



**HAL**  
open science

# A General View of Normalisation through Atomic Flows

Tom Gundersen

► **To cite this version:**

Tom Gundersen. A General View of Normalisation through Atomic Flows. Mathematics [math]. The University of Bath, 2009. English. NNT: . tel-00441540v1

**HAL Id: tel-00441540**

**<https://theses.hal.science/tel-00441540v1>**

Submitted on 16 Dec 2009 (v1), last revised 11 Aug 2010 (v3)

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# 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

**COPYRIGHT**

Attention is drawn to the fact that copyright of this thesis rests with its author. A copy of this thesis has been supplied on condition that anyone who consults it is understood to recognise that its copyright rests with the author and they must not copy it or use material from it except as permitted by law or with the consent of the author.

This thesis may be made available for consultation within the University Library and may be photocopied or lent to other libraries for the purposes of consultation.



Signature of Author

Tom Erik Gundersen

# A General View of Normalisation through Atomic Flows

Tom Erik Gundersen

DRAFT

## 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.

DRAFT

## ACKNOWLEDGEMENTS

TODO

DRAFT

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>I</b>	<b>Derivations</b>	<b>6</b>
<b>2</b>	<b>Propositional Classical Logic</b>	<b>7</b>
2.1	The Functorial Calculus . . . . .	8
2.2	The Calculus of Structures . . . . .	12
2.3	System SKS . . . . .	15
<b>II</b>	<b>Atomic Flows</b>	<b>22</b>
<b>3</b>	<b>Atomic Flows</b>	<b>23</b>
3.1	Paths and Cycles . . . . .	27
3.2	Subflows . . . . .	28
<b>4</b>	<b>Atomic Flows and Derivations</b>	<b>30</b>
4.1	Extracting Flows from Derivations . . . . .	30
4.2	A Normal Form of Derivation . . . . .	36
<b>5</b>	<b>Normal Forms</b>	<b>38</b>
<b>III</b>	<b>Normalisation</b>	<b>43</b>
<b>6</b>	<b>Global Reductions</b>	<b>44</b>

6.1	Simplifier . . . . .	46
6.2	Isolated Subflow Removal . . . . .	49
6.3	Path Breaker . . . . .	55
6.4	Multiple Isolated Subflows Removal . . . . .	62
6.4.1	Threshold Formulae . . . . .	68
<b>7</b>	<b>Local Reductions</b>	<b>77</b>
7.1	Soundness . . . . .	79
7.2	Termination and Confluence . . . . .	81
7.3	Complexity . . . . .	82
<b>8</b>	<b>Main Result</b>	<b>84</b>

DRAFT

# Chapter 1

## Introduction

**TODO** Replace this introduction:

*Proof theory* studies formal representations of mathematical proofs. Formal representations of proofs are needed in order to, among other things, automatically verify or search for proofs and prove consistency of mathematical theories.

Furthermore, representations of proofs are interesting mathematical objects in their own right. In particular, they are related to longstanding open problems: The problem of  $NP$  versus  $co-NP$ , from computational complexity theory, and the problem of *identity of proofs*, can both be stated in terms of representations of proofs.

The problem of identity of proofs was originally intended by Hilbert to be the 24th problem of his famous Paris lecture in 1900 [Thi03], and it amounts to answering the question: *When are two proofs the same?* This thesis is part of a program which tries to solve the identity of proofs problem in the setting of propositional logic.

At face value, the representation of a proof is a syntactic object, typically a tree of strings of symbols. This kind of object is not very good at conveying the ‘essence’, ‘meaning’ or *semantics* of the proof it represents. In fact, usually, the representation is extremely verbose and the underlying proof is almost completely obscured by inessential details.

*Bureaucracy* is the name we give to ‘inessential details’ in the representations of proofs. We have an instance of bureaucracy, if morally independent parts of a proof are represented as if there is a dependency between them. A consequence of bureaucracy is that proofs that are essentially the same are given seemingly different representations.

In order to solve Hilbert’s problem we need to find a way of extracting the semantics of a proof from its representation. Furthermore, depending on the context and the purpose of solving the problem, different notions of semantics might be desirable. An ideal solution would therefore not be to give one answer, but a to provide a framework for giving different answers for different purposes.

Two trivial *syntactic* solutions spring to mind:

- ‘two proofs are the same if they prove the same statement’. The problem with this solution is that it would identify proofs of widely different sizes and proofs based on completely different ideas; and
- ‘two proofs are the same if their representations are the same’. The problem with this solution is that it would distinguish proofs that are the same modulo bureaucracy.

A third syntactic solution is based on *normalisation*. Normalisation is the process of manipulating a proof in order to turn it into a *normal form*. Traditionally, normalisation means *cut elimination*. A *cut* is an inference rule embodying the concept of *modus ponens*, or the use of lemmata in a proof. A proof is in normal form if it does not contain an instance of the cut rule. Cut elimination is the most important technique in structural proof theory. From the point of view of computational interpretation of proofs, a cut corresponds to composition and cut elimination corresponds to  $\beta$ -reduction.

Using normalisation we can say: ‘two proofs are the same if they have the same normal form’. In terms of cut elimination, this approach has two problems: cut elimination is not confluent, so the normal form is not unique; and the size of the cut-free form of a proof might depend exponentially on the size of the original proof, so proofs of widely different sizes might be identified.

*Algebraic* solutions to Hilbert’s problem can be given in terms of *categorical axiomatisations* of proofs, *i.e.*, by defining equations on the representations of proofs and consider ‘two proofs to be the same if they belong to the same equivalence class’. The problems with this approach is that it is not necessarily easy to decide if two proofs are equivalent, and arguing about the size of the proofs modulo equations is not straightforward.

The approach we are following is a *geometric* one. We consider proofs to be essentially geometric objects, and their shape is their semantics. Like the algebraic approach this will allow us to have a bureaucracy free semantics, and like the syntactic approach this provides us with concrete representations of proofs whose size we can argue about. Finally, in order to provide for different notions of semantics, we consider normalisation to manipulate the shape of proofs under geometric invariants. I will now outline how we intend to obtain the geometric essence of proofs, and where this thesis fits into the program.

A language for representing proofs is called a *formalism*, and the rules by which a proof is constructed are called *inference rules*. We believe there is strong evidence that bureaucracy in the representation of proofs is what obscures their meaning, and that eliminating bureaucracy is what will allow us to discover their geometric essence. The two sources of bureaucracy we try to eliminate are: Bureaucracy caused by the formalism, and bureaucracy caused by the inference rules. These sources of bureaucracy are closely related as the formalism dictates what kind of inference rules are allowed.

A proof contains bureaucracy due to deficiencies in the formalism, if inference rules or sub-proofs can be trivially permuted. This kind of bureaucracy is often a consequence of a proof being represented as a list or a binary tree, which is traditionally the case, as opposed to something more general like a graph. The problem can be manifested by two independent sub-proofs being represented either as the first depending on the second, the

second depending on the first, or some intermediate ‘interleaving’ of the two. A ‘correct’ representation might be that the two sub-proofs are conducted in parallel (as is possible in the formalism presented in this thesis), or that one proof is, in a certain sense, conducted inside the other (as will be possible in the formalism we are developing as the successor of the one in this thesis).

The kind of bureaucracy stemming from deficiencies in the inference rules is a bit more subtle. Intuitively, if an inference rule instance ‘does too much at once’, *i.e.*, it can be replaced by several ‘smaller’ inference rule instances, it might create dependencies between parts of the proofs which do not morally depend on each other. In order to replace one inference rule occurrence by several other occurrences, we might need to discover new rules, which in turn might require a change in the formalism. In other words, recognising and eliminating this kind of bureaucracy is not necessarily straightforward.

We are influenced by Girard’s *linear logic* and *proof nets* [Gir87]. Linear logic is, roughly speaking, a restriction of classical logic by only allowing *linear* inference rules, *i.e.*, rules that do not duplicate or destroy formulae. Proof nets are geometric representations of linear logic proofs, which identify proofs modulo bureaucracy.

In the same way that linearity gave proof nets in the case of linear logic, we want to find a geometric representation of classical logic proofs. However, classical logic can not be represented using only linear rules. Moreover, we are not able to generalise linearity in the traditional, Gentzen style formalisms in a way that is useful to us.

*Deep inference* [Gug07] is a methodology which allows for more general formalisms than Gentzen’s traditional *natural deduction* and *sequent calculus* [Gen69]. In particular, deep-inference formalisms can express classical logic using only *local* [BT01] inference rules, something which is impossible in the traditional formalisms [Brü03b].

Locality is a generalisation of linearity. An inference rule is said to be local if the time needed to verify an instance of the rule is bounded by a constant (under certain mild hypotheses). In addition to linear rules, a second class of local rules are the *atomic* ones. Atomic rules are restricted to only apply to atoms. Unlike linear rules, atomic rules can duplicate or destroy formulae, but the size of the formula is bounded. Not all rules are local: In order to verify an instance of a rule that duplicates a formula of arbitrary size, the two copies of the formula must be compared. Since the size of the formula is unbounded, the rule is not local.

Intuitively, since local rules can only depend on a limited amount of information, the ‘interdependence’ of instances of local rules is limited. In terms of bureaucracy, by translating from non-local to local rules we are able to observe that more permutations are possible, and hence that more sub-proofs are independent. In this sense, we reveal more bureaucracy. The challenge lies in discovering geometric objects which represent proofs modulo this bureaucracy.

In summary, deep inference makes locality possible; locality reveals bureaucracy; eliminating bureaucracy allows us to represent proofs as geometric objects; whose shape we conjecture will lead us to a nice semantics. We like to express this using the following slogan:

Locality     $\rightarrow$     Geometry     $\rightarrow$     Semantics .

As mentioned above, proofs sometimes need to be normalised in order to be compared. The contribution of this thesis is a geometric language we call *atomic flows* [GG08], which provides a general view of normalisation.

As for natural deduction and the sequent calculus, we intend normalisation as cut elimination. However, deep-inference formalisms have a certain symmetry with respect to the cut rule, which is not present in the traditional formalisms. In particular, it means that we have an inference rule (the *interaction*) that is the proper deMorgan dual of the cut. Furthermore, cut elimination is only possible in deep inference if this symmetry is broken, in order to correspond to the asymmetry in the traditional formalisms.

Atomic flows are motivated by wanting to generalise traditional normal forms, based on the absence of the cut, to a new notion of normal forms based on geometric properties. Atomic flows are special kinds of labelled directed graphs, defined by discarding from derivations all but the information about causal relations between creation and destruction of atoms. Due to deep inference allowing classical logic to be expressed using local inference rules, all the logical information is contained in the linear rules, which we discard, and all the structural information is contained in the atomic rules, which we keep.

We claim that atomic flows give a more general view of normalisation because they provide new normal forms, of which the traditional normal forms are special cases; at the same time they show that normalisation is a less delicate process than was previously believed.

Since we argue in terms of the geometric properties of atomic flows, we are able to describe a symmetric notion of cut-freeness, which we call *streamlining*. Furthermore, in the special, asymmetric, case where cut elimination makes sense, the notions of streamlining and cut elimination coincide. We are also able to show that, contrary to expectations, streamlining can be performed in less than exponential time.

Conventional wisdom teaches us that in order to achieve normalisation, a certain *harmony* between the inference rules is needed. In other words, if we add or exchange inference rules it is not expected that our normalisation results will trivially continue to work.

However, atomic flows discard all information about logical relations and linear rules, so we found it surprising that the remaining information is sufficient to design normalisation procedures. In particular, we have found no difficulties in using our normalisation procedures with different formalisms, different logical connectives and different linear rules, as long as all the inference rules are local and implicationally complete for propositional classical logic.

Some of the results in this thesis are, or will be, available in journal articles. One article has been published, one has been submitted and three are still being written. They are, respectively:

- atomic flows were introduced and the first normalisation results were shown in *Normalisation Control in Deep Inference via Atomic Flows* with Alessio Guglielmi, published in *Logical Methods in Computer Science*;
- quasipolynomial cut elimination was shown using atomic flows in *Quasipolynomial*

*Normalisation in Deep Inference via Atomic Flows and Threshold Formulae*, with Paola Bruscoli, Alessio Guglielmi and Michel Parigot, submitted to *Mathematical Structures in Computer Science*;

- we are showing refined and generalised normalisation results in *Normalisation Control in Deep Inference via Atomic Flows II*, with Alessio Guglielmi;
- we are showing (among other things) that there is a polynomial relationship between the size of a derivation and the size of its associated atomic flow in *On the Complexity of the Switch and Medial Rules*, with Paola Bruscoli, Alessio Guglielmi and Lutz Straßburger; and
- we are defining two new deep-inference formalisms, for which atomic flows and the associated normalisation procedures are invariants in *Formalism A and Formalism B*, with Alessio Guglielmi and Michel Parigot.

Strictly speaking, our normalisation results could be expressed without the use of atomic flows. However, the principal advantage of atomic flows is that it allows us to use a graphical language to describe the gist of our ideas. Once the language is mastered, the most of the technical details of this thesis can easily be reconstructed based on the illustrations alone. Furthermore, atomic flows provide an intuition for working with normalisation, without which we would not have been able to discover our results.

In these three years we could see that many people saw something in our atomic flows. In particular, I have been working on using atomic flows for implicit computational complexity, finding computational interpretations, as well as normalisation of intuitionistic, modal and first-order logics.

At the moment atomic flows are a tool for studying normalisation. However, the most interesting direction of research on atomic flows for the future is, in my opinion, to define a new formalism based on atomic flows with the aim of solving the problem of identity of proofs. Atomic flows are promising in this regard, since they are bureaucracy free, they represent how proofs behave under different notions of normalisation, and they respect the size of proofs. However, at least two important open questions remain: how much information needs to be added to atomic flows before they preserve the validity of proofs; and how can we efficiently decide if two atomic flows are isomorphic.

The thesis is split in three parts.

- In the first part I introduce the deep-inference formalism we will be using, and a local system for classical propositional logic.
- In the second part I define atomic flows, show how flows are related to derivations and define normal forms of derivations in terms of their associated atomic flows.
- Finally, in the last part I define several normalisation procedures based on transforming atomic flows.

**Part I**  
**Derivations**

DRAFT

## Chapter 2

# Propositional Classical Logic

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

**T:** ‘bureaucracy free’ -> ‘less bureaucracy’.

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].

**T:** Removed mentioning of ‘bureaucracy’:

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 68).

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 7–8 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  $\alpha, \beta, \gamma, \delta$  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  $\alpha$  and  $\beta$ , 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  $\alpha$  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  $\alpha$  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  $\alpha$  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  $\alpha\{a/\bar{b}, \bar{b}/c\} \equiv \bar{b} \wedge [\bar{c} \vee (\bar{b} \wedge [b \vee c])]$ . 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  $\alpha$  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  $\alpha$  by formula  $\beta_i$ ; for example if  $\alpha \equiv A \vee (b \wedge c)$  then  $\alpha\{A/(c \wedge \bar{b})\} \equiv (c \wedge \bar{b}) \vee (b \wedge c)$ .

**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*  $\rho$  is an expression  $\rho \frac{\alpha}{\beta}$ , where the formulae  $\alpha$  and  $\beta$  are called *premiss* and *conclusion*, respectively. A (*deductive*) *system* is a finite set of inference rules. An *inference rule instance*  $\rho \frac{\gamma}{\delta}$  of  $\rho \frac{\alpha}{\beta}$  is such that  $\gamma$  and  $\delta$  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 substitution  $\{A_1/\beta_1, A_2/\beta_2, \dots\}$ .

## 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  $\mathcal{S}$ , and formulae  $\alpha$  and  $\beta$ ; a (*functorial calculus*)

*derivation*  $\Psi$  in  $\mathcal{S}$  from  $\alpha$  to  $\beta$ , denoted  $\Psi \parallel_{\mathcal{S}}^{\alpha}$ , is defined to be

1. a formula:  $\Psi = \alpha \equiv \beta$ ;
2. a *vertical composition*:

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

where  $\rho \frac{\beta'}{\alpha'}$  is an instance of an inference rule from  $\mathcal{S}$ , and  $\Phi_1 \parallel^{\alpha}$  and  $\Phi_2 \parallel^{\alpha'}$  are derivations; or

3. a *horizontal composition*:

$$\Psi = \frac{\alpha_1}{\Phi_1 \parallel} \wedge \frac{\alpha_2}{\Phi_2 \parallel} \quad \text{or} \quad \Psi = \frac{\alpha_1}{\Phi_1 \parallel} \vee \frac{\alpha_2}{\Phi_2 \parallel},$$

$$\frac{\beta_1}{\beta_1} \quad \beta_2$$

where  $\Phi_1 \parallel^{\alpha_1}$  and  $\Phi_2 \parallel^{\alpha_2}$  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  $\Psi$ , denoted  $|\Psi|$ , is defined to be the sum of the size of the formulae appearing in  $\Psi$ .

**Convention 2.1.2.** Given derivations  $\Phi_1 \parallel^{\alpha_1}$ ,  $\Phi_2 \parallel^{\alpha_2}$  and  $\Phi_3 \parallel^{\alpha_3}$ , and inference rule instances  $\rho_1 \frac{\beta_1}{\alpha_2}$  and  $\rho_2 \frac{\beta_2}{\alpha_3}$  we consider

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

to be equal, and we denote them both by

$$\begin{array}{c} \alpha_1 \\ \Phi_1 \parallel \\ \beta_1 \\ \rho_1 \frac{\quad}{\quad} \\ \alpha_2 \\ \Phi_2 \parallel \\ \beta_2 \\ \rho_2 \frac{\quad}{\quad} \\ \alpha_3 \\ \Phi_3 \parallel \\ \beta_3 \end{array} .$$

**T:** Added the following remark:

*Remark 2.1.3.* If desirable, Convention 2.1.2 on the previous page could be made redundant by forcing associativity of horizontal composition in Definition 2.1.1 on page 8. 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  $\frac{\alpha}{\Phi \parallel \beta}$  and a context  $\xi \{ \}$ , a derivation  $\frac{\xi \{ \alpha \}}{\Psi \parallel \xi \{ \beta \}}$ , 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

$$\begin{aligned} \xi \{ \} &\equiv \xi' \{ \} \wedge \gamma, & \xi \{ \} &\equiv \gamma \wedge \xi' \{ \}, \\ \xi \{ \} &\equiv \xi' \{ \} \vee \gamma \quad \text{or} & \xi \{ \} &\equiv \gamma \vee \xi' \{ \}. \end{aligned}$$

for some formula  $\gamma$  and a context  $\xi' \{ \}$ . By the inductive hypothesis we can construct the

derivation  $\frac{\xi' \{ \alpha \}}{\Psi' \parallel \xi' \{ \beta \}}$ , so the result follows by case (3) of Definition 2.1.1 on page 8.  $\square$

**Notation 2.1.5.** Given a derivation  $\frac{\alpha}{\Phi \parallel \beta}$  and a context  $\xi \{ \}$ , the derivation  $\frac{\xi \{ \alpha \}}{\xi \{ \beta \}}$  constructed in the proof of Lemma 2.1.4 is denoted  $\xi \{ \Phi \}$ .

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

*Proof.* We argue by structural induction on  $\Phi_1$

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

2. if

$$\Phi_1 = \rho \frac{\begin{array}{c} \alpha \\ \Phi_1' \parallel \\ \beta' \\ \alpha' \\ \Phi_1'' \parallel \\ \beta \end{array}}{\gamma},$$

then, by the inductive hypothesis, we can construct  $\Psi' \parallel$ , with size  $|\Phi_1''| + |\Phi_2| - |\beta|$ , we can then build

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

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

3. if

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

we argue by structural induction on  $\Phi_2$ :

- (a) if  $\Phi_2$  is a formula (resp., a vertical composition), the result follow by a symmetric argument to case 1 (resp., 2) above.  
 (b) if

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

then, by the first inductive hypothesis, we can construct

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

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

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

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

□

**Notation 2.1.7.** Given derivations  $\frac{\alpha}{\beta}$  and  $\frac{\beta}{\gamma}$ , the derivation  $\frac{\alpha}{\gamma}$  constructed in the proof of Lemma 2.1.6 on page 10 is denoted:

$$\frac{\frac{\alpha}{\beta}}{\gamma} .$$

## 2.2 The Calculus of Structures

We now present the calculus of structures and in Theorem 2.2.2 on the following page and Theorem 2.2.6 on page 14 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  $\mathcal{S}$ , a set of formulae,  $\mathcal{F}$ , and  $\alpha$  and  $\beta$  from  $\mathcal{F}$ ;

a *calculus of structures derivation*  $\Psi$  in  $\mathcal{S}$  from  $\alpha$  to  $\beta$ , denoted  $\frac{\alpha}{\beta}$ , is defined to be

1. a formula:  $\Psi = \alpha \equiv \beta$ ; or
2. a *vertical composition*:

$$\Psi = \rho \frac{\frac{\alpha}{\beta}}{\xi\{\alpha'\}} ,$$

where  $\rho \frac{\beta'}{\alpha'}$  is an instance of an inference rule from  $\mathcal{S}$ , and  $\frac{\alpha}{\xi\{\beta'\}}$  and  $\frac{\xi\{\alpha'\}}{\beta}$  are calculus of structures derivations.

