scriptstyle \Psi_1}\| \\ \alpha' \\ \Psi' \| \\ \beta' \\ {\scriptstyle \Psi_2}\| \\ \beta \end{array} \quad .$$

We can deal with the remaining rules by employing dual derivations to the ones shown. □

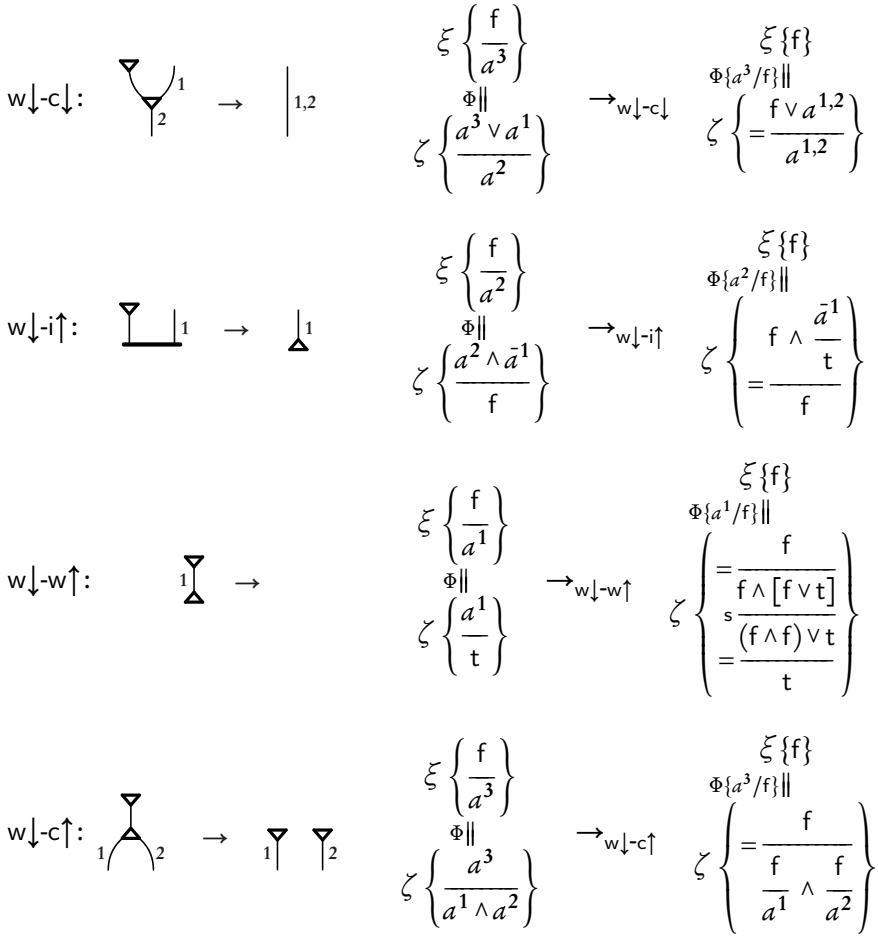


Figure 7-2: 'Downwards' reduction rules for weakening and their soundness.