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

**Theorem 2.2.2.** A calculus of structures derivation  $\frac{\alpha}{\Phi}$  can be transformed into a functorial

calculus derivation  $\frac{\alpha}{\Psi}$  such that  $|\Psi| \leq |\Phi|$ .

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

$$\Phi = \rho \frac{\frac{\alpha}{\Phi_1} \xi\{\beta'\}}{\xi\{\alpha'\} \frac{\Phi_2}{\beta}}.$$

By the inductive hypothesis, there are functorial calculus derivations  $\frac{\alpha}{\Psi_1} \xi\{\beta'\}$  and  $\frac{\xi\{\alpha'\}}{\Psi_2} \beta$ , such that  $|\Psi_1| \leq |\Phi_1|$  and  $|\Psi_2| \leq |\Phi_2|$ . By Lemma 2.1.4 on page 10, there is a functorial calculus derivation  $\xi \left\{ \rho \frac{\beta'}{\alpha'} \right\}$ , with size  $|\xi\{\beta'\}| + |\alpha'| + |\beta'|$ . By Lemma 2.1.6 on page 10, we can

combine the three functorial calculus derivations to create  $\frac{\alpha}{\Psi}$ , with size  $|\Psi_1| + |\Psi_2| + |\xi\{\beta'\}| + |\beta'| + |\alpha'| - |\xi\{\beta'\}| - |\beta'| - |\xi\{\beta'\}| - |\alpha'| = |\Psi_1| + |\Psi_2| - |\xi\{\beta'\}| \leq |\Phi_1| + |\Phi_2| = |\Phi|$ .  $\square$

**Example 2.2.3.** Figure 4-1 on page 32 has three examples of calculus of structures derivations transformed into functorial calculus derivations.

**Lemma 2.2.4.** Given a calculus of structures derivation  $\frac{\alpha}{\Phi}$  and a context  $\xi\{\beta\}$ , a calculus of

structures derivation  $\frac{\xi\{\alpha\}}{\Psi} \xi\{\beta\}$  can be constructed, such that the number of inference rule instances in  $\Psi$  is the same as the number of inference rule instances in  $\Phi$ , and the size of the largest formula in  $\Psi$  is the sum of the largest formula in  $\Phi$  and  $|\xi\{\beta\}|$ .

*Proof.* The statements follows by structural induction on  $\Phi$ .  $\square$

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

of the number of inference rule instances in  $\Phi_1$  and  $\Phi_2$  combined, and the largest formula in  $\Psi$  is the largest formula of  $\Phi_1$  or the largest formula of  $\Phi_2$ .

*Proof.* The statement follows by structural induction on  $\Phi_1$ .  $\square$

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

*Proof.* We first show how to construct  $\Psi$ : The base cases, when  $\Phi$  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):

$$\Phi = \frac{\alpha_1}{\beta_1} \wedge \frac{\alpha_2}{\beta_2} .$$

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

$$\frac{\alpha_1 \wedge \alpha_1}{\beta_1 \wedge \alpha_1} \quad \text{and} \quad \frac{\beta_1 \wedge \alpha_2}{\beta_1 \wedge \beta_2} ,$$

and by Lemma 2.2.5 there exists a calculus of structures derivation  $\frac{\alpha_1 \wedge \alpha_2}{\beta_1 \wedge \beta_2}$ .

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

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 for classical propositional logic in deep inference SKS [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 19, 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:

$$\begin{array}{ccc} \text{ai}\downarrow \frac{t}{a \vee \bar{a}} & \text{aw}\downarrow \frac{f}{a} & \text{ac}\downarrow \frac{a \vee a}{a} \\ \text{ai}\uparrow \frac{a \wedge \bar{a}}{f} & \text{aw}\uparrow \frac{a}{t} & \text{ac}\uparrow \frac{a}{a \wedge a} \end{array},$$

the *logical* inference rules:

$$\begin{array}{c} \frac{A \wedge [B \vee C]}{(A \wedge B) \vee C} \\ \text{s} \end{array} \quad \begin{array}{c} \frac{(A \wedge B) \vee (C \wedge D)}{[A \vee C] \wedge [B \vee D]} \\ \text{m} \end{array},$$

and the *invertible (logical)* rules:

$$\begin{array}{cccc} \frac{A \vee B}{B \vee A} =_{\vee c} & \frac{A \wedge B}{B \wedge A} =_{\wedge c} & \frac{A \vee [B \vee C]}{[A \vee B] \vee C} =_{a\downarrow} & \frac{(A \wedge B) \wedge C}{A \wedge (B \wedge C)} =_{a\uparrow} \\ \frac{A}{A \vee f} =_{f\downarrow} & \frac{A}{A \wedge t} =_{t\downarrow} & \frac{t \wedge A}{A} =_{f\uparrow} & \frac{f \vee A}{A} =_{t\uparrow} \\ \frac{f}{f \wedge f} =_{f\wedge\downarrow} & \frac{t}{t \vee t} =_{t\vee\downarrow} & \frac{t \vee t}{t} =_{f\wedge\uparrow} & \frac{f \wedge f}{f} =_{t\vee\uparrow} \end{array}.$$

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 30. We now show that the two approaches are morally the same.

**Definition 2.3.2.** We define the relation  $=$  such that, given formulae  $\alpha$  and  $\beta$ ,  $\alpha = \beta$  if there

is a derivation  $\Phi \parallel \frac{\alpha}{\beta} \{ =_{\vee c}, =_{\wedge c}, =_{a\downarrow}, =_{a\uparrow}, =_{f\downarrow}, =_{t\downarrow}, =_{f\uparrow}, =_{t\uparrow}, =_{f\wedge\downarrow}, =_{t\vee\downarrow}, =_{f\wedge\uparrow}, =_{t\vee\uparrow} \}$ .

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

*Remark 2.3.4.* By Notation 2.3.3 on the preceding page and Lemma 2.1.4 on page 10, for any formulae  $\alpha$  and  $\beta$  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 preceding 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

$$\frac{\alpha}{\beta} \Big\| \{ =_{\vee c}, =_{\wedge c}, =_a \downarrow, =_a \uparrow, =_f \downarrow, =_t \downarrow, =_f \uparrow, =_t \uparrow, =_{f \wedge} \downarrow, =_{t \vee} \downarrow, =_{f \wedge} \uparrow, =_{t \vee} \uparrow \}$$

whose size depends at most quadratically on the sum of the sizes of  $\alpha$  and  $\beta$ .

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

$$\frac{\alpha}{\beta} \xrightarrow{\rho} \frac{\alpha'}{\beta'}$$

we sometimes write

$$\frac{\alpha}{\beta} \cdot$$

*E.g.*, instead of the derivation

$$\frac{\frac{\frac{\alpha \vee \beta}{\beta \vee \alpha} \wedge \gamma}{\gamma \wedge [\beta \vee \alpha]} \xrightarrow{s} \frac{(\gamma \wedge \beta) \vee \alpha}{\alpha \vee \frac{\gamma \wedge \beta}{\beta \wedge \gamma}} \xrightarrow{=_{\vee c}} \frac{\gamma \wedge [\beta \vee \alpha]}{\alpha \vee (\beta \wedge \gamma)} \xrightarrow{=_{\wedge c}} \frac{[\alpha \vee \beta] \wedge \gamma}{\alpha \vee (\beta \wedge \gamma)} \xrightarrow{s} \frac{[\alpha \vee \beta] \wedge \gamma}{\alpha \vee (\beta \wedge \gamma)}$$

we write

$$\frac{[\alpha \vee \beta] \wedge \gamma}{\alpha \vee (\beta \wedge \gamma)} \cdot$$

See the proofs of Theorems 6.3.2 to 6.4.4 on pages 56–64 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  $\xi\{t\}$  and a formula  $\alpha$  there exist derivations  $\frac{\alpha \wedge \xi\{t\}}{\xi\{s\}}$  and  $\frac{\xi\{f\} \vee \alpha}{\xi\{s\}}$

$\xi\{\alpha\}$   
 $\frac{\xi\{f\} \vee \alpha}{\xi\{s\}}$ ; both of whose size depend at most quadratically on the size of  $\xi\{\alpha\}$ .

**T:** atoms  $\rightarrow$  atom occurrences

*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  $\xi\{t\}$ . The base case,  $\xi\{t\} = \{t\}$ , is trivial and the inductive cases are:

$$\frac{\alpha \wedge \xi\{t\}}{\xi\{s\}} = \frac{\alpha \wedge \xi\{t\}}{\xi\{t\} \vee \beta} \stackrel{s}{=} \frac{\left[ \begin{array}{c} \alpha \wedge \xi\{t\} \\ \Psi \wedge \beta \\ \xi\{\alpha\} \end{array} \right]}{\xi\{s\}} \quad \text{and} \quad \frac{\alpha \wedge \xi\{t\}}{\xi\{s\}} = \frac{\left( \begin{array}{c} \alpha \wedge \xi\{t\} \\ \Psi \wedge \beta \\ \xi\{\alpha\} \end{array} \right)}{\xi\{s\}},$$

for some  $\xi\{t\}$  and  $\beta$  where  $\beta$  is not a unit and  $\Psi$  and  $\Psi'$  exist by the inductive hypothesis.  $\square$

**Notation 2.3.9.** We often write  $\frac{\alpha \wedge \xi\{t\}}{\xi\{s\}}$  and  $\frac{\xi\{f\} \vee \alpha}{\xi\{s\}}$ , instead of, respectively, the

derivations  $\frac{\alpha \wedge \xi\{t\}}{\xi\{s\}}$  and  $\frac{\xi\{f\} \vee \alpha}{\xi\{s\}}$ , as defined in the proof of Lemma 2.3.8. Instead of the derivation

$$\frac{\frac{\xi\{f\} \vee \alpha}{\xi\{s\}} \wedge \xi\{t\}}{\xi\{s\}} \stackrel{s}{=} \frac{\xi\{f\} \vee \frac{\alpha \wedge \xi\{t\}}{\xi\{s\}}}{\xi\{s\}}$$

we write  $\frac{\xi\{f\} \vee \xi\{\alpha\}}{\xi\{s\}}$ .

**T:** Added Lemma with proof. This was implicitly proven in Theorem 6.4.17 on page 75.

We now show a consequence of the previous Lemma, which will be very useful in Subsection 6.4.1 on page 68.

**Lemma 2.3.10.** Given a formula  $\alpha$  and an atom  $a$ , there exist derivations  $\frac{a \wedge \alpha\{a/t\}}{\alpha\{a/f\} \vee a}$  and  $\frac{\alpha}{\alpha\{a/f\} \vee a}$

$\frac{\alpha}{\alpha\{a/f\} \vee a}$ ; both of whose size depend at most quadratically on the size of  $\alpha$ .

*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  $\alpha$ , 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:

$$\frac{\frac{a}{a \wedge a} \wedge (\xi\{a/t\})\{t\}}{\alpha \wedge (\xi\{a/t\})\{a\}} \quad \frac{\|\{s\}\}}{\alpha}$$

□

**T:** Added:

For an example of the use of Lemma 2.3.10 on the previous page see Remark 2.3.16 on page 20.

**Lemma 2.3.11.** *Given a formula  $\alpha$ , there exist derivations  $\frac{f}{\alpha} \wedge \frac{\alpha}{t}$  and  $\frac{\alpha}{\alpha} \wedge \frac{t}{t}$ ; both of whose size depend at most quadratically on the size of  $\alpha$ .*

**T:** atoms  $\rightarrow$  atom occurrences

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

$$\frac{\alpha\{a_1/f, \dots, a_n/f\}}{\alpha} \quad \frac{\|\{s\}\}}{\alpha}$$

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

$$\frac{\alpha\{a_1/f, \dots, a_n/f\}}{\alpha} \quad \text{or} \quad \frac{\frac{f}{f \wedge [t \vee f]} \wedge \frac{\alpha}{(f \wedge t) \vee t}}{t}$$

□

**Lemma 2.3.12.** *Given a formula  $\alpha$ , there exist derivations  $\frac{\alpha \vee \alpha}{\alpha} \wedge \frac{\alpha}{\alpha \wedge \alpha}$  and  $\frac{\alpha}{\alpha} \wedge \frac{\alpha}{\alpha}$ ; both of whose size depend at most quadratically on the size of  $\alpha$ .*

**T:** number of atoms in  $\alpha \rightarrow$  size of  $\alpha$ ; added two base casse

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

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

and two inductive cases:

$$\frac{m \quad \frac{(\alpha \wedge \beta) \vee (\alpha \wedge \beta)}{\alpha \vee \alpha \quad \beta \vee \beta}}{\frac{\|\{\text{ac}\downarrow, m\}}{\alpha} \wedge \frac{\|\{\text{ac}\downarrow, m\}}{\beta}}{\alpha \wedge \beta}} \quad \text{and} \quad \frac{= \quad \frac{[\alpha \vee \beta] \vee [\alpha \vee \beta]}{\alpha \vee \alpha \quad \beta \vee \beta}}{\frac{\|\{\text{ac}\downarrow, m\}}{\alpha} \vee \frac{\|\{\text{ac}\downarrow, m\}}{\beta}}{\alpha \vee \beta}}.$$

□

**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 previous page and Lemma 2.3.12 on the preceding page correspond to (co)weakening and (co)contractions, respectively. For this reason we sometimes

write the inference rules  $w\downarrow \frac{f}{\alpha}$ ,  $w\uparrow \frac{\alpha}{t}$ ,  $c\downarrow \frac{\alpha \vee \alpha}{\alpha}$  and  $c\uparrow \frac{\alpha}{\alpha \wedge \alpha}$  instead of the derivations  $\frac{f}{\alpha \wedge \alpha}$ ,  $\frac{\alpha}{\alpha \vee \alpha}$ ,  $\frac{\alpha \vee \alpha}{\alpha}$  and  $\frac{\alpha}{\alpha \wedge \alpha}$ , respectively.

$\frac{\alpha}{\alpha \wedge \alpha}$ ,  $\frac{\alpha \vee \alpha}{\alpha}$  and  $\frac{\alpha}{\alpha \wedge \alpha}$ , 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  $\alpha$ . We show by induction on the number of atoms appearing in  $\alpha$  that there exists a proof of  $\alpha$  in SKS. For the base case, let  $\alpha$  consist only of units. Then, since  $\alpha$  is a tautology, we can build

$$\frac{t}{\|\{\text{=}_f\downarrow, \text{=}_t\downarrow, \text{=}_{f\wedge}\downarrow, \text{=}_{t\vee}\downarrow\}} \alpha$$

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

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

$$\frac{t}{\|\{\text{=}_f\downarrow\}} \alpha\{a/f\};$$

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

$$\Phi = \begin{array}{c} t \\ \parallel \\ \alpha\{a/t, \bar{a}/f\} \\ \parallel \{\text{ai}\downarrow, \text{aw}\downarrow\} \\ \alpha\{a/[a \vee \bar{a}]\} \\ \parallel \{\text{ss}\downarrow\} \\ \left[ \begin{array}{c} \bar{a} \vee \dots \vee \bar{a} \\ \alpha \vee \quad \parallel \{\text{ac}\downarrow\} \\ \bar{a} \end{array} \right] \end{array} .$$

Using  $\Phi$  and the inductive hypothesis we can build the desired derivation:

$$\begin{array}{c} t \\ \parallel \\ \alpha\{a/f, \bar{a}/t\} \\ \parallel \{\text{aw}\downarrow\} \\ \alpha\{\bar{a}/t\} \\ \parallel \\ \alpha\{\bar{a}/[\alpha \vee \bar{a}]\} \\ \parallel \{\text{ss}\downarrow\} \\ \alpha \vee \dots \vee \alpha \\ \parallel \text{c}\downarrow \\ \alpha \end{array} .$$

□

*Remark 2.3.15.* Given any formulae  $\alpha$  and  $\beta$  and any context  $\xi\{ \}$ , then, by a construction similar to the one in the proof of Lemma 2.3.8 on page 17, we can build a derivation  $\xi\{\alpha \vee \beta\}$   $\parallel \{\text{s}, =_{\vee}, =_{\wedge}, =_a\downarrow\}$ . If we use this derivation instead of the rule  $\text{ss}\downarrow$  in the proof of Theorem 2.3.14 on the preceding page, it follows that the system

$$\{\text{ai}\downarrow, \text{ac}\downarrow, \text{aw}\downarrow, \text{s}, \text{m}, =_{\vee}, =_{\wedge}, =_a\downarrow, =_f\downarrow, =_t\downarrow, =_{f\wedge}\downarrow, =_{t\vee}\downarrow\}$$

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  $\uparrow$ .

**T:** Changed the following remark to illustrate the use of Lemma 2.3.10 on page 17.

*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 preceding page, by using the

following as the inductive case:

$$\begin{array}{c}
 \begin{array}{c}
 \text{t} \\
 \parallel \\
 \left( \frac{\frac{\text{t}}{a \vee \bar{a}} \wedge \alpha\{a/t, \bar{a}/f\}}{(a \wedge \alpha\{a/t, \bar{a}/f\}) \vee \bar{a}} \wedge \alpha\{a/f, \bar{a}/t\} \right) \\
 \text{s} \frac{\quad}{\quad} \\
 \left( a \wedge \begin{array}{c} \alpha\{a/t, \bar{a}/f\} \\ \parallel\{\text{aw}\downarrow\} \\ \alpha\{a/t\} \\ \parallel\{\text{ac}\uparrow, \text{s}\} \\ \alpha \end{array} \right) \vee \left( \bar{a} \wedge \begin{array}{c} \alpha\{a/f, \bar{a}/t\} \\ \parallel\{\text{aw}\downarrow\} \\ \alpha\{\bar{a}/t\} \\ \parallel\{\text{ac}\uparrow, \text{s}\} \\ \alpha \end{array} \right) \\
 \text{c}\downarrow \frac{\quad}{\quad} \\
 \alpha
 \end{array}
 \end{array}
 ,
 \end{array}$$

where we have used the derivations constructed in the proof of Lemma 2.3.10 on page 17.

**Part II**  
**Atomic Flows**

DRAFT

# Chapter 3

## Atomic Flows

**T:** formalism  $\rightarrow$  invariant

**T:** abstract representation

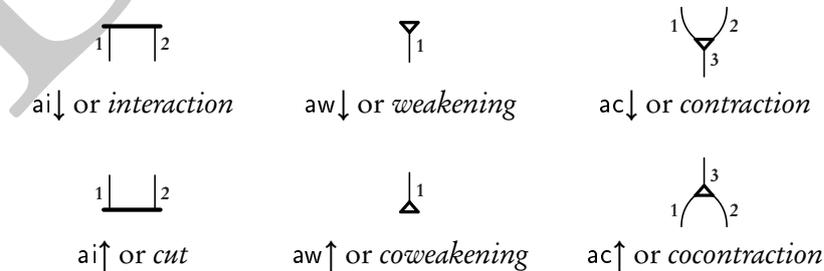
In this chapter we introduce the main tool used in this thesis, a geometric proof invariant called ‘atomic flows’. The main contribution is the novel uses of atomic flows. An atomic flow is a directed graph obtained from a derivation by only retaining information about the creation and destruction of atom occurrences. Atomic flows are, essentially, specialised Buss flow graphs [Bus91, Car97]. Furthermore, atomic flows are also essentially the same as the kind of proof nets developed in [Str05, Str09], based on work done in [LS05]. The only difference is that these proof nets implement (co)associativity of (co)contraction, 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.

Notably, the atomic flow of a derivation completely disregards all the logical relations and associated inference rule instances; so, an atomic flow is not a derivation.

**T:** Deleted here all mention of mapping from flows to derivations, as this was misleading in this context. It will be covered in greater detail later instead.

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:



**Definition 3.0.2.** An (*atomic*) *flow* is a tuple  $(V, E, \eta, up, lo)$ , denoted  $\phi$  or  $\psi$ , such that:

1.  $V$  is a finite set of *vertices*, denoted by  $v$ ;
2.  $E$  is a finite set of *edges*, denoted by  $\epsilon, \iota$  or small numerals  $1, 2, \dots$ ;
3.  $\eta: V \rightarrow \{ai\downarrow, ai\uparrow, aw\downarrow, aw\uparrow, ac\downarrow, ac\uparrow\}$  maps vertices to their *labels*;
4.  $up: E \rightarrow V \cup \{\top\}$  and  $lo: E \rightarrow V \cup \{\perp\}$  are, respectively, the *upper* and *lower* maps, and  $\top$  and  $\perp$  are special vertices not belonging to  $V$ ; we define, for every  $v \in V \cup \{\top, \perp\}$ , the set  $L_v = \{\epsilon \mid up(\epsilon) = v\}$  of *lower edges of  $v$* , the set  $U_v = \{\epsilon \mid lo(\epsilon) = v\}$  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
  - if  $\eta(v) = ai\downarrow$  then  $|L_v| = 2$  and  $|U_v| = 0$ ,
  - if  $\eta(v) = ai\uparrow$  then  $|L_v| = 0$  and  $|U_v| = 2$ ,
  - if  $\eta(v) = aw\downarrow$  then  $|L_v| = 1$  and  $|U_v| = 0$ ,
  - if  $\eta(v) = aw\uparrow$  then  $|L_v| = 0$  and  $|U_v| = 1$ ,
  - if  $\eta(v) = ac\downarrow$  then  $|L_v| = 1$  and  $|U_v| = 2$ ,
  - if  $\eta(v) = ac\uparrow$  then  $|L_v| = 2$  and  $|U_v| = 1$ ;
6. there is no sequence  $\epsilon_1, \dots, \epsilon_b$  of edges of  $V$  such that  $up(\epsilon_i) = lo(\epsilon_{i+1 \pmod{b}})$ , for  $1 \leq i \leq b$ ;
7. there is a *polarity assignment*  $\pi: E \rightarrow \{-, +\}$  such that, for every  $v \in V$ ,
  - (a) if  $\eta(v) \in \{ac\downarrow, ac\uparrow\}$  then  $\pi(E_v) = \{-\}$  or  $\pi(E_v) = \{+\}$ ;
  - (b) if  $\eta(v) \in \{ai\downarrow, ai\uparrow\}$  then  $\pi(E_v) = \{-, +\}$ .

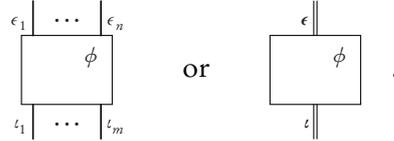
Given an atomic flow  $\phi$ , we say that the sets  $L_\top = \{\epsilon_1, \dots, \epsilon_b\}$  and  $U_\perp = \{\iota_1, \dots, \iota_k\}$  contain, respectively, the *upper* and *lower edges of  $\phi$* .

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  $\perp$  represent the top and bottom of a derivation: we can consider  $\top$  the vertex that creates all the atom occurrences in the premiss and  $\perp$  the vertex that destroys all atom occurrences in the conclusion.

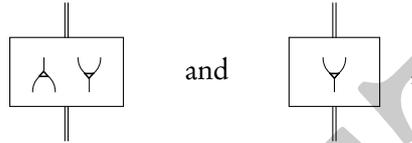
The polarity assignment condition (7) ensures that atoms in (co)contractions have the same polarity, and those in interactions and cuts have dual polarities (as happens in derivations). Every atomic flow has  $2^n$  polarity assignments, where  $n$  is the number of connected components in the graph. 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 3.0.3.** Let  $\phi$  be a flow with upper edges  $\epsilon = \epsilon_1, \dots, \epsilon_n$  and lower edges  $\iota = \iota_1, \dots, \iota_m$ , 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



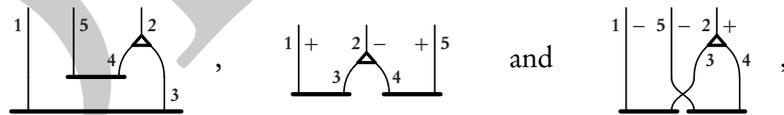
do not contain  $ai\downarrow$ ,  $ai\uparrow$ ,  $aw\downarrow$ ,  $aw\uparrow$  vertices, and the flow to the right does not contain  $ac\uparrow$  vertices.

In general, we represent atomic flows as directed-graph diagrams, except that the special vertices  $\top$  and  $\perp$  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  $\perp$ . Sometimes we identify vertices with their labels.

**Example 3.0.4.** Consider the flow

$$A = (\{ \nu_1, \nu_2, \nu_3 \}, \{ 1, 2, 3, 4, 5 \}, \{ \nu_1 \mapsto ai\uparrow, \nu_2 \mapsto ac\uparrow, \nu_3 \mapsto ai\uparrow \}, \{ 1 \mapsto \top, 2 \mapsto \top, 3 \mapsto \nu_2, 4 \mapsto \nu_2, 5 \mapsto \top \}, \{ 1 \mapsto \nu_1, 2 \mapsto \nu_2, 3 \mapsto \nu_1, 4 \mapsto \nu_3, 5 \mapsto \nu_3 \}) ;$$

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 contraction and two cointeraction vertices; it has three upper edges, 1, 2 and 5, and no lower edges.

**Example 3.0.5.** The flow

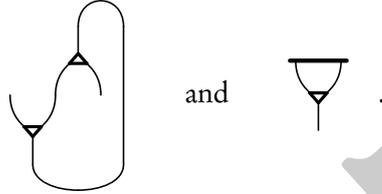


is obtained by juxtaposing (*i.e.*, composing by identifying zero edges):

- three edges,
- a flow obtained by composing a cut vertex with a cocontraction vertex, and
- a flow obtained by composing an identity 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 3.0.6.** 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.

**T:** Rephrased slightly to avoid overfull.

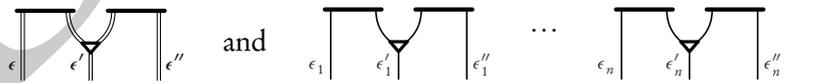
**Definition 3.0.7.** Given two flows  $\phi_1 = (V_1, E_1, \eta_1, up_1, lo_1)$  and  $\phi_2 = (V_2, E_2, \eta_2, up_2, lo_2)$ , an *(atomic) flow isomorphism* between  $\phi_1$  and  $\phi_2$  is a pair of functions  $(f_V, f_E)$ , such that

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

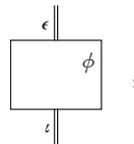
such that, for every  $\epsilon$  in  $E_1$ ,

- for every  $v$  in  $V_1$ ,  $up_1(\epsilon) = v$  (resp.,  $lo_1(\epsilon) = v$ ) if and only if  $up_2(f_E(\epsilon)) = f_V(v)$  (resp.,  $lo_2(f_E(\epsilon)) = f_V(v)$ ); and
- $up_1(\epsilon) = \top$  (resp.,  $lo_1(\epsilon) = \perp$ ) if and only if  $up_2(f_E(\epsilon)) = \top$  (resp.,  $lo_2(f_E(\epsilon)) = \perp$ ).

**Notation 3.0.8.** We extend the double-line notation to collections of isomorphic flows. For example, for  $n \geq 0$ ;  $\epsilon = \epsilon_1, \dots, \epsilon_n$ ;  $\epsilon' = \epsilon'_1, \dots, \epsilon'_n$ ; and  $\epsilon'' = \epsilon''_1, \dots, \epsilon''_n$ , the following diagrams represent the same flow:



**Notation 3.0.9.** Given a flow



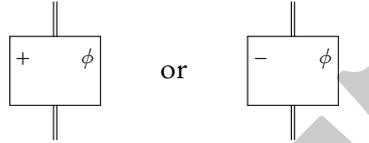
and a flow  $\psi$  which is isomorphic to  $\phi$ , whenever we write

$$\psi = \begin{array}{c} \parallel \\ f(\epsilon) \\ \boxed{f(\phi)} \\ f(i) \\ \parallel \end{array},$$

we mean that  $f$  is a given flow isomorphism between  $\phi$  and  $\psi$ .

**T: Removed convention.**

**Notation 3.0.10.** Given a flow  $\phi$  and a polarity assignment  $\pi$  for  $\phi$ , whenever we write



respectively, we mean that all the edges in  $\phi$  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 24.

**Definition 3.0.11.** Given a flow  $\phi$  and a polarity assignment  $\pi$  for  $\phi$ , the polarity assignment  $\bar{\pi}$  for  $\phi$  is defined to be, for every  $\epsilon$  in  $\phi$ :

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

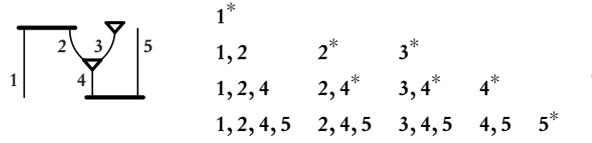
### 3.1 Paths and Cycles

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  $(V, E, \eta, up, lo)$  and  $\epsilon_1, \dots, \epsilon_h \in E$  such that, for  $1 \leq i < h$ , we have  $lo(\epsilon_i) = up(\epsilon_{i+1})$ ,  $up(\epsilon_1) = v$  and  $lo(\epsilon_h) = v'$ , we say that  $\epsilon_1, \dots, \epsilon_h$  is a *path from  $v$  to  $v'$*  and that  $\epsilon_h, \dots, \epsilon_1$  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  $\epsilon_1, \dots, \epsilon_k, \epsilon_{k+1}, \dots, \epsilon_h$  such that  $\epsilon_k \neq \epsilon_{k+1}$  and, for some  $v'' \in V$  with  $\eta(v'') \in \{ai\downarrow, ai\uparrow\}$ , we have that  $\epsilon_1, \dots, \epsilon_k$  is an ai-path from  $v$  to  $v''$  and  $\epsilon_{k+1}, \dots, \epsilon_h$  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:



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

**T:** Removed superfluos clause and rephrased slightly to avoid overfull.

**Definition 3.2.1.** Given two flows  $\phi_1 = (V_1, E_1, \eta_1, up_1, lo_1)$  and  $\phi_2 = (V_2, E_2, \eta_2, up_2, lo_2)$ , we say that  $\phi_1$  is a *subflow* of  $\phi_2$ , if

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

$$up_1(\epsilon) = \begin{cases} up_2(\epsilon) & \text{if } up_2(\epsilon) \in V_1, \\ \top & \text{otherwise.} \end{cases} \quad \text{and}$$

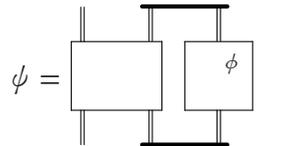
$$lo_1(\epsilon) = \begin{cases} lo_2(\epsilon) & \text{if } lo_2(\epsilon) \in V_1, \\ \perp & \text{otherwise.} \end{cases} \quad ; \text{ and}$$

- if  $v_1$  and  $v_2$  are vertices in  $\phi_1$ , and there is a vertex  $v'$  in  $\phi_2$ , such that there are paths from  $v_1$  to  $v'$  and from  $v'$  to  $v_2$  in  $\phi_2$ , then  $v'$  is a vertex in  $\phi_1$ .

**T:** Renamed a  $\psi$  to a  $\phi$  and added example.

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

**Example 3.2.3.** In the following flow,  $\phi$  is an isolated subflow of  $\psi$ :



For other examples of isolated subflows see Definition 6.2.1 on page 50 and Definition 6.4.1 on page 62.

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

DRAFT



3. for each inference rule instance of  $\Phi$  of kind

$$\begin{array}{cccc}
& \frac{\alpha \wedge [\beta \vee \gamma]}{(\alpha \wedge \beta) \vee \gamma} & , & \frac{(\alpha \wedge \beta) \vee (\gamma \wedge \delta)}{[\alpha \vee \gamma] \wedge [\beta \vee \delta]} & , \\
=_{vc} \frac{\alpha \vee \beta}{\beta \vee \alpha} & , & =_{\wedge c} \frac{\alpha \wedge \beta}{\beta \wedge \alpha} & , & =_{a\downarrow} \frac{\alpha \vee [\beta \vee \gamma]}{[\alpha \vee \beta] \vee \gamma} & , & =_{a\uparrow} \frac{(\alpha \wedge \beta) \wedge \gamma}{\alpha \wedge (\beta \wedge \gamma)} & , \\
=_{f\downarrow} \frac{\alpha}{\alpha \vee f} & , & =_{t\downarrow} \frac{\alpha}{\alpha \wedge t} & , & =_{f\uparrow} \frac{t \wedge \alpha}{\alpha} & \text{and} & =_{t\uparrow} \frac{f \vee \alpha}{\alpha}
\end{array}$$

all the atom occurrences in  $\alpha$ ,  $\beta$ ,  $\gamma$  and  $\delta$  in the premiss are respectively mapped to the same edges of  $\phi$  as the atom occurrences in  $\alpha$ ,  $\beta$ ,  $\gamma$  and  $\delta$  in the conclusion.

*Proof.* We sketch the proof: We proceed by induction on  $\Phi$ . If  $\Phi$  is a formula or a horizontal composition of derivations, the result is immediate. Otherwise, if  $\Phi$  is a vertical composition of two derivations by an inference rule, the result follows by induction and a case analysis of each inference rule of SKS.  $\square$

**T:** I decided not to do this for lack of time (unless you insist of course):

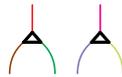
**TODO** Macro for “mapped from occurrences of”.

**Definition 4.1.2.** Given a derivation  $\Phi$ , we say that the flow  $\phi$  constructed in the proof of Proposition 4.1.1 on the previous page is the (atomic) flow associated with the derivation  $\Phi$ . Sometimes, when an atom occurrence  $a$  in  $\Phi$  is mapped to an edge  $\epsilon$  in  $\phi$ , we decorate  $a$  with the label  $\epsilon$  or the label  $\phi$ .

**Example 4.1.3.** Figure 4-1 on the following page has three examples of derivations and their associated flows, where colors are used to indicate the mapping from atom occurrences to edges.

**Definition 4.1.4.** Given a derivation  $\Phi$  with flow  $\phi$ , and an atom  $a$ , the restriction of  $\phi$  to  $a$  is the largest subflow  $\phi_a$  of  $\phi$ , such that every edge of  $\phi_a$  is mapped to from  $a$  or  $\bar{a}$ .

**Example 4.1.5.** Consider the rightmost derivation and its associated flow in Figure 4-1 on the next page. The restriction of this flow to  $a$  is:



**T:** Gave the motivation for the following theorem.

We now show that the definition of atomic flows is ‘minimal’, in the sense that, if we restrict the definition, Proposition 4.1.1 on the preceding page is no longer true.

**Theorem 4.1.6.** Every atomic flow is associated with some derivation.

$$\begin{array}{l}
\text{ai}\downarrow \frac{t}{a \vee \bar{a}} \\
= \frac{(a \wedge t) \vee (t \wedge \bar{a})}{[a \vee t] \wedge [t \vee \bar{a}]} \\
\text{m} \\
= \frac{[a \vee t] \wedge [\bar{a} \vee t]}{([a \vee t] \wedge \bar{a}) \vee t} \\
\text{s} \\
= \frac{(\bar{a} \wedge [a \vee t]) \vee t}{[(\bar{a} \wedge a) \vee t] \vee t} \\
\text{s} \\
= \frac{(a \wedge \bar{a}) \vee t}{\text{ai}\uparrow \frac{f \vee t}{t}} \\
= \frac{f \vee t}{t}
\end{array}$$

$$\begin{array}{l}
\text{ai}\downarrow \frac{(a \wedge [\bar{a} \vee t]) \wedge \bar{a}}{(a \wedge [\bar{a} \vee [\bar{a} \vee a]]) \wedge \bar{a}} \\
= \frac{(a \wedge [(\bar{a} \vee \bar{a}) \vee a]) \wedge \bar{a}}{[(a \wedge [\bar{a} \vee \bar{a}]) \vee a] \wedge \bar{a}} \\
\text{s} \\
\text{ac}\downarrow \frac{[(a \wedge \bar{a}) \vee a] \wedge \bar{a}}{[f \vee a] \wedge \bar{a}} \\
\text{ai}\uparrow \\
= \frac{f \vee a \wedge \bar{a}}{a \wedge \bar{a}} \\
\text{ac}\uparrow \\
= \frac{a \wedge \bar{a}}{(a \wedge a) \wedge \bar{a}} \\
= \frac{a \wedge (a \wedge \bar{a})}{a \wedge f}
\end{array}$$

$$\begin{array}{l}
\text{ac}\uparrow \frac{[a \vee b] \wedge a}{[(a \wedge a) \vee b] \wedge a} \\
\text{ac}\uparrow \frac{[(a \wedge a) \vee (b \wedge b)] \wedge a}{[(a \wedge a) \vee (b \wedge b)] \wedge (a \wedge a)} \\
\text{m} \\
= \frac{([a \vee b] \wedge [a \vee b]) \wedge (a \wedge a)}{([a \vee b] \wedge a) \wedge ([a \vee b] \wedge a)}
\end{array}$$

$$\begin{array}{l}
\frac{t}{a \vee \bar{a}} \\
\text{m} \\
\frac{[a \vee t] \wedge [t \vee \bar{a}]}{\left[ \begin{array}{l} \text{s} \\ \frac{[a \vee t] \wedge \bar{a}}{a \wedge \bar{a}} \vee t \end{array} \right]} \\
\text{s} \\
= \frac{\left( \begin{array}{l} \text{s} \\ \frac{a \wedge \left[ \frac{\bar{a} \vee t}{\bar{a} \vee a} \right]}{a \wedge \frac{\bar{a}}{a}} \wedge \bar{a} \end{array} \right)}{a \wedge \frac{a \wedge \bar{a}}{f}}
\end{array}$$

$$\frac{a}{a \wedge a} \vee \frac{b}{b \wedge b} \wedge \frac{a}{a \wedge a} \\
\text{m} \\
\frac{[a \vee b] \wedge [a \vee b]}{[a \vee b] \wedge [a \vee b]}$$

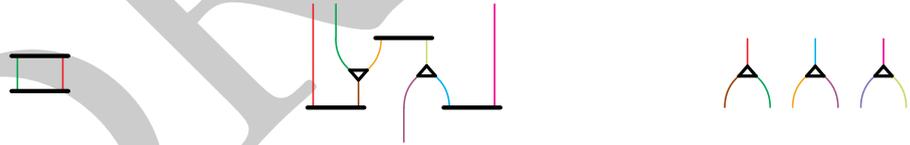


Figure 4-1: Examples of derivations in the calculus of structures (top row), their translation into the functorial calculus (middle row), and the flows associated with the latter (bottom row).

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

$$\frac{(\xi\{t\} \wedge \zeta\{a\}) \vee t}{\|\{s,m\}}, \quad (\xi\{a\} \wedge \zeta\{f\}) \vee t$$

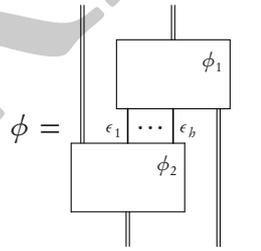
in other words we can ‘move’ the atom  $a$  from the context  $\xi\{ \}$  to the context  $\zeta\{ \}$  by using a derivation whose flow contains no vertices:

$$\begin{aligned} & \left[ \begin{array}{c} \frac{\xi\{t\} \wedge \zeta\{a\}}{ss} \\ \frac{\xi\{a\} \vee \zeta\{f\}}{m} \\ \frac{(\xi\{a\} \wedge t) \vee (t \wedge \zeta\{f\})}{s} \\ \frac{[\xi\{a\} \vee t] \wedge [t \vee \zeta\{f\}]}{s} \\ \frac{[\xi\{a\} \vee t] \wedge [\zeta\{f\} \vee t]}{s} \\ \frac{([\xi\{a\} \vee t] \wedge \zeta\{f\}) \vee t}{s} \end{array} \right] \vee t \\ &= \frac{\left[ \frac{\zeta\{f\} \wedge [\xi\{a\} \vee t]}{s} \right] \vee t}{\|\{s,m\}} \\ &= \frac{(\zeta\{f\} \wedge \xi\{a\}) \vee t}{\|\{s,m\}} \\ &= \frac{(\xi\{a\} \wedge \zeta\{f\}) \vee t}{\|\{s,m\}} \end{aligned}$$

This construction can be used repeatedly to build the derivation  $\Psi$ , for  $b \geq 0$ :

$$\frac{(\xi\{t\} \cdots \{t\} \wedge \zeta\{a_1\} \cdots \{a_b\}) \vee t}{\Psi \|\{s,m\}}, \quad (\xi\{a_1\} \cdots \{a_b\} \wedge \zeta\{f\} \cdots \{f\}) \vee t$$

We can now prove the theorem by induction on the number of vertices of a given flow  $\phi$ . The cases where  $\phi$  only has zero or one vertex are trivial. Let us then suppose that  $\phi$  has more than one vertex; then  $\phi$  can be considered as composed of two flows  $\phi_1$  and  $\phi_2$ , each with fewer vertices than  $\phi$ , as follows:



where  $b \geq 0$  (this can possibly be done in many different ways). By the inductive hypothesis, there exist derivations  $\frac{\gamma}{\Phi_1 \|\{s,m\}}$  and  $\frac{\xi\{a_1^{\epsilon_1}\} \cdots \{a_b^{\epsilon_b}\}}{\Phi_2 \|\{s,m\}}$  whose flows are, respectively,  $\phi_1$  and  $\phi_2$ .