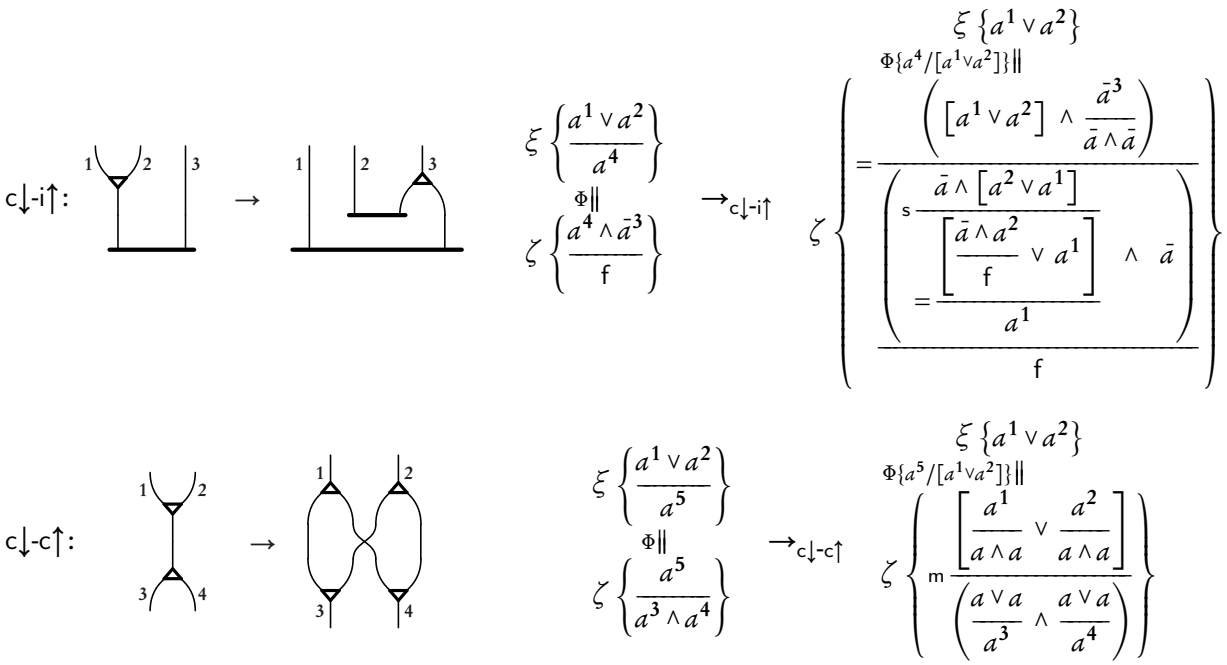


Figure 7-3: 'Downwards' reduction rules for contraction and their soundness.

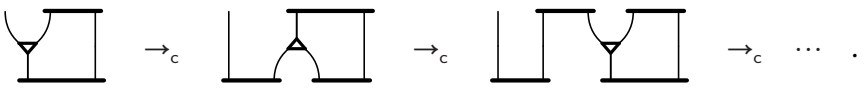
Remark 7.1.3. The previous soundness theorem only depends on the switch and medial rules for the reductions in Figure 7-3. Any system obtained from SKS by replacing s and m with linear rules that can derive them would support a soundness theorem like the one above, for the same reduction rules. For example, we could think of replacing s with the rule $\mathsf{s}'\dfrac{[\alpha \vee \beta] \wedge [\gamma \vee \delta]}{(\alpha \wedge \gamma) \vee [\beta \vee \delta]}$, from which s is derivable.

7.2 Termination and Confluence

Theorem 7.2.1. *Flow rewriting system* w *is terminating.*

Proof. At every reduction, the number of edges decreases. □

Remark 7.2.2. Flow rewriting system c is not terminating:



We see that if a contraction vertex belongs to an ai-cycle, reductions by c make it 'bounce' in the ai-cycle and create a trail; while bouncing, the vertex alternates between contraction and cocontraction.

Theorem 7.2.3. *Flow rewriting system* c *is terminating on the set of cycle-free flows.*

Proof. Let ϕ be a cycle-free flow. We associate to each contraction (resp., cocontraction) vertex v its *rank* $\mathsf{r}_v = \sum_{p_i \in I_v} h_i$, where I_v is the set of all maximal ai-paths $p_i = \epsilon^i_1, \ldots, \epsilon^i_{h_i}$ from v, such that ϵ^i_1 is the lower (resp., upper) edge of v (so, the rank of a vertex is the sum of the lengths of certain maximal ai-paths from it). Note that every (co)contraction vertex has non-zero rank. We prove that a reduction of ϕ by c decreases the sum of the ranks of the (co)contraction vertices of ϕ. First note that the rank of the vertices not involved in the reduction step stays the same (see Remark 7.0.10 on page 75). We then need to show that the sum of the ranks decreases for the vertices involved. There are three cases, depending on the reduction rule:

c↓-i↑: a contraction vertex v is replaced by a cocontraction vertex v', and $\mathsf{r}_{v'} = \mathsf{r}_v - n$, where $n > 0$ is the number of maximal ai-paths from v whose first edge is the lower edge of v;

i↓-c↑: this is dual to the previous case;

c↓-c↑: a contraction vertex v and a cocontraction vertex v' are replaced by two contraction vertices v_1 and v_2 and two cocontraction vertices v'_1 and v'_2; we have $\mathsf{r}_{v_1} + \mathsf{r}_{v_2} = \mathsf{r}_v - n$, where $n > 0$ is the number of maximal ai-paths from v whose first edge is the lower edge of v; analogously, we have $\mathsf{r}_{v'_1} + \mathsf{r}_{v'_2} = \mathsf{r}_{v'} - n'$, where $n' > 0$ is the number of maximal ai-paths from v' whose first edge is the upper edge of v'.