## 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 identity 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

$$\frac{\alpha}{\Phi} \parallel \beta \quad \text{and} \quad \Psi = \frac{\left( \frac{t}{a_1 \vee \bar{a}_1} \wedge \dots \wedge \frac{t}{a_n \vee \bar{a}_n} \wedge \alpha \right)}{\parallel_{\text{SKS}\{\text{ai}\downarrow, \text{ai}\uparrow\}}}, \left[ \beta \vee \frac{b_m \wedge \bar{b}_m}{f} \vee \dots \vee \frac{b_1 \wedge \bar{b}_1}{f} \right],$$

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

**Convention 4.2.2.** Given a derivation  $\Phi$  and an ai-decomposed form of  $\Phi$ :

$$\frac{\left( \frac{t}{a_1 \vee \bar{a}_1} \wedge \dots \wedge \frac{t}{a_n \vee \bar{a}_n} \wedge \frac{t}{c_1 \vee \bar{c}_1} \wedge \dots \wedge \frac{t}{c_k \vee \bar{c}_k} \wedge \alpha \right)}{\parallel_{\text{SKS}\{\text{ai}\downarrow, \text{ai}\uparrow\}}}, \left[ \beta \vee \frac{d_l \wedge \bar{d}_l}{f} \vee \dots \vee \frac{d_1 \wedge \bar{d}_1}{f} \vee \frac{b_m \wedge \bar{b}_m}{f} \vee \dots \vee \frac{b_1 \wedge \bar{b}_1}{f} \right],$$

we sometimes want to single out only some of the interaction or cut instances. We therefore also call the following, partially sequentialised, derivation an ai-decomposed form of  $\Phi$ :

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

$$\left[ \beta \vee \frac{b_m \wedge \bar{b}_m}{f} \vee \dots \vee \frac{b_1 \wedge \bar{b}_1}{f} \right]$$

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

*Proof.* Using Lemma 2.3.8 on page 17 apply, from top-to-bottom and left-to-right, the following transformations to each of the identity and cut instances in  $\Phi$ :

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

to obtain an ai-decomposed form of  $\Phi$ . The size of the ai-decomposed form obtained in this way depends at most cubically on the size of  $\Phi$ , since, by Lemma 2.3.8 on page 17, each of the transformations increase the size of the derivation at most quadratically and the number of transformations is bound by the size of  $\Phi$ .  $\square$

*Remark 4.2.4.* The only reason to insist on performing the transformations in the proof of Theorem 4.2.3 on the preceding page 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  $\Phi$ , the ai-decomposed form of  $\Phi$  obtained as described in the proof of Theorem 4.2.3 on the previous page is called *the (canonical) ai-decomposed form of  $\Phi$* .

## 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 following 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 *weakly streamlined* atomic flows, in case (3) of Definition 5.0.1. An atomic flow is weakly streamlined if it contains no paths from interaction to cut vertices. This is the topic of Chapter 6 on page 44.

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.

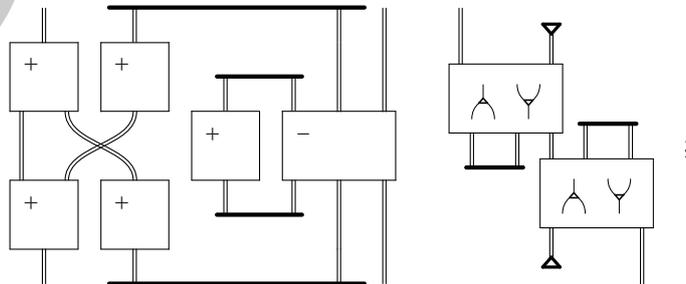
**T:** Added one bullet point for each normal form:

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

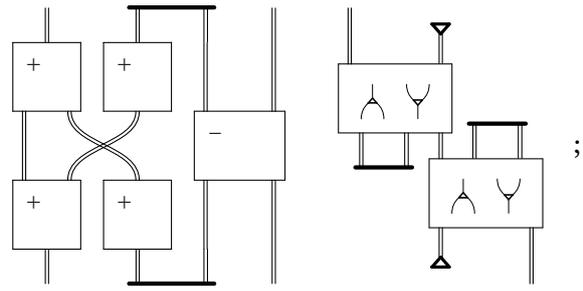
1. a flow is in simple form with respect to a given polarity assignment, if all the edges with a positive polarity assignment can be partitioned into two classes, the ones that belong to paths connecting identity and cut vertices (the rightmost box marked with a + in the below figure) and the ones that 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 positive 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

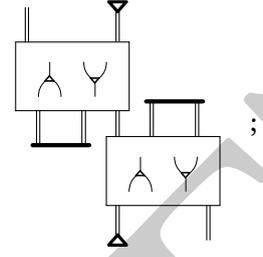
1. *in simple form with respect to the polarity assignment  $\pi$*  if it can be represented as



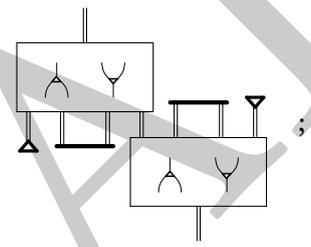
2. *weakly streamlined with respect to the polarity assignment  $\pi$*  if it can be represented as



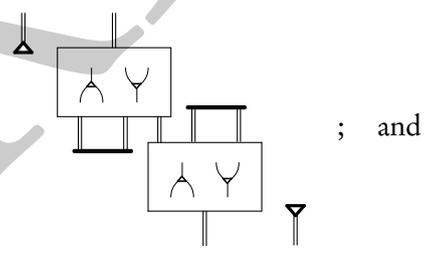
3. *weakly streamlined* if it can be represented as



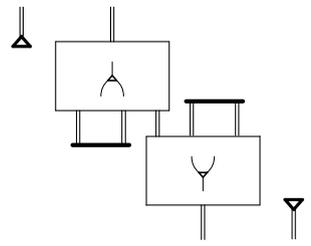
4. *streamlined* if it can be represented as



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



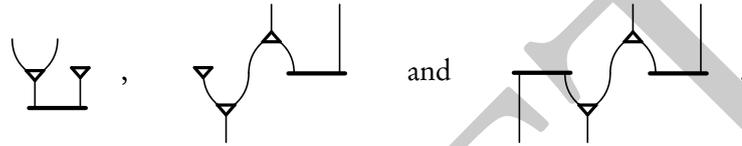
6. *hyper streamlined* if it can be represented as



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

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

**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 39.

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

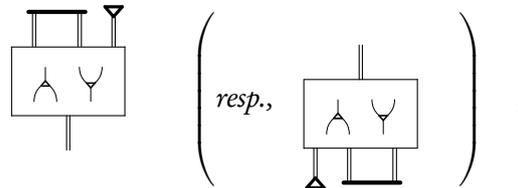
**Proposition 5.0.6.** A streamlined flow with no pair of upper (resp., lower) edges such that there is an  $a$ -path between them, contains no cut (resp., axiom) vertices.

**T:** Removed the use of the name KS, as this was the only place it was used and as it was not defined.

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  $\text{SKS} \setminus \{a_i\uparrow, a_c\uparrow, a_w\uparrow\}$ .

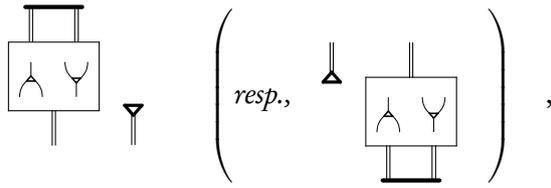
**Proposition 5.0.7.** Given an atomic flow with no upper (resp., lower) edges, it can be represented as

1.



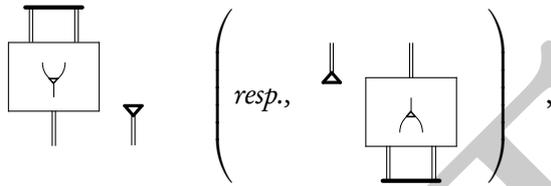
if it is streamlined;

2.



*if it is super streamlined; and*

3.



*if it is hyper streamlined.*

DRAFT

**Part III**

**Normalisation**

DRAFT

## 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 property, 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.

**T:** Added paragraph:

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.

**T:** Changed from  $f$  and  $g$  to  $u$  and  $l$ .

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

We call  $\phi$  a *redex* and  $\psi$  a *contractum* of  $R$ .

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

**T:** Added explanation:

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 following page.

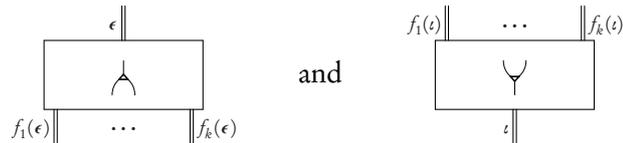
**Definition 6.0.3.** A reduction  $R$  is *sound* if, for every  $\phi$  and  $\psi$ , such that  $\phi R \psi$ , and for every derivation  $\Phi$  with flow  $\phi$ , there is a derivation  $\Psi$  with atomic flow  $\psi$  such that  $\Phi$  and  $\Psi$  have the same premiss and conclusion; in this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

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

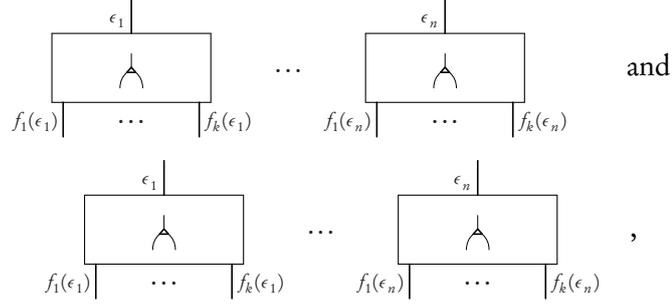
**T:** Added remark:

*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 next page, Definition 6.2.1 on page 50, Definition 6.3.1 on page 55 and Definition 6.4.1 on page 62 we have establish the following convention: Let  $\epsilon = \epsilon_1, \dots, \epsilon_n$ ,  $\iota = \iota_1, \dots, \iota_m$ ,  $\epsilon' = \epsilon'_1, \dots, \epsilon'_n$  and  $\iota' = \iota'_1, \dots, \iota'_m$ , then, when we write



we mean

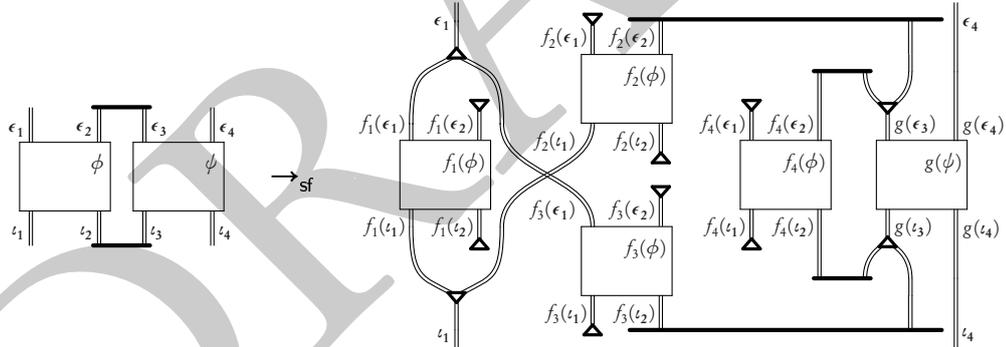


respectively. In other words, edges are not connected in unexpected ways.

## 6.1 Simplifier

Consider a flow  $\phi'$  with polarity assignment  $\pi$ , such that  $\phi$  is the subflow of  $\phi'$  containing all the edges with a positive polarity assignment. We can observe that  $\phi$  contains four types of paths: 1) paths from  $\top$  to  $\perp$ ; 2) paths from an interaction vertex to  $\perp$ ; 3) paths from  $\top$  to a cut vertex; and 4) paths from an interaction vertex to a cut vertex. We can turn  $\phi'$  into simple form with respect to  $\pi$  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  $\phi$  each of which only contains one of the above types of paths.

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



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

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

*Proof.* Let  $\Phi$  be a derivation with flow  $\phi'$ , such that  $\phi' \rightarrow_{sf} \psi'$ . We show that there exists a derivation  $\Psi$  with flow  $\psi'$  and with the same premiss and conclusion as  $\Phi$ . In the following, we refer to the figure in Definition 6.1.1.

Assume all the edges in  $\phi$  are mapped to from occurrences of the atoms  $a_1, \dots, a_n$ , and let

$$\left( \frac{t}{a_1^\phi \vee \bar{a}_1^\phi} \wedge \dots \wedge \frac{t}{a_n^\phi \vee \bar{a}_n^\phi} \wedge \alpha \right)_{\Phi' \parallel \{a_i \downarrow, a_i \uparrow\}} \left[ \beta \vee \frac{a_n^\phi \wedge \bar{a}_n^\phi}{f} \vee \dots \vee \frac{a_1^\phi \wedge \bar{a}_1^\phi}{f} \right]$$

be the ai-decomposed form of  $\Phi$ .

We show several intermediate derivations which will be used to build  $\Psi$ . To make it easier to verify the flow of  $\Psi$ , 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  $\Psi$ .

Consider the substitution

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

We can then obtain, by Proposition 4.1.8 on page 35, the derivation  $\Phi' \sigma$  with flow

$$\begin{array}{ccccc} \begin{array}{|c|} \hline f_1(\epsilon_1) & f_1(\epsilon_2) \\ \hline f_1(\phi) \\ \hline f_1(\iota_1) & f_1(\iota_2) \\ \hline \end{array} & \begin{array}{|c|} \hline f_2(\epsilon_1) & f_2(\epsilon_2) \\ \hline f_2(\phi) \\ \hline f_2(\iota_1) & f_2(\iota_2) \\ \hline \end{array} & \begin{array}{|c|} \hline f_3(\epsilon_1) & f_3(\epsilon_2) \\ \hline f_3(\phi) \\ \hline f_3(\iota_1) & f_3(\iota_2) \\ \hline \end{array} & \begin{array}{|c|} \hline f_4(\epsilon_1) & f_4(\epsilon_2) \\ \hline f_4(\phi) \\ \hline f_4(\iota_1) & f_4(\iota_2) \\ \hline \end{array} & \begin{array}{|c|} \hline g(\epsilon_3) & g(\epsilon_4) \\ \hline g(\psi) \\ \hline g(\iota_3) & g(\iota_4) \\ \hline \end{array} \end{array}.$$

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

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

which allow us to build

$$\frac{\alpha}{\Psi_\top \parallel} \quad \text{and} \quad \frac{\beta \sigma}{\Psi_\perp \parallel},$$

with flows

$$\begin{array}{|c|} \hline f_1(\epsilon_1) & f_3(\epsilon_1) \\ \hline \wedge \\ \hline f_2(\epsilon_1) & f_4(\epsilon_1) \\ \hline \end{array} \parallel \begin{array}{|c|} \hline g(\epsilon_4) \\ \hline \end{array} \quad \text{and} \quad \begin{array}{|c|} \hline f_1(\iota_1) & f_2(\iota_1) \\ \hline \vee \\ \hline f_3(\iota_1) & f_4(\iota_1) \\ \hline \end{array} \parallel \begin{array}{|c|} \hline g(\iota_4) \\ \hline \end{array},$$

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

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

and

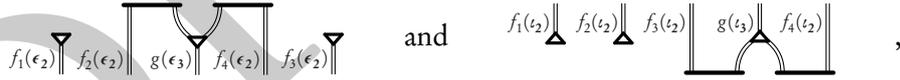
$$\Psi_{f,i} = \frac{\left( \left[ \frac{a_i^{f_1(\phi)}}{t} \vee \frac{a_i^{f_2(\phi)}}{t} \right] \wedge [a_i^{f_3(\phi)} \vee a_i^{f_4(\phi)}] \right) \wedge \frac{\bar{a}_i^{g(\psi)}}{\bar{a}_i \wedge \bar{a}_i}}{\frac{s}{\frac{[a_i^{f_3(\phi)} \vee a_i^{f_4(\phi)}] \wedge \bar{a}_i}{a_i^{f_3(\phi)}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

which allow us to build

$$\Psi_t = \left( \frac{t}{[a_1^\phi \vee \bar{a}_1^\psi] \sigma} \wedge \dots \wedge \frac{t}{[a_n^\phi \vee \bar{a}_n^\psi] \sigma} \right) \quad \text{and}$$

$$\Psi_f = \left[ \frac{(a_n^\phi \wedge \bar{a}_n^\psi) \sigma}{\Psi_{f,n} \parallel f} \vee \dots \vee \frac{(a_1^\phi \wedge \bar{a}_1^\psi) \sigma}{\Psi_{f,1} \parallel f} \right],$$

with flows



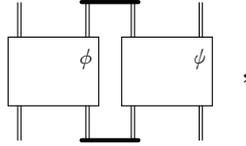
respectively. Combining these derivations we can build

$$\Psi = \frac{\frac{\alpha}{\Psi_t \wedge \Psi_T \parallel} \left( \left[ a_1^\phi \vee \bar{a}_1^\psi \right] \wedge \dots \wedge \left[ a_n^\phi \vee \bar{a}_n^\psi \right] \wedge \alpha \right) \sigma}{\frac{\Phi' \sigma \parallel}{\Psi_\perp \vee \Psi_f \parallel} \left[ \beta \vee \left( a_n^\phi \wedge \bar{a}_n^\psi \right) \vee \dots \vee \left( a_1^\phi \wedge \bar{a}_1^\psi \right) \right] \sigma} \beta$$

with the desired flow.

We know that the size of  $\Phi'\sigma$  depends at most polynomially on the size of  $\Phi$  by Theorem 4.2.3 on page 36 and Proposition 4.1.8 on page 35, and it is straightforward to observe that the sizes of  $\Psi_{\vdash}$ ,  $\Psi_{\top}$ ,  $\Psi_{\vdash}$  and  $\Psi_{\perp}$  depend at most linearly on the size of  $\Phi$ , so the size of  $\Psi$  depends at most polynomially on the size of  $\Phi$ .  $\square$

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



such that all the edges in  $\phi$  are mapped to from occurrences of  $a_1, \dots, a_n$  and no edges in  $\psi$  are mapped to from occurrences of  $a_1, \dots, a_n$ . We then define  $\text{Si}(\Phi, a_1, \dots, a_n)$  to be such that  $\Phi \rightarrow_{\text{sf}} \text{Si}(\Phi, a_1, \dots, a_n)$ , where  $\phi$  and  $\psi$  are the flows, by the same names, shown in Definition 6.1.1 on page 46.

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

1.  $\text{Si}(\Phi, a_1, \dots, a_n)$  is in simple form with respect to  $a_1, \dots, a_n$ ;
2. for any atom  $b$ , if  $\Phi$  is weakly streamlined with respect to  $b$ , then  $\text{Si}(\Phi, a_1, \dots, a_n)$  is weakly streamlined with respect to  $b$ ; and
3. the size of  $\text{Si}(\Phi, a_1, \dots, a_n)$  depends at most polynomially on the size of  $\Phi$ .

*Proof.* In the following we refer to the figure in Definition 6.1.1 on page 46:

- by case (1) of Definition 5.0.1 on page 39;
- 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  $\text{Si}(\Phi, a_1, \dots, a_n)$  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  $\Phi$  whose edges are mapped to from occurrences of  $b$ ; and
- by Theorem 6.1.3 on page 46.