□

Conjecture 7.2.4. Flow rewriting system w ∪ c is 'confluent'.

Remark 7.2.5. It seems straightforward to verify the statement by checking each critical pair of w ∪ c. However, as pointed out by François Lamarche, this is not enough. It is not immediate how the edges and vertices added to an atomic flow by a reduction should be named. We know that if the flows ψ and ψ' are both normal forms of the flow ϕ, with respect to w∪c, then there exists an isomorphism between them. However, due to non-trivial automorphisms in ψ and ψ' this isomorphism might not be unique, and we lack a way to construct an isomorphism in a uniform way. This problem will be the focus of future research.

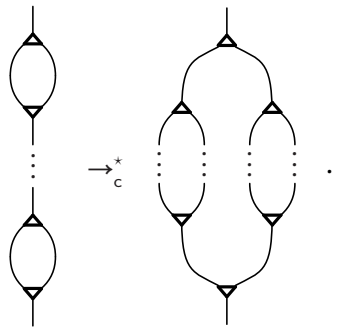
Remark 7.2.6. Notice that, by Remark 4.1.3 on page 29, the mapping from a redex in a flow to the corresponding inference rule instances in a derivation might not be unique. This means that a chain of reductions of flows does not uniquely determine the order of the reductions of derivations. However, once we have a notion of confluence of the local reductions, it will follow that the normal form of derivation is unique.

7.3 Complexity

Proposition 7.3.1. *Given a derivation* Φ*, there exists a derivation* Ψ*, such that* $\Phi \to^\star_{\mathsf{w}} \Psi$*,* Ψ *is normal for* w *and the size of* Ψ *depends at most linearly on the size of* Φ*.*

Proof. The number of reductions used to arrive at Ψ is bound by the number of edges in the flow of Φ, so by the size of Φ. Furthermore, each reductions shown in Figure 7-2 on page 76 grows the derivation by at most a constant. Hence, the size of Ψ depends at most linearly on the size of Φ. □

Remark 7.3.2. Normalising by c can blow the size of flows exponentially, in particular in a situation like the following (noted by Lutz Straßburger):



In fact, if there are n couples of cocontraction/contraction vertices like the two shown above on the left, then there are 2^n maximal ai-paths, and their number (and length) is conserved by $\rightarrow_c^\star$ (see Remark 7.0.10 on page 75). Exactly one ai-path passes through each edge in the middle portion of the flow on the right. It follows that normalising derivations by c can also blow their size exponentially.

Chapter 8

Main Result

We now present the main result of this thesis: Three procedures for obtaining weakly streamlined derivations. Corollaries of the main results are: cut elimination, super-streamlining and hyper-streamlining.

Theorem 8.0.1. *Given a derivation Φ and distinct and pairwise non-dual atoms $a_1, \ldots, a_n$, such that $a_1, \ldots, a_n$ and their duals are all the atoms appearing in Φ,*

1. *let*
$$\Phi' = \mathsf{ISR}(\ldots \mathsf{ISR}(\mathsf{Si}(\Phi, a_1, \ldots, a_n), a_1), \ldots, a_n) \quad \textit{and}$$
$$\Phi'' = \mathsf{ISR}(\ldots \mathsf{ISR}(\mathsf{Si}(\Phi', \bar{a}_1, \ldots, \bar{a}_n), \bar{a}_1), \ldots, \bar{a}_n)\,,$$
then
   (a) *Φ'' is weakly streamlined, and*
   (b) *the size of Φ'' depends at most exponentially on the size of Φ;*
2. *let $\Phi' = \mathsf{PB}(\ldots \mathsf{PB}(\Phi, a_1), \ldots, a_n)$, then*
   (a) *Φ' is weakly streamlined, and*
   (b) *the size of Φ' depends at most exponentially on the size of Φ; and*
3. *let*
$$\Phi' = \mathsf{QMISR}(\mathsf{Si}(\Phi, a_1, \ldots, a_n), a_1, \ldots, a_n) \quad \textit{and}$$
$$\Phi'' = \mathsf{QMISR}(\mathsf{Si}(\Phi', \bar{a}_1, \ldots, \bar{a}_n), \bar{a}_1, \ldots, \bar{a}_n)\,,$$
then
   (a) *Φ'' is weakly streamlined, and*
   (b) *the size of Φ'' depends at most quasipolynomially on the size of Φ.*

Proof. The statements follow by Proposition 6.1.5 on page 47, Proposition 6.2.6 on page 51, Proposition 6.3.5 on page 56 and Proposition 6.4.20 on page 72. □

Corollary 8.0.2. *Given a derivation (resp., proof)* Φ*, there exists a super-streamlined derivation (resp., cut-free proof)* Ψ *with the same premiss and conclusion as* Φ*, such that the size of* Ψ *depends at most quasipolynomially on the size of* Φ*.*

Proof. The result follows by Theorem 8.0.1 on the preceding page, Proposition 7.0.11 on page 75, Theorem 7.1.2 on page 75 and Proposition 7.3.1 on page 78. □

Corollary 8.0.3. *Given a derivation* Φ*, there exists a hyper-streamlined derivation* Ψ *with the same premiss and conclusion as* Φ*.*

Proof. The result follows by Theorem 8.0.1 on the preceding page, Proposition 7.0.11 on page 75, Proposition 7.0.12 on page 75, Theorem 7.1.2 on page 75 and Proposition 7.3.1 on page 78. □

Index

Bibliography

[AGP02] Albert Atserias, Nicola Galesi, and Pavel Pudlák. Monotone simulations of non-monotone proofs. *Journal of Computer and System Sciences*, 65(4):626–638, 2002.

[BG09] Paola Bruscoli and Alessio Guglielmi. On the proof complexity of deep inference. *ACM Transactions on Computational Logic*, 10(2):1–34, 2009. Article 14. `http://cs.bath.ac.uk/ag/p/PrComplDI.pdf`.

[BGGP10] Paola Bruscoli, Alessio Guglielmi, Tom Gundersen, and Michel Parigot. A quasipolynomial cut-elimination procedure in deep inference via atomic flows and threshold formulae. Accepted by LPAR-16. `http://cs.bath.ac.uk/ag/p/QPNDI.pdf`, 2010.

[BL05] Kai Brünnler and Stéphane Lengrand. On two forms of bureaucracy in derivations. In Paola Bruscoli, François Lamarche, and Charles Stewart, editors, *Structures and Deduction*, pages 69–80. Technische Universität Dresden, 2005. ICALP Workshop. ISSN 1430-211X. `http://www.iam.unibe.ch/~kai/Papers/sd05.pdf`.

[Bru02] Paola Bruscoli. A purely logical account of sequentiality in proof search. In Peter J. Stuckey, editor, *Logic Programming, 18th International Conference*, volume 2401 of *Lecture Notes in Computer Science*, pages 302–316. Springer-Verlag, 2002. `http://cs.bath.ac.uk/pb/bvl/bvl.pdf`.

[Brü03a] Kai Brünnler. Atomic cut elimination for classical logic. In M. Baaz and J. A. Makowsky, editors, *CSL 2003*, volume 2803 of *Lecture Notes in Computer Science*, pages 86–97. Springer-Verlag, 2003. `http://www.iam.unibe.ch/~kai/Papers/ace.pdf`.

[Brü03b] Kai Brünnler. Two restrictions on contraction. *Logic Journal of the IGPL*, 11(5):525–529, 2003. `http://www.iam.unibe.ch/~kai/Papers/RestContr.pdf`.

[Brü04] Kai Brünnler. *Deep Inference and Symmetry in Classical Proofs*. Logos Verlag, Berlin, 2004. `http://www.iam.unibe.ch/~kai/Papers/phd.pdf`.

[Brü06a] Kai Brünnler. Cut elimination inside a deep inference system for classical predicate logic. *Studia Logica*, 82(1):51–71, 2006. `http://www.iam.unibe.ch/~kai/Papers/q.pdf`.

[Brü06b] Kai Brünnler. Deep inference and its normal form of derivations. In Arnold Beckmann, Ulrich Berger, Benedikt Löwe, and John V. Tucker, editors, *Computability in Europe 2006*, volume 3988 of *Lecture Notes in Computer Science*, pages 65–74. Springer-Verlag, July 2006. `http://www.iam.unibe.ch/~kai/Papers/n.pdf`.

[Brü06c] Kai Brünnler. Deep sequent systems for modal logic. In Guido Governatori, Ian Hodkinson, and Yde Venema, editors, *Advances in Modal Logic*, volume 6, pages 107–119. College Publications, 2006. `http://www.aiml.net/volumes/volume6/Bruennler.ps`.