$\square$

## 6.2 Isolated Subflow Removal

Given a derivation  $\Phi$  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  $\Phi$ , which is weakly streamlined with respect to  $a$ .

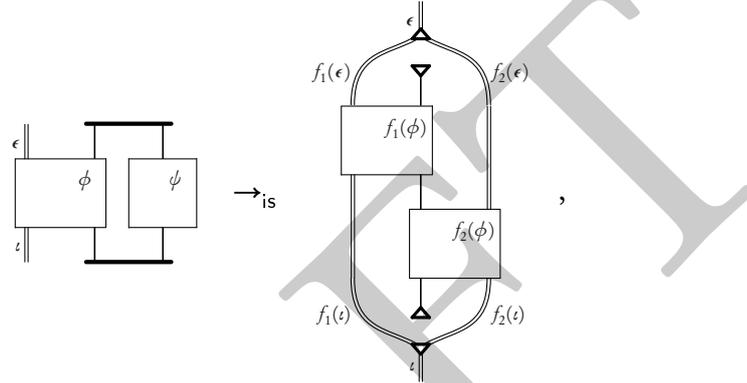
**T:** occurrences of

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.

**T:** connected flow  $\rightarrow$  connected component

**Definition 6.2.1.** We define the reduction  $\rightarrow_{is}$  (where  $is$  stands for *isolated subflow*) as follows, for any flow  $\phi$  and any connected component  $\psi$  that does not contain identity or cut vertices:



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

**T:** Corrected remark (cf. change to Definition 6.3.1 on page 55).

*Remark 6.2.2.* The condition on the flow  $\psi$  in Definition 6.2.1 ensures that all the edges in  $\psi$  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  $v_{ai\downarrow}$  to  $v_{ai\uparrow}$ .

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

*Proof.* Let  $\Phi$  be a derivation with flow  $\phi'$ , such that  $\phi' \rightarrow_{is} \psi'$ . We show that there exists a derivation  $\Psi$  with flow  $\psi'$  and with the same premiss and conclusion as  $\Phi$ . In the following, we refer to the figure in Definition 6.2.1.

Since  $\psi$  is connected, we assume, by Convention 4.2.2 on page 36, that the following derivation is an ai-decomposed form of  $\Phi$ :

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

for some atom  $a$  and formulae  $\alpha$  and  $\beta$ .

We obtain the two derivations  $\Phi_t$  and  $\Phi_f$  from  $\Phi'$  as follows:

$$\Phi_t = \frac{[t \vee \bar{a}] \wedge \alpha}{\beta \vee \bar{a}} \quad \text{and} \quad \Phi_f = \frac{\bar{a} \wedge \alpha}{\beta \vee (f \wedge \bar{a})} .$$

Since  $\psi$  is connected and contains no identity or cut vertices, the mapping from all the occurrences  $a^\psi$  to edges of  $\psi$  is surjective. Hence, we know that both derivation  $\Phi_t$  and  $\Phi_f$  have a flow isomorphic to  $\phi$ . We combine  $\Phi_t$  and  $\Phi_f$  to get the desired derivation  $\Psi$  with flow  $\psi'$  and the same premiss and conclusion as  $\Phi$ :

$$\Psi = \frac{\text{c}\uparrow \frac{\alpha}{\left( \frac{f}{t \vee \frac{\bar{a}}{\Phi_t}} \right) \wedge \alpha} \quad \wedge \quad \alpha}{\beta \vee \bar{a}} \quad \text{and} \quad \frac{\bar{a} \wedge \alpha}{\beta \vee \left( f \wedge \frac{\bar{a}}{t} \right)} \quad \text{c}\downarrow}{\beta} .$$

We know that the size of  $\Phi_t$  and the size of  $\Phi_f$  depend polynomially on the size of  $\Phi$  by Theorem 4.2.3 on page 36 and Proposition 4.1.8 on page 35, and that the size of  $\Psi$  depends at most quadratically on the size of  $\alpha$  and  $\beta$  by Lemma 2.3.12 on page 18, so the size of  $\Psi$  depends polynomially on the size of  $\Phi$ .  $\square$

**T:** Added explanation:

We now show the basic properties of  $\rightarrow_{is}$ . 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  $\perp$  vertices.

**T:** Exchanged  $\top$  and  $\perp$  in item 2, and clarified statement.

**T:** Changed from itemize to enumerate to make it a bit clearer.

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

1.  $v = f_1(v')$  or  $v = f_2(v')$ ;
2. if there is a path from  $v$  to  $\perp$  (resp.,  $\top$ ), then there is a path from  $v'$  to  $\perp$  (resp.,  $\top$ ); and

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

*Proof.* We consider each case separately:

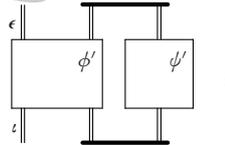
1. by definition;
2. any path from  $v$  to  $\perp$  (resp.,  $\top$ ) must contain an edge  $\epsilon$ , such that, for some lower (resp., upper) edge  $\epsilon'$  of  $\phi$ ,  $f_1(\epsilon') = \epsilon$  or  $f_2(\epsilon') = \epsilon$ . Hence, there is a path from  $v'$  to  $\perp$  (resp.,  $\top$ ); 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
  - $v = f_1(v')$  and  $\hat{v} = f_2(\hat{v}')$  (resp.,  $v = f_2(v')$  and  $\hat{v} = f_1(\hat{v}')$ ), then there is a path from  $v'$  to  $v_{ai\uparrow}$  (resp.,  $v_{ai\downarrow}$ ).

□

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

$$\left( \frac{t}{a^{\psi'} \vee \bar{a}} \wedge \dots \wedge \frac{t}{a^{\psi'} \vee \bar{a}} \wedge \alpha \right) \stackrel{\Phi'}{\parallel} \left[ \beta \vee \frac{a^{\psi'} \wedge \bar{a}}{f} \vee \dots \vee \frac{a^{\psi'} \wedge \bar{a}}{f} \right],$$

with flow

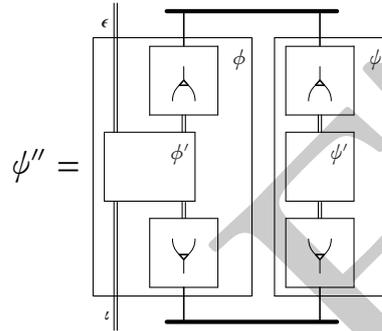


where  $\psi'$  is the juxtaposition of all the isolated subflows mapped to from occurrences of  $a$  in

$\Phi$ . Consider the derivation

$$\Psi = \left( \begin{array}{c} t \\ a \vee \bar{a} \\ \parallel \{c\} \\ [a \vee \bar{a}] \wedge \dots \wedge [a \vee \bar{a}] \\ \Phi \\ (a \wedge \bar{a}) \vee \dots \vee (a \wedge \bar{a}) \\ \beta \vee \parallel \{c\} \\ a \wedge \bar{a} \\ f \end{array} \right) \wedge \alpha$$

with flow



We then define  $\text{ISR}(\Phi, a)$  to be such that  $\Psi \rightarrow_{\text{is}} \text{ISR}(\Phi, a)$ , where  $\phi$  and  $\psi$  are the flows, by the same names, shown in Definition 6.2.1 on page 50.

**T:** Changed itemize to enumerate.

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

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

**T:** Fixed typo in second case:

*Proof.* If  $\Phi$  is weakly streamlined with respect to  $a$ , the result is trivial. Assume  $\Phi$  is not weakly streamlined with respect to  $a$ , and let  $\phi$ ,  $\psi$ ,  $\phi'$ ,  $\psi'$  and  $\psi''$  be the flows given in Definition 6.2.5 on the previous page, then

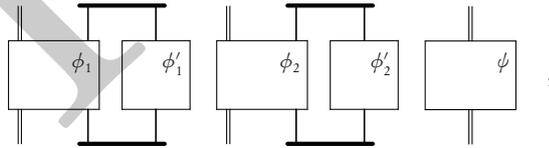
1. by definition there is no path in  $\phi$  from an interaction to a cut vertex whose edges are mapped to from instances of  $a$ . By Lemma 6.2.4 on page 51, we know that if there is a path from an interaction to a cut vertex in the flow of  $\text{ISR}(\Phi, a)$  whose edges are mapped to from instances of  $a$ , then there must be a path from an interaction to a cut vertex in  $\phi$  whose edges are mapped to from instances  $a$ . Hence, the statement follows by contradiction;
2. (a) if the flow of  $\text{ISR}(\Phi, a)$  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 51, there is a path from an interaction vertex to a cut vertex in  $\phi$ , so also in  $\phi'$ , 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  $\text{ISR}(\Phi, a)$  such that there is a path from  $v$  to  $\hat{v}$  and a path from  $v$  to  $\perp$  (resp.,  $\top$ ), both of whose edges are mapped to from instances of  $b$ , then, by Lemma 6.2.4 on page 51, there is an interaction (resp., cut) vertex  $v'$  and a cut (resp., interaction) vertex  $\hat{v}'$  in  $\phi$  such that there is a path from  $v$  to  $\hat{v}$  and a path from  $v$  to  $\perp$  (resp.,  $\top$ ), both of whose edges are mapped to from instances of  $b$ . Furthermore, since we can assume that  $b$  is not  $a$  or  $\bar{a}$ ,  $\phi$  restricted to  $b$  equals  $\phi'$  restricted to  $b$ . Hence, the statement follows by contradiction.
3. the statement follows by Theorem 6.2.3 on page 50.

□

**T: Added example:**

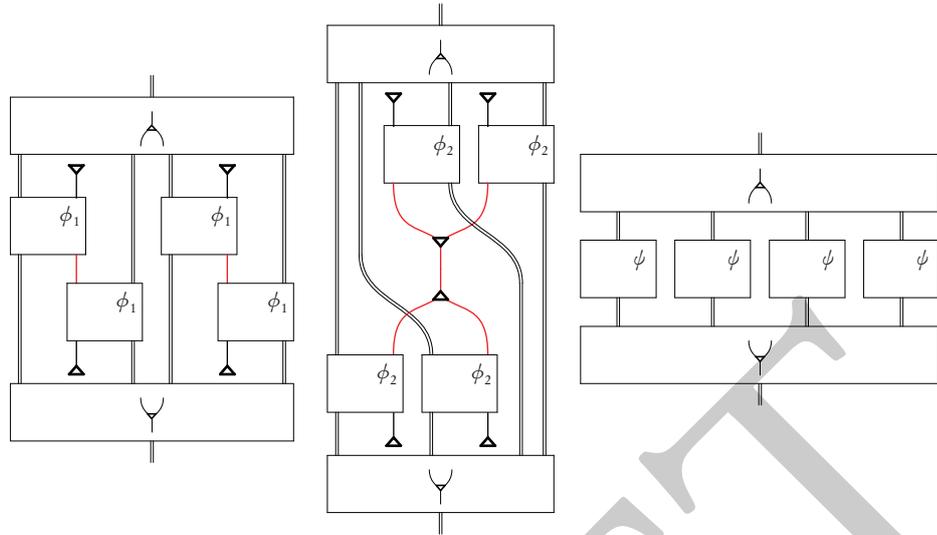
We now give an example of an application of ISR. In particular we want to show its inherent non-confluency.

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



and where all the edges in  $\phi_1$  (resp.,  $\phi'_1$ ) are mapped to from  $a_1$  (resp.,  $\bar{a}_1$ ) and all the edges in  $\phi_2$  (resp.,  $\phi'_2$ ) are mapped to from  $a_2$  (resp.,  $\bar{a}_2$ ), and there are no edges in  $\psi$  that are mapped to from  $a_1$  or  $a_2$ , then the flow associated with  $\text{ISR}(\text{ISR}(\Phi, a_1), a_2)$  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  $\phi_1$  and the subflow containing  $\phi_2$ . Note that, in order to improve readability, we have removed a contraction and a cocontraction vertex from the subflow containing  $\phi_2$ , by using *weakening reductions*. Weakening reductions are defined in Definition 7.0.8 on page 79.

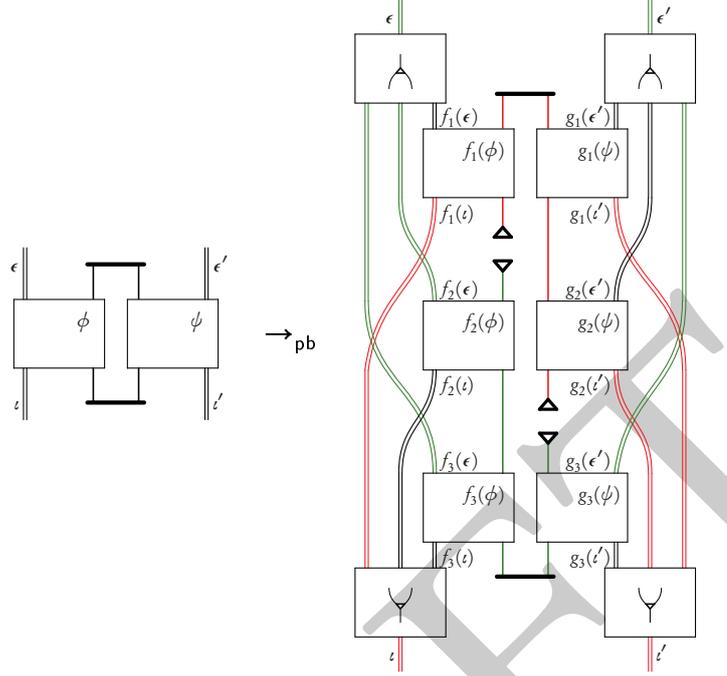
### 6.3 Path Breaker

Given a derivation  $\Phi$  and an atom  $a$ , the operator, PB, defined in this section produces a derivation with the same premiss and conclusion as  $\Phi$ , 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.

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

**Definition 6.3.1.** We define the reduction  $\rightarrow_{\text{pb}}$  (where pb stands for *path breaker*) as follows,

for any two flows  $\phi$  and  $\psi$ :



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

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

*Proof.* Let  $\Phi$  be a derivation with flow  $\phi'$ , such that  $\phi' \rightarrow_{pb} \psi'$ . We show that there exists a derivation  $\Psi$  with flow  $\psi'$  and with the same premiss and conclusion as  $\Phi$ . In the following, we refer to the figure in Definition 6.3.1 on the previous page.

Since the evidenced interaction and cut vertices belong to the same connected component, we assume, by Convention 4.2.2 on page 36, that the following derivation is an ai-decomposed form of  $\Phi$ :

$$\left( \frac{t}{a^\phi \vee \bar{a}^\psi} \wedge \alpha \right) \begin{array}{c} \Phi \\ \parallel \\ \left[ \beta \vee \frac{a^\phi \wedge \bar{a}^\psi}{f} \right] \end{array} ,$$

for some atom  $a$  and formulae  $\alpha$  and  $\beta$ .

We combine three copies of  $\Phi'$  to obtain the desired derivation  $\Psi$  with flow  $\psi'$  and the same premiss and conclusion as  $\Phi$ :

$$\begin{array}{c}
\text{c}\downarrow \frac{\alpha}{\left( \frac{t}{af_1(\phi) \vee \bar{a}g_1(\psi)} \wedge \alpha \right)} \\
\left[ \beta \vee \left( \frac{af_1(\phi)}{t} \wedge \bar{a}g_1(\psi) \right) \right] \wedge \text{c}\uparrow \frac{\alpha}{\alpha \wedge \alpha} \\
\hline
\text{s} \left[ \beta \vee \left( \left[ \frac{f}{af_2(\phi)} \vee \bar{a}g_2(\psi) \right] \wedge \alpha \right) \right] \wedge \alpha \\
\left[ \beta \vee \left( af_2(\phi) \wedge \frac{\bar{a}g_2(\psi)}{t} \right) \right] \\
\hline
\text{s} \left( \left[ af_3(\phi) \vee \frac{f}{\bar{a}g_3(\psi)} \right] \wedge \alpha \right) \\
\text{c}\downarrow \frac{\beta \vee \beta}{\beta} \vee \left[ \beta \vee \frac{af_3(\phi) \wedge \bar{a}g_3(\psi)}{f} \right] \\
\hline
\text{c}\downarrow \beta
\end{array}$$

We know that the size of  $\Phi'$  depends at most cubically on the size of  $\Phi$  by Theorem 4.2.3 on page 36, and that the size of  $\Psi$  depends at most quadratically on the size of  $\alpha$  and  $\beta$  by Lemma 2.3.12 on page 18, so  $\Psi$  depends polynomially on the size of  $\Phi$ .  $\square$

**T:** Added explanation:

We now show the basic properties of  $\rightarrow_{\text{pb}}$ . 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  $\perp$  vertices, and that it breaks all the paths between the evidenced interaction and cut vertices.

**T:** Changed from itemize to enumerate.

**T:** Added one case and its proof, which used to be in the later Proposition.

**Lemma 6.3.3.** *In the following we refer to the names given in Definition 6.3.1 on page 55. Given two flows  $\phi$  and  $\psi$ , such that  $\phi \rightarrow_{\text{pb}} \psi$ , then, given an interaction (resp., cut) vertex  $v$  in  $\psi$ , there is an interaction (resp., cut) vertex  $v'$  in  $\phi$ , such that*

1. for some  $1 \leq i \leq 3$ ,  $v = f_i(v')$  or  $v = g_i(v')$ , or  $v = v_{\text{ai}\downarrow}$  and  $v' = v'_{\text{ai}\downarrow}$  (resp.,  $v = v_{\text{ai}\uparrow}$  and  $v' = v'_{\text{ai}\uparrow}$ );
2. if there is a path from  $v$  to  $\perp$  (resp.,  $\top$ ) in  $\psi$ , then there is a path from  $v'$  to  $\perp$  (resp.,  $\top$ ) in  $\phi$ ;

3. if there is a cut (resp., interaction) vertex  $\hat{v}$  in  $\psi$ , such that there is a path from  $v$  to  $\hat{v}$  in  $\psi$ , then there is a cut (resp., interaction) vertex  $\hat{v}'$  in  $\phi$ , such that, for some  $1 \leq i \leq 3$ ,  $\hat{v} = f_i(\hat{v}')$  or  $\hat{v} = g_i(\hat{v}')$ , or  $\hat{v} = v_{\text{ai}\uparrow}$  and  $\hat{v}' = v'_{\text{ai}\uparrow}$  (resp.,  $\hat{v} = v_{\text{ai}\downarrow}$  and  $\hat{v}' = v'_{\text{ai}\downarrow}$ ); and there is a path from  $v'$  to  $\hat{v}'$  in  $\phi$ ; and
4. there is no path from  $v_{\text{ai}\downarrow}$  to  $v_{\text{ai}\uparrow}$ .

*Proof.* We consider each case separately:

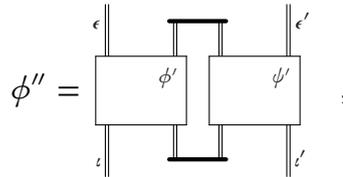
1. by definition;
2. any path from  $v$  to  $\perp$  (resp.,  $\top$ ) in  $\psi$  must contain an edge  $\epsilon$ , such that, for some lower (resp., upper) edge  $\epsilon'$  of  $\phi$  and some  $1 \leq i \leq 3$ ,  $f_i(\epsilon') = \epsilon$  or  $g_i(\epsilon') = \epsilon$ . Hence, there is a path from  $v'$  to  $\perp$  (resp.,  $\top$ ) in  $\phi$ ;
3. we have to consider two cases:
  - (a) for some  $1 \leq i \leq 3$ ,  $v = f_i(v')$  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  $\phi$ , 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  $v'_{\text{ai}\uparrow}$  (resp.,  $v'_{\text{ai}\downarrow}$ ) in  $\phi$ ; and
4. in Definition 6.3.1 we have colored the edges that might occur in paths from  $v_{\text{ai}\downarrow}$  in red and paths that might occur in path to  $v_{\text{ai}\uparrow}$  in green. Since the red and the green edges never coincide, there can be no paths from  $v_{\text{ai}\downarrow}$  to  $v_{\text{ai}\uparrow}$ .