[Brü06d] Kai Brünnler. Locality for classical logic. *Notre Dame Journal of Formal Logic*, 47(4):557–580, 2006. `http://www.iam.unibe.ch/~kai/Papers/LocalityClassical.pdf`.

[BT01] Kai Brünnler and Alwen Fernanto Tiu. A local system for classical logic. In R. Nieuwenhuis and A. Voronkov, editors, *LPAR 2001*, volume 2250 of *Lecture Notes in Computer Science*, pages 347–361. Springer-Verlag, 2001. http://www.iam.unibe.ch/~kai/Papers/lcl-lpar.pdf.

[Bus91] Samuel R. Buss. The undecidability of k-provability. *Annals of Pure and Applied Logic*, 53(1):75–102, 1991.

[Car97] Alessandra Carbone. Interpolants, cut elimination and flow graphs for the propositional calculus. *Annals of Pure and Applied Logic*, 83:249–299, 1997.

[DG04] Pietro Di Gianantonio. Structures for multiplicative cyclic linear logic: Deepness vs cyclicity. In J. Marcinkowski and A. Tarlecki, editors, *CSL 2004*, volume 3210 of *Lecture Notes in Computer Science*, pages 130–144. Springer-Verlag, 2004. http://www.dimi.uniud.it/~pietro/papers/Soft-copy-ps/scll.ps.gz.

[Gen69] Gerhard Gentzen. Investigations into logical deduction. In M. E. Szabo, editor, *The Collected Papers of Gerhard Gentzen*, pages 68–131. North-Holland, Amsterdam, 1969.

[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. http://www.lmcs-online.org/ojs/viewarticle.php?id=341.

[GGP10] Alessio Guglielmi, Tom Gundersen, and Michel Parigot. A proof calculus which reduces syntactic bureaucracy. Accepted by RTA 2010., 2010.

[GGS10] Alessio Guglielmi, Tom Gundersen, and Lutz Straßburger. Breaking paths in atomic flows for classical logic. Accepted by LICS 2010. http://cs.bath.ac.uk/ag/p/BPAF.pdf, 2010.

[GS01] Alessio Guglielmi and Lutz Straßburger. Non-commutativity and MELL in the calculus of structures. In L. Fribourg, editor, *CSL 2001*, volume 2142 of *Lecture Notes in Computer Science*, pages 54–68. Springer-Verlag, September 2001. http://cs.bath.ac.uk/ag/p/NoncMELLCoS.pdf.

[GS02] Alessio Guglielmi and Lutz Straßburger. A non-commutative extension of MELL. In M. Baaz and A. Voronkov, editors, *LPAR 2002*, volume 2514 of *Lecture Notes in Computer Science*, pages 231–246. Springer-Verlag, October 2002. http://www.lix.polytechnique.fr/~lutz/papers/NEL.pdf.

[GS09] Alessio Guglielmi and Lutz Straßburger. A system of interaction and structure V: The exponentials and splitting. Submitted. http://www.lix.polytechnique.fr/~lutz/papers/NEL-splitting.pdf, 2009.

[GT07] Rajeev Goré and Alwen Tiu. Classical modal display logic in the calculus of structures and minimal cut-free deep inference calculi for S5. *Journal of Logic and Computation*, 17(4):767–794, 2007. http://users.rsise.anu.edu.au/~tiu/papers/cmdl.pdf.

[Gug04] Alessio Guglielmi. Formalism A. http://cs.bath.ac.uk/ag/p/AG11.pdf, 2004.

[Gug07] Alessio Guglielmi. A system of interaction and structure. *ACM Transactions on Computational Logic*, 8(1):1–64, 2007. http://cs.bath.ac.uk/ag/p/SystIntStr.pdf.

[Gui06] Yves Guiraud. The three dimensions of proofs. *Annals of Pure and Applied Logic*, 141(1-2):266–295, 2006. http://www.loria.fr/~guiraudy/recherche/cos1.pdf.

[Jeř09] Emil Jeřábek. Proof complexity of the cut-free calculus of structures. *Journal of Logic and Computation*, 19(2):323–339, 2009. http://www.math.cas.cz/~jerabek/papers/cos.pdf.

[Kah06] Ozan Kahramanoğulları. Reducing nondeterminism in the calculus of structures. In M. Hermann and A. Voronkov, editors, *LPAR 2006*, volume 4246 of *Lecture Notes in Computer Science*, pages 272–286. Springer-Verlag, 2006. http://dx.doi.org/10.1007/11916277_19.

[Kah07] Ozan Kahramanoğulları. System BV is NP-complete. *Annals of Pure and Applied Logic*, 152(1–3):107–121, 2007. http://dx.doi.org/10.1016/j.apal.2007.11.005.

[Kah08] Ozan Kahramanoğulları. Maude as a platform for designing and implementing deep inference systems. In J. Visser and V. Winter, editors, *Proceedings of the Eighth International Workshop on Rule Based Programming (RULE 2007)*, volume 219 of *Electronic Notes in Theoretical Computer Science*, pages 35–50. Elsevier, 2008. http://www.doc.ic.ac.uk/~ozank/Papers/rule07.pdf.

[LS05a] François Lamarche and Lutz Straßburger. Constructing free boolean categories. In Prakash Panangaden, editor, *20th Annual IEEE Symposium on Logic in Computer Science*, pages 209–218. IEEE, 2005. http://www.lix.polytechnique.fr/~lutz/papers/FreeBool-long.pdf.

[LS05b] François Lamarche and Lutz Straßburger. Naming proofs in classical propositional logic. In Paweł Urzyczyn, editor, *Typed Lambda Calculi and Applications*, volume 3461 of *Lecture Notes in Computer Science*, pages 246–261. Springer-Verlag, 2005. http://www.lix.polytechnique.fr/~lutz/papers/namingproofsCL.pdf.

[LS06] François Lamarche and Lutz Straßburger. From proof nets to the free *-autonomous category. *Logical Methods in Computer Science*, 2(4):3:1–44, 2006. http://arxiv.org/pdf/cs.LO/0605054.

[SL04] Lutz Straßburger and François Lamarche. On proof nets for multiplicative linear logic with units. In J. Marcinkowski and A. Tarlecki, editors, *CSL 2004*, volume 3210 of *Lecture Notes in Computer Science*, pages 145–159. Springer-Verlag, 2004. http://www.lix.polytechnique.fr/~lutz/papers/multPN.pdf.

[Sta78] Richard Statman. Bounds for proof-search and speed-up in the predicate calculus. *Annals of Mathematical Logic*, 15:225–287, 1978.

[Sto07] Phiniki Stouppa. A deep inference system for the modal logic S5. *Studia Logica*, 85(2):199–214, 2007. http://www.iam.unibe.ch/til/publications/pubitems/pdfs/sto07.pdf.

[Str02] Lutz Straßburger. A local system for linear logic. In M. Baaz and A. Voronkov, editors, *LPAR 2002*, volume 2514 of *Lecture Notes in Computer Science*, pages 388–402. Springer-Verlag, 2002. http://www.lix.polytechnique.fr/~lutz/papers/lls-lpar.pdf.

[Str03a] Lutz Straßburger. *Linear Logic and Noncommutativity in the Calculus of Structures*. PhD thesis, Technische Universität Dresden, 2003. http://www.lix.polytechnique.fr/~lutz/papers/dissvonlutz.pdf.

[Str03b] Lutz Straßburger. MELL in the calculus of structures. *Theoretical Computer Science*, 309:213–285, 2003. http://www.lix.polytechnique.fr/~lutz/papers/els.pdf.

[Str05] Lutz Straßburger. From deep inference to proof nets. In Paola Bruscoli, François Lamarche, and Charles Stewart, editors, *Structures and Deduction*, pages 2–18. Technische Universität Dresden, 2005. ICALP Workshop. ISSN 1430-211X. http://www.lix.polytechnique.fr/~lutz/papers/deepnet-SD05.pdf.

[Str09] Lutz Straßburger. From deep inference to proof nets via cut elimination. *Journal of Logic and Computation*, 2009. In press. http://www.lix.polytechnique.fr/~lutz/papers/deepnet.pdf.

[Tiu06a] Alwen Tiu. A local system for intuitionistic logic. In M. Hermann and A. Voronkov, editors, *LPAR 2006*, volume 4246 of *Lecture Notes in Computer Science*, pages 242–256. Springer-Verlag, 2006. http://users.rsise.anu.edu.au/~tiu/localint.pdf.

[Tiu06b] Alwen Tiu. A system of interaction and structure II: The need for deep inference. *Logical Methods in Computer Science*, 2(2:4):1–24, 2006. http://arxiv.org/pdf/cs.LO/0512036.

[Weg87] Ingo Wegener. *The Complexity of Boolean Functions*. John Wiley & Sons Ltd and B. G. Teubner, Stuttgart, 1987.