□

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

$$\left( \frac{t}{a^\psi \vee \bar{a}} \wedge \dots \wedge \frac{t}{a^\psi \vee \bar{a}} \wedge \alpha \right)_{\Phi \parallel} \left[ \beta \vee \frac{a^\psi \wedge \bar{a}}{f} \vee \dots \vee \frac{a^\psi \wedge \bar{a}}{f} \right],$$

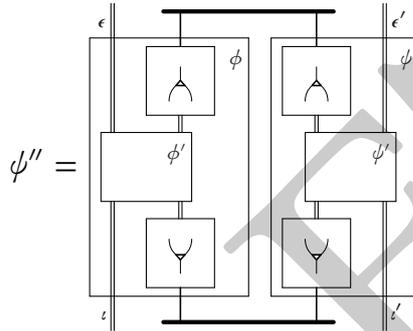
with flow



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

$$\Psi = \left( \begin{array}{c} \frac{t}{a \vee \bar{a}} \\ \parallel \{c\uparrow\} \\ [a \vee \bar{a}] \wedge \dots \wedge [a \vee \bar{a}] \end{array} \wedge \alpha \right) \Bigg|_{\Phi'} \Bigg|_{\beta \vee \left( \begin{array}{c} (a \wedge \bar{a}) \vee \dots \vee (a \wedge \bar{a}) \\ \parallel \{c\downarrow\} \\ \frac{a \wedge \bar{a}}{f} \end{array} \right)}$$

with flow



We then define  $\text{PB}(\Phi, a)$  to be such that  $\Psi \rightarrow_{\text{pb}} \text{PB}(\Phi, a)$ , where  $\phi$  and  $\psi$  are the flows, by the same names, shown in Definition 6.3.1 on page 55.

**T:** Moved some details to the previous lemma.

**Proposition 6.3.5.** *Given an atom  $a$  and a derivation  $\Phi$ ,*

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

**T:** Fixed typo in second case:

*Proof.* If  $\Phi$  is weakly streamlined with respect to both  $a$  and  $\bar{a}$ , the result is trivial. Assume  $\Phi$  is not weakly streamlined with respect to both  $a$  and  $\bar{a}$ , and let  $\phi, \psi, \phi', \psi', \phi''$  and  $\psi''$  be the flows given in Definition 6.3.4 on the previous page and let  $v_{\text{ai}\downarrow}$  (resp.,  $v_{\text{ai}\uparrow}$ ) be the evidenced interaction (resp., cut) vertex in  $\psi''$ , 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  $v_{\text{ai}\downarrow}$  (resp.,  $v_{\text{ai}\uparrow}$ ). The statement then follows by Lemma 6.3.3 on page 57;

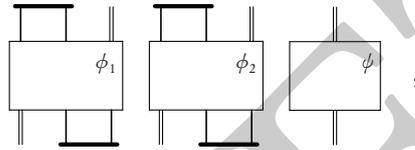
2. if the flow of  $\text{PB}(\Phi, a)$  contains a path from an interaction vertex to a cut vertex whose edges are mapped to from instances of  $b$ , then, by Lemma 6.3.3, there is a path from an interaction to a cut vertex in  $\phi$  or  $\psi$ , so also in  $\phi'$  or  $\psi'$ , whose edges are mapped to from instances of  $b$ . Hence, the statement follows by contradiction; and
3. the statement follows by Theorem 6.3.2 on page 56.

□

**T: Added informal text:**

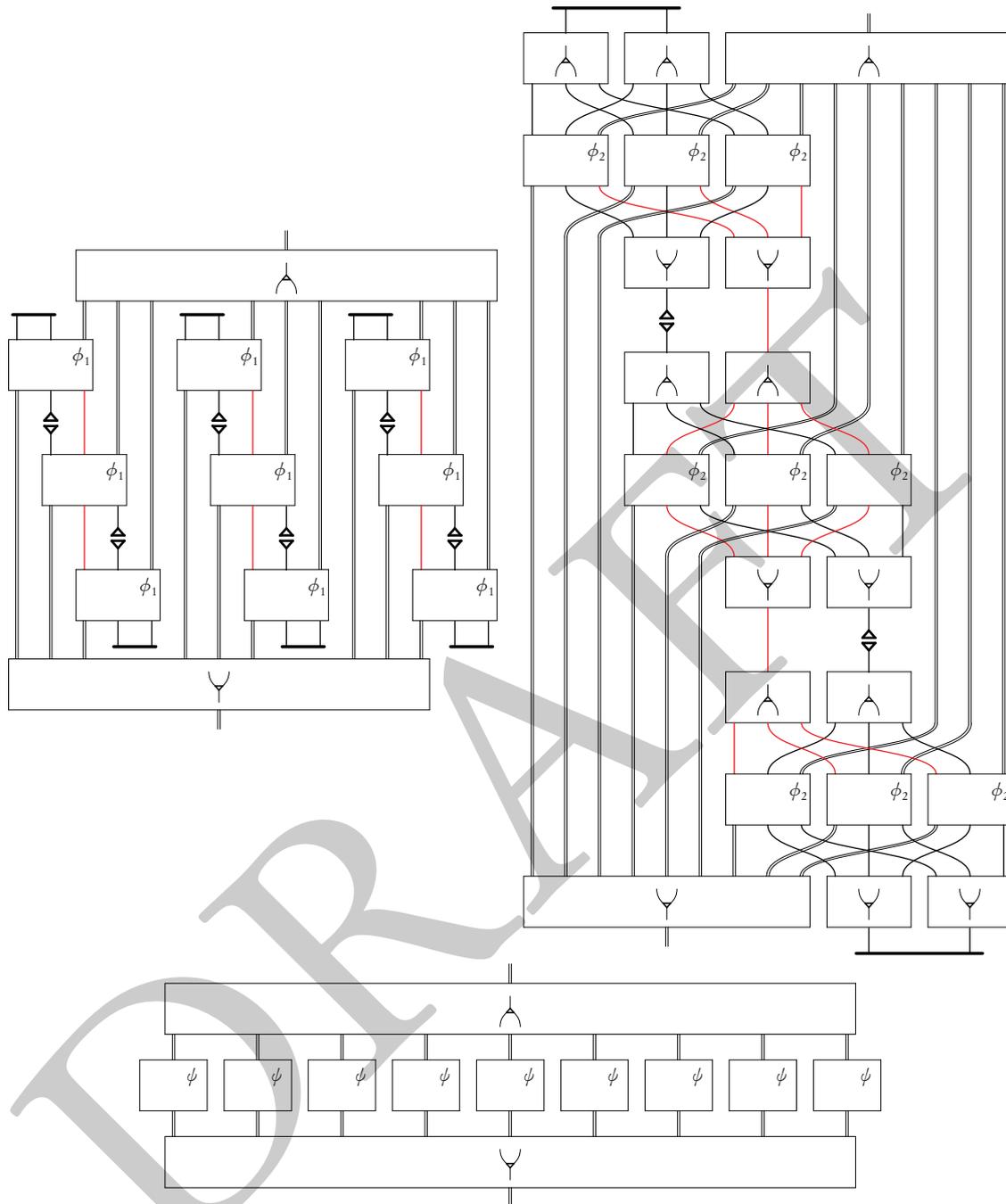
We now give an example of an application of PB. In particular we want to show its inherent non-confluency.

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



and where all the edges in  $\phi_1$  are mapped to from  $a_1$  and  $\bar{a}_1$ , and all the edges in  $\phi_2$  are mapped to from  $a_2$  and  $\bar{a}_2$ , and there are no edges in  $\psi$  that are mapped to from  $a_1$  or  $a_2$ , then the flow associated with  $\text{PB}(\text{PB}(\Phi, a_1), a_2)$  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  $\phi_1$  and the subflows containing  $\phi_2$ .

## 6.4 Multiple Isolated Subflows Removal

With the operator ISR we can produce weakly streamlined 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,  $\text{MISR}_n$ , 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  $\text{MISR}_n$ .

**T: Added explanation:**

The operator is defined in terms of the following flow reduction. Unlike the flow reductions of the preceding sections, we here present 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,  $\eta_k$ , which are used as a sort of sharing mechanism. We are at this stage not able to describe the flows  $\eta_k$  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 64.

In Subsection 6.4.1 on page 68, we present one possible combination of valid parameters, which yields quasipolynomial 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

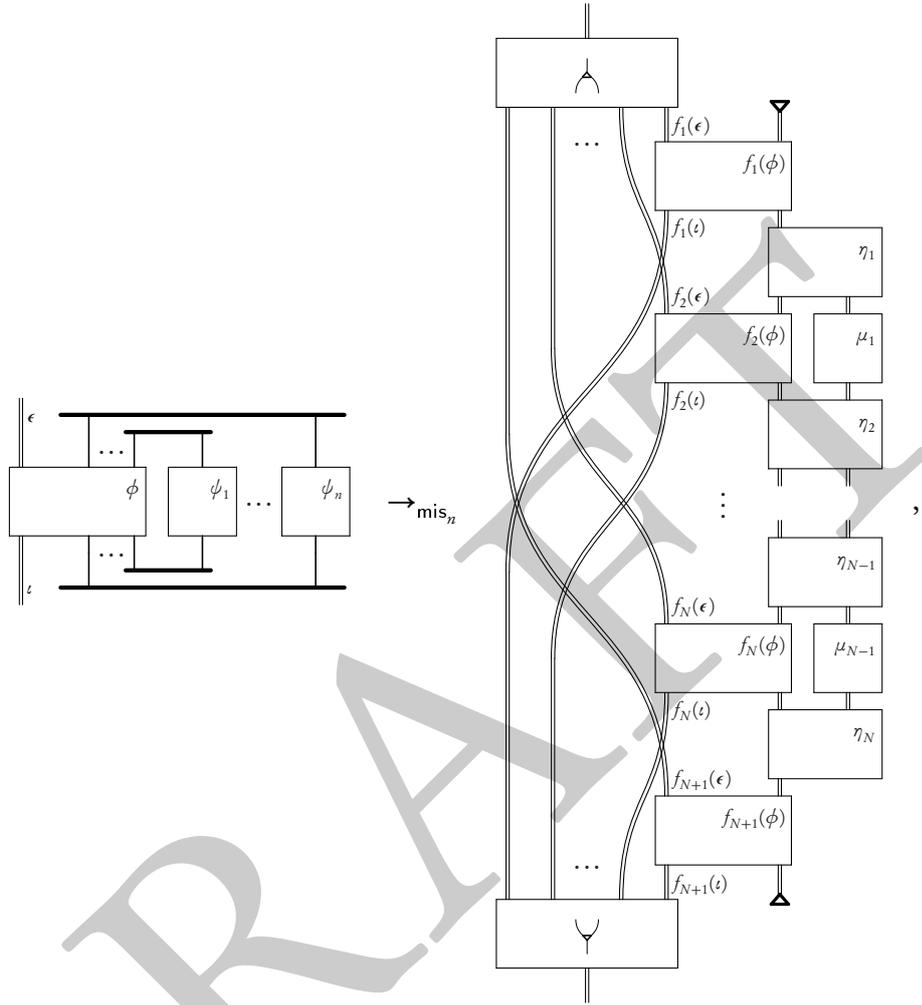
- atoms  $a_1, \dots, a_n$ ;
- an  $N > 0$ ;
- for  $0 \leq k \leq N$ , formulae  $\gamma_{k,1}, \dots, \gamma_{k,n}$ , such that
  - $\gamma_{0,1} = \dots = \gamma_{0,n} = \mathbf{t}$ , and
  - $\gamma_{N,1} = \dots = \gamma_{N,n} = \mathbf{f}$ ; and
- for  $1 \leq k \leq N$ , a derivation

$$\Gamma_k = \frac{(a_1 \wedge \gamma_{k-1,1}) \vee \dots \vee (a_n \wedge \gamma_{k-1,n})}{\|\text{SKS} \setminus \{\text{ai}\downarrow, \text{ai}\uparrow\}\}}, \quad [a_1 \vee \gamma_{k,1}] \wedge \dots \wedge [a_n \vee \gamma_{k,n}]$$

let, for  $1 \leq k \leq N$ ,  $\eta_k$  be the flow of  $\Gamma_k$ , and let

$$\mu_k = \begin{array}{c} \parallel \\ \parallel \\ \boxed{f_{1,1}(\psi_1)} \quad \dots \quad \boxed{f_{1,l_1}(\psi_1)} \quad \dots \quad \boxed{f_{n,1}(\psi_n)} \quad \dots \quad \boxed{f_{n,l_n}(\psi_n)} \\ \parallel \\ \parallel \end{array},$$

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



where we call the evidenced interaction (resp., cut) vertices  $v_{\text{ai}\downarrow,1}, \dots, v_{\text{ai}\downarrow,n}$  (resp.,  $v_{\text{ai}\uparrow,1}, \dots, v_{\text{ai}\uparrow,n}$ ).

**T:** Added a missing 'is':

*Remark 6.4.2.* The reduction  $\rightarrow_{\text{mis}_n}$  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  $\rightarrow_{\text{mis}_n}$ .

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

**T:** connected flow -> connected component

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

*Proof.* Let  $\Phi$  be a derivation with flow  $\phi'$ , such that  $\phi' \rightarrow_{\text{mis}_n} \psi'$ . We show that there exists a derivation  $\Psi$  with flow  $\psi'$  and with the same premiss and conclusion as  $\Phi$ . In the following, we refer to the figures in Definition 6.4.1 on page 62.

Since each of  $\psi_1, \dots, \psi_n$  is connected, we assume, by Convention 4.2.2 on page 36, that the following derivation is an ai-decomposed form of  $\Phi$ :

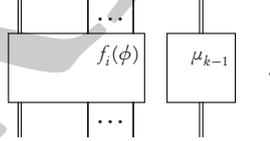
$$\frac{\frac{t}{a_1 \vee \bar{a}_1^{\psi_1}} \wedge \frac{t}{a_n \vee \bar{a}_n^{\psi_n}} \wedge \alpha}{\beta \vee \frac{a_1 \wedge \bar{a}_1^{\psi_1}}{f} \vee \frac{a_n \wedge \bar{a}_n^{\psi_n}}{f}}, \quad \Phi' \parallel$$

for some atoms  $a_1, \dots, a_n$  (that, without loss of generality, we assume coincide with the atoms given in Definition 6.4.1 on page 62) and formulae  $\alpha$  and  $\beta$ .

For every  $0 \leq k \leq N$ , we obtain the derivation  $\Phi_k$  from  $\Phi'$  as follows:

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

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



We combine  $\Phi_0, \dots, \Phi_N, \Gamma_1, \dots, \Gamma_N$  to get the desired derivation  $\Psi$  with flow  $\psi'$  and the same

premiss and conclusion as  $\Phi$ :

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

Since  $\max\{|\gamma_{0,1}|, \dots, |\gamma_{N,n}|\}$  is less than or equal to  $\max\{|\Gamma_1|, \dots, |\Gamma_N|\}$ , we know that the size of  $\Phi_0, \dots, \Phi_N$  depend at most cubically on the size of  $\Phi$  and at most quadratically on the size of  $\max\{|\Gamma_1|, \dots, |\Gamma_N|\}$  by Theorem 4.2.3 on page 36 and Proposition 4.1.8 on page 35, and that the size of  $\Psi$  depends at most cubically on the size of  $\alpha$  and  $\beta$  by Lemma 2.3.12 on page 18, so the size of  $\Psi$  depends linearly on  $N$ , polynomially on the size of  $\Phi$  and at most polynomially on the size of  $\max\{|\Gamma_1|, \dots, |\Gamma_N|\}$ .  $\square$

**T:** Added explanation:

We now show the basic properties of  $\rightarrow_{\text{mis}}$ . 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  $\perp$  vertices.

**T:** Changed from itemize to enumerate.

**T:** Exchanged  $\top$  and  $\perp$  in item 2, and clarified statement.

**Lemma 6.4.5.** *In the following we refer to the names given in Definition 6.4.1 on page 62. Given two flows  $\phi$  and  $\psi$  and an  $n > 0$ , such that  $\phi \rightarrow_{\text{mis}_n} \psi$  then, given an interaction (resp., cut) vertex  $v$  in  $\psi$ , there is an interaction (resp., cut) vertex  $v'$  in  $\phi$ , such that*

1. for some  $1 \leq i \leq N + 1$ ,  $\nu = f_i(\nu')$ ;
2. if there is a path from  $\nu$  to  $\perp$  (resp.,  $\top$ ), then there is a path from  $\nu'$  to  $\perp$  (resp.,  $\top$ ); and
3. if there is a cut (resp., interaction) vertex  $\hat{\nu}$  in  $\psi$ , such that there is a path from  $\nu$  to  $\hat{\nu}$ , then there is a cut (resp., interaction) vertex  $\hat{\nu}'$  in  $\phi$ , such that, for some  $1 \leq i \leq N + 1$ ,  $\hat{\nu} = f_i(\hat{\nu}')$ , or, for some  $1 \leq i \leq n$ ,  $\hat{\nu}' = \nu_{\text{ai}\uparrow, i}$  (resp.,  $\hat{\nu}' = \nu_{\text{ai}\downarrow, i}$ ); and there is a path from  $\nu'$  to  $\hat{\nu}'$ .

*Proof.* We consider each case separately:

1. the statement follows by definition;
2. any path from  $\nu$  to  $\perp$  (resp.,  $\top$ ) must contain an edge  $\epsilon$ , such that, for some lower (resp., upper) edge  $\epsilon'$  of  $\phi$  and some  $1 \leq i \leq N + 1$ ,  $f_i(\epsilon') = \epsilon$ . Hence, there is a path from  $\nu'$  to  $\perp$  (resp.,  $\top$ ); and
3. we have to consider two cases:
  - (a) for some  $1 \leq i \leq N + 1$ ,  $\nu = f_i(\nu')$  and  $\hat{\nu} = f_i(\hat{\nu}')$ , then there is a path from  $\nu'$  to  $\hat{\nu}'$ ; or
  - (b) for some  $1 \leq i < j \leq N + 1$ ,  $\nu = f_i(\nu')$  and  $\hat{\nu} = f_j(\hat{\nu}')$  (resp.,  $\nu = f_j(\nu')$  and  $\hat{\nu} = f_i(\hat{\nu}')$ ), then, for some  $1 \leq i \leq n$ , there is a path from  $\nu'$  to  $\nu_{\text{ai}\uparrow, i}$  (resp.,  $\nu_{\text{ai}\downarrow, i}$ ).

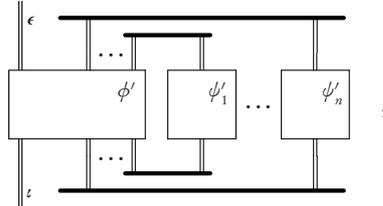
□

**Definition 6.4.6.** For every  $n > 0$ , given the atoms, formulae and derivations described in Definition 6.4.1 on page 62, the *Multiple Isolated Subflow Remover*,  $\text{MISR}_n$ , is an operator whose arguments are atoms  $a_1, \dots, a_n$  (that, without loss of generality, we assume coincide with the atoms given in Definition 6.4.1), and a derivation  $\Phi$  that is in simple form with respect to  $a_1, \dots, a_n$ . If  $n = 1$  and  $\Phi$  is weakly streamlined with respect to  $a_1$ , then  $\text{MISR}_1(\Phi, a_1) = \Phi$ ; if  $n > 1$  and, for some  $1 \leq i \leq n$ ,  $\Phi$  is weakly streamlined with respect to  $a_i$ , then  $\text{MISR}_n(\Phi, a_1, \dots, a_n) = \text{MISR}_{n-1}(\Phi, a_1, \dots, a_{i-1}, a_{i+1}, \dots, a_n)$ ; otherwise, consider the following ai-decomposed form of  $\Phi$ :

$$\left( \frac{t}{a_1^{\psi_1} \vee \bar{a}_1} \wedge \dots \wedge \frac{t}{a_1^{\psi_1} \vee \bar{a}_1} \wedge \dots \wedge \frac{t}{a_n^{\psi_n} \vee \bar{a}_n} \wedge \dots \wedge \frac{t}{a_n^{\psi_n} \vee \bar{a}_n} \wedge \alpha \right)_{\Phi \parallel} ,$$

$$\left[ \beta \vee \frac{a_n^{\psi_n} \wedge \bar{a}_n}{f} \vee \dots \vee \frac{a_n^{\psi_n} \wedge \bar{a}_n}{f} \vee \dots \vee \frac{a_1^{\psi_1} \wedge \bar{a}_1}{f} \vee \dots \vee \frac{a_1^{\psi_1} \wedge \bar{a}_1}{f} \right]$$

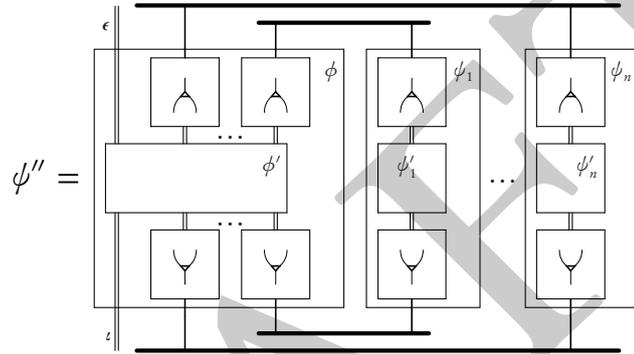
with flow



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

$$\Psi = \left( \begin{array}{c} \frac{t}{a_1 \vee \bar{a}_1} \quad \wedge \dots \wedge \quad \frac{t}{a_n \vee \bar{a}_n} \quad \wedge \quad \alpha \\ \frac{\|\{c\downarrow\}}{[a_1 \vee \bar{a}_1] \wedge \dots \wedge [a_1 \vee \bar{a}_1]} \quad \frac{\|\{c\downarrow\}}{[a_n \vee \bar{a}_n] \wedge \dots \wedge [a_n \vee \bar{a}_n]} \end{array} \right) \Phi \parallel \left[ \begin{array}{c} (a_n \wedge \bar{a}_n) \vee \dots \vee (a_n \wedge \bar{a}_n) \quad (a_1 \wedge \bar{a}_1) \vee \dots \vee (a_1 \wedge \bar{a}_1) \\ \frac{\|\{c\downarrow\}}{a_n \wedge \bar{a}_n} \quad \vee \dots \vee \quad \frac{\|\{c\downarrow\}}{a_1 \wedge \bar{a}_1} \\ \beta \vee \quad \frac{f}{f} \end{array} \right]$$

with flow



We then define  $\text{MISR}_n(\Phi, a_1, \dots, a_n)$  to be such that  $\Psi \rightarrow_{\text{mis}} \text{MISR}_n(\Phi, a_1, \dots, a_n)$ , where  $\phi, \psi_1, \dots, \psi_n$  are the flows, by the same names, shown in Definition 6.4.1 on page 62.

**T:** Changed itemize to enumerate.

**Proposition 6.4.7.** *Given the atoms, formulae and derivations described in Definition 6.4.1 on page 62, and atoms  $a_1, \dots, a_n$  and a derivation  $\Phi$  that is in simple form with respect to  $a_1, \dots, a_n$ ,*

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

**T:** Fixed typo in second case:

*Proof.* If  $\Phi$  is weakly streamlined with respect to some atom from  $a_1, \dots, a_n$ , the result follows by induction. Assume  $\Phi$  is not weakly streamlined with respect to any atom from  $a_1, \dots, a_n$ , and let  $\phi, \psi_1, \dots, \psi_n, \phi', \psi'_1, \dots, \psi'_n$  and  $\psi''$  be the flows given in Definition 6.4.6 on page 66, then

1. by definition there is no path in  $\phi$  from an interaction to a cut vertex whose edges are mapped to from instances of one of  $a_1, \dots, a_n$ . By Lemma 6.4.5 on page 65, we know that if there is a path from an interaction to a cut vertex in the flow of  $\text{MISR}_n(\Phi, a_1, \dots, a_n)$  whose edges are mapped to from instances of one of  $a_1, \dots, a_n$ , then there must be a path from an interaction to a cut vertex in  $\phi$  whose edges are mapped to from instances of one of  $a_1, \dots, a_n$ . Hence, the statement follows by contradiction;
2. (a) if the flow of  $\text{MISR}_n(\Phi, a_1, \dots, a_n)$  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  $\phi$ , so also in  $\phi'$ , 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 (reps., interaction) vertex  $\hat{v}$  in the flow of  $\text{MISR}_n(\Phi, a_1, \dots, a_n)$  such that there is a path from  $v$  to  $\hat{v}$  and a path from  $v$  to  $\perp$  (reps.,  $\top$ ), both of whose edges are mapped to from  $b$ , then, by Lemma 6.4.5, there is an interaction (reps., cut) vertex  $v'$  and a cut (reps., interaction) vertex  $\hat{v}'$  in  $\phi$  such that there is a path from  $v$  to  $\hat{v}$  and a path from  $v$  to  $\perp$  (reps.,  $\top$ ), both of whose edges are mapped to from  $b$ . Furthermore, since we can assume that  $b$  is not any of  $a_1, \dots, a_n$  or their duals,  $\phi$  restricted to  $b$  equals  $\phi'$  restricted to  $b$ . Hence, the statement follows by contradiction.
3. the statement follows by Theorem 6.4.4 on page 64.

□

*Remark 6.4.8.* Given the atoms, formulae and derivations described in Definition 6.4.1 on page 62, we can prove by induction on  $k$ , that, for every  $1 \leq i \leq n$  and every  $0 \leq k \leq N$ , the formula  $\gamma_{k,i}$  is

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

It follows by contradiction that  $N \geq n$ . Furthermore, if  $N = n$ , we know that  $\gamma_{k,i}$  is true if and only if at least  $k$  of the atoms  $a_1, \dots, a_{i-1}, a_{i+1}, \dots, a_n$  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 *Quasipolynomial Normalisation in Deep Inference via Atomic Flows and Threshold Formulae*, 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  $\Gamma$  that adhere to the condition of Definition 6.4.1 on page 62. The complexity of the  $\Gamma$  derivations dominates the complexity of the streamlined proof, and is due to the complexity of certain threshold formulae, on which the  $\Gamma$  derivations are based. The  $\Gamma$  derivations are constructed in Definition 6.4.16 on page 75; this directly leads to Theorem 6.4.17 on page 75, which states a crucial property of the  $\Gamma$  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 75. Efficiently obtaining the property stated in Theorem 6.4.17 crucially depends also on the proof system we adopt.

**T:** Used to be: In the following,  $\lfloor x \rfloor$  denotes the maximum integer  $n$  such that  $n \leq x$ .

**T:** Added last sentence after François' suggestion:

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

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

**T:** Redefined the macro outputting  $a_l^n$  to output  $(a_1, \dots, a_n)$ . It is no longer a vector of  $n-l+1$  atoms, but it is the inputs to a function of arity  $n-l+1$ .

**T:** Added explanation:

We now define a class of operators  $\theta_k^n$ , 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.

**T:** Changed away from using  $p$  and  $q$ .

**T:** No longer require the atoms to be distinct, as it the restriction might just as well be put where it is used (it will then also be clear straight away why it is needed).

**T:** Changed the definition to make it clear that the the names of the atoms are local variables (in programming parlance), so we have a wellfounded definition. The old definition was in terms of global variables (so it could not rely on an implicit renaming). This is the old definition: Consider  $n > 0$ , distinct atoms  $a_1, \dots, a_n$ , and let  $p = \lfloor n/2 \rfloor$  and  $q = n - p$ ; for  $k \geq 0$ , we define the *threshold formulae*  $\theta_k^n(a_1, \dots, a_n)$  as follows:

- for any  $n > 0$  let  $\theta_0^n(a_1, \dots, a_n) \equiv \text{t}$ ;
- for any  $n > 0$  and  $k > n$  let  $\theta_k^n(a_1, \dots, a_n) \equiv \text{f}$ ;
- $\theta_1^1(a_1) \equiv a_1$ ;
- for any  $n > 1$  and  $0 < k \leq n$  let  $\theta_k^n(a_1, \dots, a_n) \equiv \bigvee_{\substack{i+j=k \\ 0 \leq i \leq p \\ 0 \leq j \leq q}} (\theta_i^p(a_1, \dots, a_p) \cdot \theta_j^q(a_{p+1}, \dots, a_n))$ .

**Definition 6.4.9.** For every  $n > 0$  and  $k \geq 0$ , we define the operator  $\theta_k^n$  inductively as follows:

$$\theta_k^n(a_1, \dots, a_n) = \begin{cases} \text{t} & \text{if } k = 0 \\ \text{f} & \text{if } k > n \\ a_1 & \text{if } n = k = 1 \\ \bigvee_{\substack{i+j=k \\ 0 \leq i \leq \underline{n} \\ 0 \leq j \leq \bar{n}}} (\theta_i^{\underline{n}}(a_1, \dots, a_{\underline{n}}) \wedge \theta_j^{\bar{n}}(a_{\underline{n}+1}, \dots, a_n)) & \text{otherwise.} \end{cases}$$

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

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

The formulae for threshold functions adopted in [AGP02] correspond, for each choice of  $k$  and  $n$ , to  $\bigvee_{i \geq k} \theta_i^n(a_1, \dots, a_n)$ . 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  $k \geq 0$ ,  $\left| \theta_k^n(a_1, \dots, a_n) \right| \leq \left| \theta_{\underline{n}+1}^n(a_1, \dots, a_n) \right|$ .*

**T:**  $O(\cdot)$  is not defined, but it is very standard, so I leave it undefined.

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

**T:** Added a bit more explanation to the proof, and changed away from using  $p$  and  $q$ .

*Proof.* Observe that  $\left| \theta_k^n(a_1, \dots, a_n) \right| \leq \left| \theta_k^{n+1}(a_1, \dots, a_{n+1}) \right|$ . Consider:

$$\begin{aligned} \left| \theta_{\underline{n}+1}^n(a_1, \dots, a_n) \right| &= \sum_{\substack{i+j=\underline{n}+1 \\ 0 \leq i \leq \underline{n} \\ 0 \leq j \leq \bar{n}}} \left( \left| \theta_i^n(a_1, \dots, a_{\underline{n}}) \right| + \left| \theta_j^{\bar{n}}(a_{\underline{n}+1}, \dots, a_n) \right| \right) \\ &\leq \sum_{\substack{i+j=\underline{n}+1 \\ 0 \leq i, j \leq \bar{n}}} \left( \left| \theta_i^{\bar{n}}(a_1, \dots, a_{\bar{n}}) \right| + \left| \theta_j^{\bar{n}}(a_1, \dots, a_{\bar{n}}) \right| \right) \\ &\leq 2(\bar{n} + 1) \left| \theta_{(\bar{n})+1}^{\bar{n}}(a_1, \dots, a_{\bar{n}}) \right|, \end{aligned} \quad (6.1)$$

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

$$\begin{aligned} \left| \theta_{\underline{n}+1}^n(a_1, \dots, a_n) \right| &\leq 2(\bar{n} + 1) \bar{n}^{h \log \bar{n}} \\ &\leq n^2 n^{h \log(\frac{2}{3}n)} = n^{h \log n - h \log \frac{3}{2} + 2} = n^{h \log n}. \end{aligned}$$

□

**Theorem 6.4.12.** *For any  $k \geq 0$  the size of  $\theta_k^n(a_1, \dots, a_n)$  is  $n^{O(\log n)}$ .*

*Proof.* It immediately follows from Proposition 6.4.10 and Lemma 6.4.11. □

**T:** Specify that we are considering  $n$  distinct atoms and change away from using  $p$  and  $q$ .

$$\begin{aligned}
\theta_0^2(a, b) &\equiv t \text{ ,} \\
\theta_1^2(a, b) &\equiv (\theta_1^1(a) \wedge \theta_0^1(b)) \vee (\theta_0^1(a) \wedge \theta_1^1(b)) \equiv (a \wedge t) \vee (t \wedge b) \\
&= a \vee b \text{ ,} \\
\theta_2^2(a, b) &\equiv \theta_1^1(a) \wedge \theta_1^1(b) \\
&\equiv a \wedge b \text{ ,} \\
\theta_0^3(a, b, c) &\equiv t \text{ ,} \\
\theta_1^3(a, b, c) &\equiv (\theta_1^1(a) \wedge \theta_0^2(b, c)) \vee (\theta_0^1(a) \wedge \theta_1^2(b, c)) \equiv (a \wedge t) \vee (t \wedge [(b \wedge t) \vee (t \wedge c)]) \\
&= a \vee b \vee c \text{ ,} \\
\theta_2^3(a, b, c) &\equiv (\theta_1^1(a) \wedge \theta_1^2(b, c)) \vee (\theta_0^1(a) \wedge \theta_2^2(b, c)) \\
&= (a \wedge [b \vee c]) \vee (b \wedge c) \text{ ,} \\
\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 \text{ ,} \\
\theta_0^5(a, b, c, d, e) &\equiv t \text{ ,} \\
\theta_1^5(a, b, c, d, e) &\equiv (\theta_1^2(a, b) \wedge \theta_0^3(c, d, e)) \vee (\theta_0^2(a, b) \wedge \theta_1^3(c, d, e)) \\
&= a \vee b \vee c \vee d \vee e \text{ ,} \\
\theta_2^5(a, b, c, d, e) &\equiv (\theta_2^2(a, b) \wedge \theta_0^3(c, d, e)) \vee (\theta_1^2(a, b) \wedge \theta_1^3(c, d, e)) \vee (\theta_0^2(a, b) \wedge \theta_2^3(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) \text{ ,} \\
\theta_3^5(a, b, c, d, e) &\equiv (\theta_2^2(a, b) \wedge \theta_1^3(c, d, e)) \vee (\theta_1^2(a, b) \wedge \theta_2^3(c, d, e)) \vee (\theta_0^2(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) \text{ ,} \\
\theta_4^5(a, b, c, d, e) &\equiv (\theta_2^2(a, b) \wedge \theta_2^3(c, d, e)) \vee (\theta_1^2(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) \text{ ,} \\
\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 \text{ ,} \\
\theta_6^5(a, b, c, d, e) &\equiv f \text{ .}
\end{aligned}$$

Figure 6-1: Examples of threshold formulae.

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

$$w \uparrow \frac{\theta_{\underline{n}}^n(a_1, \dots, a_{\underline{n}})\{a_l/f\} \wedge \theta_{\bar{k}}^{\bar{n}}(a_{\underline{n}+1}, \dots, a_n)}{f} = w \uparrow \frac{a_1 \wedge \dots \wedge a_{l-1} \wedge a_{l+1} \wedge \dots \wedge a_{\underline{n}} \wedge \theta_{\bar{k}}^{\bar{n}}(a_{\underline{n}+1}, \dots, a_n)}{t} \wedge f.$$

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

$$w \uparrow \frac{\theta_{\bar{k}}^n(a_1, \dots, a_{\underline{n}}) \wedge \theta_{\underline{n}}^{\bar{n}}(a_{\underline{n}+1}, \dots, a_n)\{a_l/f\}}{f} = w \uparrow \frac{\theta_{\bar{k}}^n(a_1, \dots, a_{\underline{n}}) \wedge a_{\underline{n}+1} \wedge \dots \wedge a_{l-1} \wedge a_{l+1} \wedge \dots \wedge a_n}{t} \wedge f.$$

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 page 75.

**T:** Broke one case into two to avoid overfull.

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

- For  $n > 1$  and  $1 \leq l \leq n$ , we define the derivations  $\Upsilon_{k,l}^n(a_1, \dots, a_n)$  and  $\Delta_{k,l}^n(a_1, \dots, a_n)$  as follows:

$$\Upsilon_{k,l}^n(a_1, \dots, a_n) = \begin{cases} w \uparrow \frac{(\theta_{\underline{n}}^n(a_1, \dots, a_{\underline{n}})\{a_l/f\} \wedge \theta_{\bar{k}-\underline{n}}^{\bar{n}}(a_{\underline{n}+1}, \dots, a_n))}{f} & \text{if } \underline{n} \leq k \leq n \text{ and } l \leq \underline{n} \\ w \uparrow \frac{\theta_{\bar{k}-\bar{n}}^n(a_1, \dots, a_{\underline{n}}) \wedge (\theta_{\underline{n}}^{\bar{n}}(a_{\underline{n}+1}, \dots, a_n)\{a_l/f\})}{f} & \text{if } \bar{n} \leq k \leq n \text{ and } \underline{n} < l \\ f & \text{otherwise} \end{cases}$$

and

$$\Delta_{k,l}^n(a_1, \dots, a_n) = \begin{cases} w \downarrow \frac{f}{\theta_{\bar{k}}^{\bar{n}}(a_{\underline{n}+1}, \dots, a_n)} & \text{if } 0 < k \leq \bar{n} \text{ and } l \leq \underline{n} \\ w \downarrow \frac{f}{\theta_{\bar{k}}^n(a_1, \dots, a_{\underline{n}})} & \text{if } 0 < k \leq \underline{n} \text{ and } \underline{n} < l \\ f & \text{otherwise} \end{cases}.$$

- For  $k \geq 0$  and  $1 \leq l \leq n$ , we define the derivations  $\Gamma_{k,l}^n(a_1, \dots, a_n)$ , recursively on  $n$ , as follows:
  - $\Gamma_{0,1}^1(a_1) = t$ ;
  - for  $k > 0$ ,  $\Gamma_{k,1}^1(a_1) = f$ ;
  - for  $k > n$ ,  $\Gamma_{k,l}^n(a_1, \dots, a_n) = f$ ;

- for  $n > 1$ ,  $k \leq n$  and  $l \leq \underline{n}$ , let  $\Gamma_{k,l}^n(a_1, \dots, a_n)$  be

$$\bigvee_{\substack{i+j=k \\ 0 \leq i < \underline{n} \\ 0 \leq j \leq \bar{n}}} \left( \Gamma_{i,l}^{\underline{n}}(a_1, \dots, a_{\underline{n}}) \wedge \Theta_j^{\bar{n}}(a_{\underline{n}+1}, \dots, a_n) \right) \vee \Upsilon_{k,l}^n(a_1, \dots, a_n) \vee \Delta_{k+1,l}^n(a_1, \dots, a_n)$$

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

$$\bigvee_{\substack{i+j=k \\ 0 \leq i \leq \underline{n} \\ 0 \leq j < \bar{n}}} \left( \Theta_i^{\underline{n}}(a_1, \dots, a_{\underline{n}}) \wedge \Gamma_{j,l-\underline{n}}^{\bar{n}}(a_{\underline{n}+1}, \dots, a_n) \right) \vee \Upsilon_{k,l}^n(a_1, \dots, a_n) \vee \Delta_{k+1,l}^n(a_1, \dots, a_n) .$$

**T:** Removed example of  $\Gamma_{k,l}^n(a_1, \dots, a_n)$ , as it was not very helpful.

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

$$\frac{\Theta_k^n(a_1, \dots, a_n) \{a_l/f\}}{\|\{aw\downarrow, aw\uparrow\}}, \quad \Theta_{k+1}^n(a_1, \dots, a_n) \{a_l/t\}$$

and  $|\Gamma_{k,l}^n(a_1, \dots, a_n)|$  is  $n^{O(\log n)}$ .

**T:** Broke the derivation horizontally in a different way to avoid overfull.

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

$$\frac{\Theta_k^n(a_1, \dots, a_n) \{a_l/f\}}{\Gamma_{k,l}^n(a_1, \dots, a_n) \|\} \quad \Theta_{k+1}^n(a_1, \dots, a_n) \{a_l/t\} = \bigvee_{\substack{i+j=k \\ 0 \leq i < p \\ 0 \leq j \leq q}} \left( \frac{\Theta_i^p(a_1, \dots, a_p) \{a_l/f\}}{\Gamma_{i,l}^p(a_1, \dots, a_p) \|\} \wedge \Theta_j^q(a_{p+1}, \dots, a_n) \right) \vee$$

$$\vee \text{w}\uparrow \frac{(\Theta_p^p(a_1, \dots, a_p)) \{a_l/f\} \wedge \Theta_{k-p}^q(a_{p+1}, \dots, a_n)}{f} \vee \text{w}\downarrow \frac{f}{\Theta_{k+1}^q(a_{p+1}, \dots, a_n)} .$$

General (co)weakening rule instances can be replaced by atomic ones because of Lemma 2.3.11 on page 18. The size bound on  $\Gamma_{k,l}^n(a_1, \dots, a_n)$  follows from Proposition 4.1.8 on page 35 and Theorem 6.4.12 on page 72.  $\square$

**T:** Rephrased slightly to avoid overfull

**T:** Extracted a constructin with superswitches and (co)contractions into its own Lemma.

**Definition 6.4.16.** Consider  $n > 0$ , distinct atoms  $a_1, \dots, a_n$ . For  $k \geq 0$ , we define the derivation  $\Gamma_k^n(a_1, \dots, a_n)$  to be:

$$\left[ \left( \begin{array}{c} \Theta_k^n(a_1, \dots, a_n)\{a_1/f\} \\ a_1 \wedge \Gamma_{k,1}^n(a_1, \dots, a_n) \\ \Theta_{k+1}^n(a_1, \dots, a_n)\{a_1/t\} \\ \parallel \{ac\uparrow, s\} \\ \Theta_{k+1}^n(a_1, \dots, a_n) \end{array} \right) \vee \dots \vee \left( \begin{array}{c} \Theta_k^n(a_1, \dots, a_n)\{a_n/f\} \\ a_n \wedge \Gamma_{k,n}^n(a_1, \dots, a_n) \\ \Theta_{k+1}^n(a_1, \dots, a_n)\{a_n/t\} \\ \parallel \{ac\uparrow, s\} \\ \Theta_{k+1}^n(a_1, \dots, a_n) \end{array} \right) \right] \\ \parallel \{c\downarrow\} \\ \Theta_{k+1}^n(a_1, \dots, a_n) \\ \parallel \{c\uparrow\} \\ \left( \begin{array}{c} \Theta_{k+1}^n(a_1, \dots, a_n) \\ \parallel \{ac\downarrow, s\} \\ [a_1 \vee \Theta_{k+1}^n(a_1, \dots, a_n)\{a_1/f\}] \end{array} \wedge \dots \wedge \begin{array}{c} \Theta_{k+1}^n(a_1, \dots, a_n) \\ \parallel \{ac\downarrow, s\} \\ [a_n \vee \Theta_{k+1}^n(a_1, \dots, a_n)\{a_n/f\}] \end{array} \right)$$

where we use the derivations constructed in the proof of Lemma 2.3.10 on page 17.

**Theorem 6.4.17.** For any  $n > 0$  and  $k \geq 0$ , the derivation  $\Gamma_k^n(a_1, \dots, a_n)$  has shape

$$\left( \begin{array}{c} (a_1 \wedge \Theta_k^n(a_1, \dots, a_n)\{a_1/f\}) \vee \dots \vee (a_n \wedge \Theta_k^n(a_1, \dots, a_n)\{a_n/f\}) \\ \parallel \text{SKS} \setminus \{ai\downarrow, ai\uparrow\} \\ [a_1 \vee \Theta_{k+1}^n(a_1, \dots, a_n)\{a_1/f\}] \wedge \dots \wedge [a_n \vee \Theta_{k+1}^n(a_1, \dots, a_n)\{a_n/f\}] \end{array} \right),$$

and  $|\Gamma_k^n(a_1, \dots, a_n)|$  is  $n^{O(\log n)}$ .

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

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

to be special cases of  $\rightarrow_{\text{mis}_n}$  and  $\text{MISR}_n$ , respectively, such that, given atoms  $(a_1, \dots, a_n)$ ,

- $N = n$ ;
- for  $0 \leq k \leq n$  and  $1 \leq i \leq n$ ,  $\gamma_{k,i} = (\Theta_k^n(a_1, \dots, a_n))\{a_i/f\}$ ; and
- for  $1 \leq k \leq n$ ,  $\Gamma_k = \Gamma_k^n(a_1, \dots, a_n)$ .

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

*Proof.* The result follows by Theorem 6.4.4 on page 64, Definition 6.4.9 on page 70 and Theorem 6.4.17.  $\square$

**T:** Changed from itemize to enumerate.

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

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

*Proof.* The statements follow by Proposition 6.4.7 on page 67 and Theorem 6.4.17 on the preceding page.  $\square$

# Chapter 7

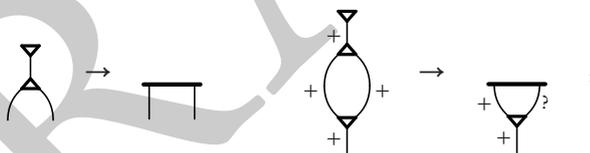
## Local Reductions

**TODO non-trivial automorphism.**

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 79, we explore some of their basic properties, in the two short Sections 7.2 on page 81 and 7.3 on page 82.

**Definition 7.0.1.** In Figure 7-1 on the following page, we define graphical expressions of the kind  $r: \phi' \rightarrow \psi'$ , where  $r$  is a name and  $\phi'$  and  $\psi'$  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  $\phi'$  to flow  $\psi'$  is a quadruple  $(\phi', \psi', f, g)$  such that:

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

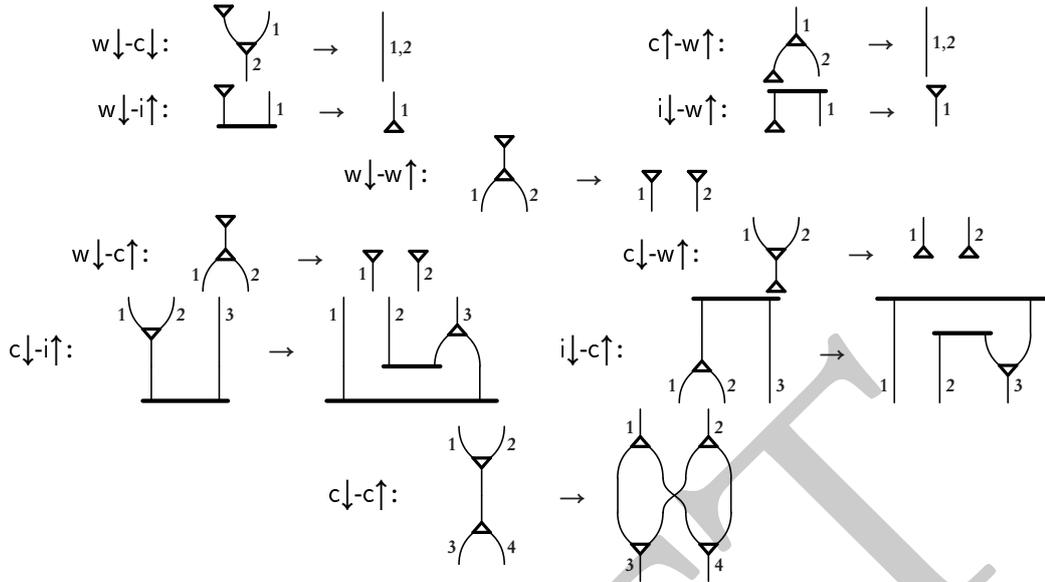


Figure 7-1: Atomic-flow reduction rules.

we define reduction rules with graphical expressions  $r : \phi' \rightarrow \psi'$ , where  $f$  and  $g$  are indicated by labelling edges. For every reduction rule  $r : \phi' \rightarrow \psi'$ , the reduction  $\rightarrow_r$  is defined, such that  $\phi \rightarrow_r \psi$  if and only if  $\phi'$  appears as a subflow in  $\phi$  and we obtain  $\psi$  by replacing  $\phi'$  with  $\psi'$  in  $\phi$ , 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  $\psi$  in  $\phi \rightarrow_r \psi$  is a proper atomic flow, if  $\phi$  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  $F = \{r_1, \dots, r_b\}$  we define  $\rightarrow_F = \rightarrow_{r_1} \cup \dots \cup \rightarrow_{r_b}$ . The reflexive transitive

closure of  $\rightarrow_F$  is denoted by  $\rightarrow_F^*$ . 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  $\phi_1 \rightarrow_F \phi_2 \rightarrow_F \dots$ , 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  $\phi$  is *normal* for flow rewriting system  $F$  if there is no flow  $\psi$  such that  $\phi \rightarrow_F \psi$ .

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

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

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

$$\{ c \downarrow - i \uparrow, i \downarrow - c \uparrow, c \downarrow - c \uparrow \} .$$

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  $\phi$ , if  $\phi \rightarrow_w^* \psi$  and  $\psi$  is normal for  $w$ , then  $\psi$  is super streamlined.*

**Proposition 7.0.12.** *Given a super-streamlined flow  $\phi$ , if  $\phi \rightarrow_c^* \psi$  and  $\psi$  is normal for  $c$ , then  $\psi$  is hyper streamlined.*

## 7.1 Soundness

**Definition 7.1.1.** A reduction rule  $r$  is *sound* if  $\rightarrow_r$  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 81.

**Theorem 7.1.2.** *The reduction rules  $w \downarrow - c \downarrow, w \downarrow - i \uparrow, w \downarrow - w \uparrow, w \downarrow - c \uparrow, c \downarrow - i \uparrow, c \downarrow - c \uparrow, c \uparrow - w \uparrow, i \downarrow - w \uparrow, c \downarrow - w \uparrow$  and  $i \downarrow - c \uparrow$  are sound.*

*Proof.* For  $r \in \{w \downarrow - c \downarrow, w \downarrow - i \uparrow, w \downarrow - w \uparrow, w \downarrow - c \uparrow, c \downarrow - i \uparrow, c \downarrow - c \uparrow\}$  and  $r: \phi' \rightarrow \psi'$  as in the left columns of Figures 7-2 on the following page and 7-3 on page 81, for every  $\phi$  and  $\psi$  such that  $\phi \rightarrow_r \psi$  and for every  $\Phi$  with flow  $\phi$ , the right columns of the tables provide reductions  $\Phi \rightarrow_r \Psi$ , where  $\Psi$  has flow  $\psi$ , as follows. If  $\Phi' \rightarrow_r \Psi'$  is the reduction provided by the table, then

$$\begin{array}{c} \alpha \\ \Psi_1 \parallel \\ \alpha' \\ \Phi = \Phi' \parallel \\ \beta' \\ \Psi_2 \parallel \\ \beta \end{array} \quad \text{and} \quad \begin{array}{c} \alpha \\ \Psi_1 \parallel \\ \alpha' \\ \Psi = \Psi' \parallel \\ \beta' \\ \Psi_2 \parallel \\ \beta \end{array} .$$



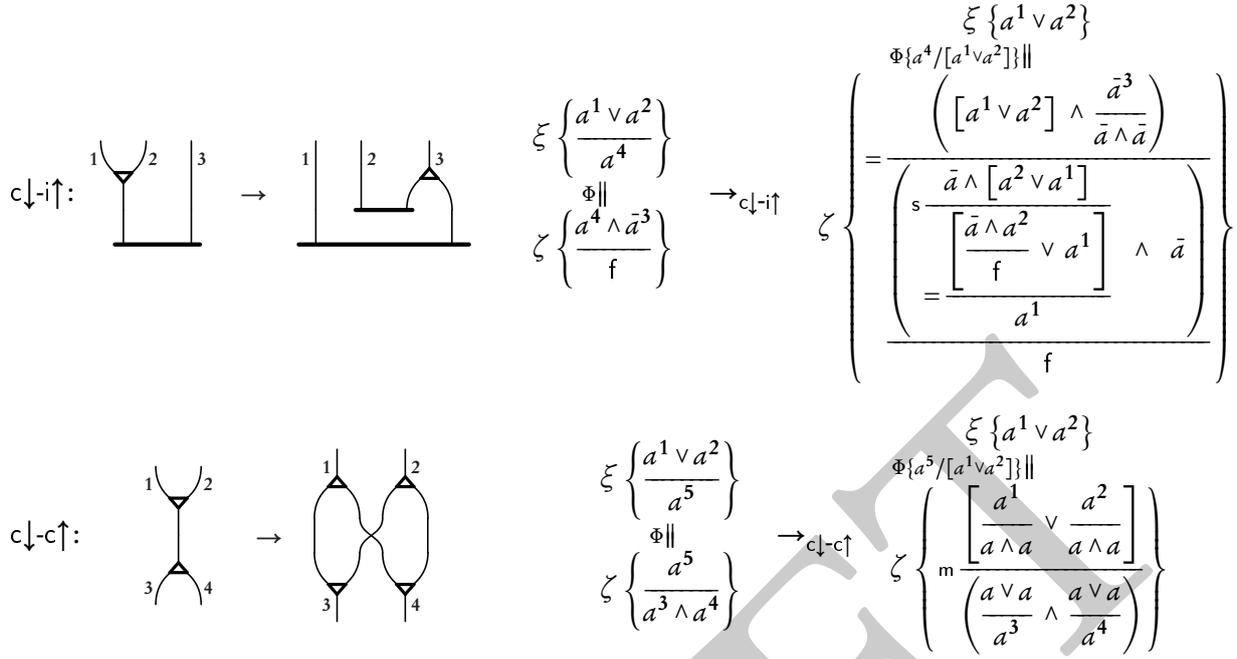


Figure 7-3: ‘Downwards’ reduction rules for contraction and their soundness.

We can deal with the remaining rules by employing dual derivations to the ones shown.  $\square$

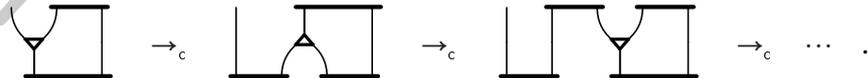
*Remark 7.1.3.* The previous soundness theorem only depends on the switch and medial rules for the reductions in Figure 7-3. Any system obtained from SKS by replacing  $s$  and  $m$  with linear rules that can derive them would support a soundness theorem like the one above, for the same reduction rules.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

*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  $\phi$  be a cycle-free flow. We associate to each contraction (resp., cocontraction) vertex  $v$  its *rank*  $r_v = \sum_{p_i \in I_v} h_i$ , where  $I_v$  is the set of all maximal ai-paths  $p_i = \epsilon_1^i, \dots, \epsilon_{h_i}^i$  from  $v$ , such that  $\epsilon_1^i$  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  $\phi$  by  $c$  decreases the sum of the ranks of the (co)contraction vertices of  $\phi$ . First note that the rank of the vertices not involved in the reduction step stays the same (see Remark 7.0.10 on page 79). 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:

$c\downarrow\text{-}i\uparrow$ : a contraction vertex  $v$  is replaced by a cocontraction vertex  $v'$ , and  $r_{v'} = r_v - n$ , where  $n > 0$  is the number of maximal ai-paths from  $v$  whose first edge is the lower edge of  $v$ ;

$i\downarrow\text{-}c\uparrow$ : this is dual to the previous case;

$c\downarrow\text{-}c\uparrow$ : 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  $r_{v_1} + r_{v_2} = r_v - n$ , where  $n > 0$  is the number of maximal ai-paths from  $v$  whose first edge is the lower edge of  $v$ ; analogously, we have  $r_{v'_1} + r_{v'_2} = r_{v'} - n'$ , where  $n' > 0$  is the number of maximal ai-paths from  $v'$  whose first edge is the upper edge of  $v'$ .

□

**Theorem 7.2.4.** *Flow rewriting system  $w \cup c$  is confluent.*

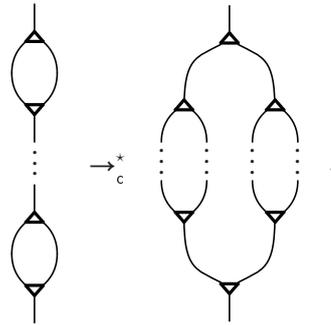
*Proof.* It is straightforward to verify the statement by checking each critical pair of  $w \cup c$ . □

### 7.3 Complexity

**Proposition 7.3.1.** *Given a derivation  $\Phi$ , there exists a derivation  $\Psi$ , such that  $\Phi \rightarrow_w^* \Psi$ ,  $\Psi$  is normal for  $w$  and the size of  $\Psi$  depends at most linearly on the size of  $\Phi$ .*

*Proof.* The number of reductions used to arrive at  $\Psi$  is bound by the number of edges in the flow of  $\Phi$ , so by the size of  $\Phi$ . Furthermore, each reductions shown in Figure 7-2 on page 80 grows the derivation by at most a constant. Hence, the size of  $\Psi$  depends at most linearly on the size of  $\Phi$ . □

*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  $\rightarrow_c^*$  (see Remark 7.0.10 on page 79). 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  $\Phi$  and distinct and pairwise non-dual atoms  $a_1, \dots, a_n$ , such that  $a_1, \dots, a_n$  and their duals are all the atoms appearing in  $\Phi$ ,*

1. *let*

$$\begin{aligned}\Phi' &= \text{ISR}(\dots \text{ISR}(\text{Si}(\Phi, a_1, \dots, a_n), a_1), \dots, a_n) \text{ and} \\ \Phi'' &= \text{ISR}(\dots \text{ISR}(\text{Si}(\Phi', \bar{a}_1, \dots, \bar{a}_n), \bar{a}_1), \dots, \bar{a}_n),\end{aligned}$$

*then*

- (a)  $\Phi''$  is weakly streamlined, and
- (b) the size of  $\Phi''$  depends at most exponentially on the size of  $\Phi$ ;

2. *let  $\Phi' = \text{PB}(\dots \text{PB}(\Phi, a_1), \dots, a_n)$ , then*

- (a)  $\Phi'$  is weakly streamlined, and
- (b) the size of  $\Phi'$  depends at most exponentially on the size of  $\Phi$ ; and

3. *let*

$$\begin{aligned}\Phi' &= \text{QMISR}(\text{Si}(\Phi, a_1, \dots, a_n), a_1, \dots, a_n) \text{ and} \\ \Phi'' &= \text{QMISR}(\text{Si}(\Phi', \bar{a}_1, \dots, \bar{a}_n), \bar{a}_1, \dots, \bar{a}_n),\end{aligned}$$

*then*

- (a)  $\Phi''$  is weakly streamlined, and
- (b) the size of  $\Phi''$  depends at most quasipolynomially on the size of  $\Phi$ .

*Proof.* The statements follow by Proposition 6.1.5 on page 49, Proposition 6.2.6 on page 53, Proposition 6.3.5 on page 59 and Proposition 6.4.20 on page 76.  $\square$

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

*Proof.* By Theorem 8.0.1 on the preceding page, Proposition 7.0.11 on page 79, Theorem 7.1.2 on page 79 and Proposition 7.3.1 on page 82.  $\square$

**Corollary 8.0.3.** *Given a derivation  $\Phi$ , there exists a hyper-streamlined derivation  $\Psi$  with the same premiss and conclusion as  $\Phi$ .*

*Proof.* By Theorem 8.0.1 on the preceding page, Proposition 7.0.11 on page 79, Proposition 7.0.12 on page 79, Theorem 7.1.2 on page 79 and Proposition 7.3.1 on page 82.  $\square$

DRAFT

# Index

- SKS, 14
- ai-decomposed form, 36
  - canonical, 37
- ai-path, 27
  - length, 27
  - maximal, 27
- atomic flow, *see* flow
- atoms
  - negation, 8
  - renaming, 8
- connected component, 29
- derivation
  - calculus of structures, 12
    - vertical composition, 12
  - functorial calculus, 8
    - horizontal composition, 9
    - vertical composition, 9
- flow, 24
  - associated with derivation, 31
  - edges, 24
  - isomorphism, 26
  - labels, 23, 24
  - lower edges, 24
  - polarity assignment, 24
  - restriction to atom, 31
  - upper edges, 24
  - vertices, 24
- flow rewriting system, 78
  - normal, 79
  - terminating, 79
- formula
  - context, 8
  - size, 8
  - ground, 8
  - instance, 8
  - size, 8
- formulae, 7
- hole, 8
- inference rule, 8
  - conclusion, 8
  - instance, 8
  - invertible, 15
  - logical, 15
  - premiss, 8
  - structural, 15
- Isolated Subflow Remover
  - operator, 52
  - reduction, 50
- Multiple Isolated Subflows Remover
  - operator, 66
  - reduction, 63
- path, 27
  - length, 27
- Path Breaker
  - operator, 58
  - reduction, 55
- Quasipolynomial Multiple Isolated Subflows Remover
  - operator, 75
  - reduction, 75
- reduction, 45
  - by rule, 78
  - liftable, 45
  - rule, 77
    - sound, 79
  - sound, 45

simple form  
  derivation  
    with respect to atom, 41  
  flow, 39  
Simplifier, 49  
simplifier  
  reduction, 46  
streamlined, 40  
  derivation, 41  
  hyper, 40  
  super, 40  
  weakly, 40  
    with respect to atom, 41  
    with respect to polarity, 40  
subflow, 28  
  isolated, 28  
substitution, 8  
system, 8  
  
threshold  
  formulae, 68, 70

DRAFT

# 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/PrComp1DI.pdf>.
- [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/bv1/bv1.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/1c1-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/sc11.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>.
- [Gir87] Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.
- [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/cmd1.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.
- [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.L0/0605054>.
- [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>.
- [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/l1s-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>.
- [Thi03] Rüdiger Thiele. Hilbert’s twenty-fourth problem. *American Mathematical Monthly*, 110:1–24, 2003.
- [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.L0/0512036>.
- [Weg87] Ingo Wegener. *The Complexity of Boolean Functions*. John Wiley & Sons Ltd and B. G. Teubner, Stuttgart, 1987.