

Universidade Nova de Lisboa
Faculdade de Ciências e Tecnologia
Departamento de Informática

Technische Universität Wien

A Proof Theory of Interpolation

Ismael Everardo Bárcenas Patiño

Master Thesis in
Computational Logic

Dissertação apresentada na Faculdade de Ciências e
Tecnologia da Universidade Nova de Lisboa para obtenção do
grau de Mestre em Lógica Computacional

supervision:
João Rasga

Lisboa
2007

Abstract

Schematic systems is a metamathematical framework used to describe deductive systems. In this Master Thesis we describe sufficient conditions for a general class of sequent calculi, presented by schematic systems, to enjoy the Craig Interpolation Theorem. These conditions are a refinement of the ones in [9] already known to be weaker than symmetricity. The proof of interpolation is based on Maehara's technique and on the idea of tracing the flow of the formulas in a deduction. Sequent systems for classical, intuitionistic and minimal first order logic are presented by schematic systems and their properties of cut-elimination and interpolation are proved. Moreover, we study the combinatorial nature of interpolation by means of structured sets. Finally, we present an analysis of the interpolant complexity (number of symbols of the interpolant) for propositional deductive systems and show its application to some fundamental questions in complexity theory.

Contents

1	Introduction	5
2	Deductive systems	7
2.1	Basic concepts	7
2.2	Schematic systems	9
2.2.1	Regular schematic systems	12
2.2.2	Classical logic	14
2.3	Restricted schematic systems	20
2.3.1	Intuitionistic and minimal logics	20
2.4	Logical graphs	22
2.5	Structured sets	25
3	Interpolation	29
3.1	Craig interpolation	29
3.1.1	Basic results	30
3.1.2	Some applications	34
3.1.3	Structured sets	35
3.1.4	Regular schematic systems	37
3.1.5	Restricted regular schematic systems	41
3.2	Interpolant complexity	48
4	Conclusion	55
4.1	Results	55
4.2	Further directions	56
	Bibliography	57

Chapter 1

Introduction

Logic can be seen as the analysis of methods of reasoning. In studying these methods, logic is interested in the form rather than the content of the argument. The systematic formalization and cataloguing of valid methods of reasoning are some of the main tasks of logicians. If the work uses mathematical techniques or if it is primarily devoted to the study of mathematical reasoning, then it may be called *mathematical logic*. Main subfields of mathematical logic are: *set theory*, *proof theory*, *model theory*, and *computability (recursion theory)*.

Formal deductions are the object of study of proof theory, which may be divided into *structural proof theory* and *interpretational proof theory*. Structural proof theory is based on a combinatorial analysis of the structure of formal deductions. Syntactical translations of one formal theory into another is the main objective of interpretational proof theory.

Some of the goals of mathematical logic are:

- provide *languages* for the precise formulation of *statements*,
- investigate *mechanisms* for finding out the *truth* or *falsity* of statements.

In Chapter 2 we describe general notions for languages and *deductive systems*. Two frameworks for deductive systems are of our interest, *schematic systems* and *structured sets*.

The notion of schematic system is an original idea of Parikh [29] whose main purpose is to serve as a general framework for formal deduction. We will present schematic systems for *classical*, *intuitionistic* and *minimal first order logics* without and with *equality*. We also present the notion of *logical graphs* [29] whose aim is to study the structure of formal deductions by means of flows of occurrences of the formulas in a deduction.

In order to understand the combinatorial nature of the *interpolation property* we will describe another general framework for deductions called structured sets [10]. Deductive systems for classical and intuitionistic first order logics are described in the context of structured sets.

We consider the interpolation property known as the *Craig interpolation theorem* [14] which states that given a deduction of $A \rightarrow B$, where A and B are formulas of classical first order logic, there are a formula C , called the interpolant, and deductions of $A \rightarrow C$ and $C \rightarrow B$ such that C contains only those predicates and free variables occurring in both A and B . Craig's interpolation theorem (published 50 years ago) has proved to be a central logical property that has been used to reveal a deep harmony between the syntax and semantics of several logics. Also, Craig's result has proven particularly useful in the development of formal methods and tools for the design or the analysis of computer systems [26]. We describe in the first section of Chapter 3, as in [40], [37] and [20], the Craig's interpolation theorem for classical first order logic, and then present the interpolation property of classical first order logic with equality as well as for intuitionistic and minimal first order logic with and without equality.

Then in Chapter 3 we describe the interpolation property of structured sets as first introduced in [10]. We present sufficient conditions for interpolation in schematic systems which are a refinement of the conditions introduced in [9]. Moreover, we also introduce sufficient conditions for interpolation in restricted schematic systems.

It is stated in [13] that there exists a deductive system for classical propositional logic in which all tautologies have deductions of polynomial size (number of symbols) if and only if $\text{NP}=\text{co-NP}$. Several deductive systems and a hierarchy with respect to their polynomial simulation have been proposed in [30] but to prove exponential lower bounds for the deductive systems of the hierarchy seems to be a hard task. Certain combinatorial principles provide good candidates for this task but it is equally hard to show whether or not such principles have deductions of polynomial size. This is because all short deductions for such tautologies must satisfy some non-trivial structural properties. Such properties in the context of interpolation are analyzed in the last part of Chapter 3.

We conclude this work by reviewing the obtained results and pointing out further directions to investigate.

Chapter 2

Deductive systems

The notion of *schematic system* is an original idea of Parikh [29] whose main purpose is to serve as a general framework for formal deduction. We will describe schematic systems as well as *restricted schematic systems*. *Schematic systems* for *classical first order logic* and *classical first order logic with equality* will be presented. We also describe restricted schematic systems for *intuitionistic* and *minimal first order logics*. Important results on such systems, as *cut-elimination* and the *subformula property* are described.

The notion of *logical graphs*, due to Buss [7] and whose original aim was to study how the influence of a formula spreads through a deduction, will be introduced, in order to serve as a tool to study *interpolation* and *interpolant complexity*. A generic set theoretic approach for deductive systems, called *structured sets* [10], is introduced in order to study the combinatorial nature of interpolation.

2.1 Basic concepts

We will denote the set $\{0, 1, 2, 3, 4, \dots\}$ as \mathbb{N}_0 , and the set $\mathbb{N}_0 \setminus \{0\}$ as \mathbb{N} .

Definition 2.1.1 (Signature). A *first order signature* or simply a *signature* is a tuple $\langle F, P, C, Q \rangle$, where $F = \{F_i\}_{i \in \mathbb{N}_0}$, $P = \{P_i\}_{i \in \mathbb{N}_0}$, $Q = \{Q_i\}_{i \in \mathbb{N}}$ and $C = \{C_i\}_{i \in \mathbb{N}}$ are families of sets.

Sets F, P, C and Q of a signature Σ will be denoted by $F_\Sigma, P_\Sigma, C_\Sigma$ and Q_Σ , respectively. Unless otherwise is mentioned we will consider that:

- $F_i = \{f^i, g^i, h^i, f_0^i, g_0^i, h_0^i, f_1^i, \dots\}$ for $i \in \mathbb{N}_0$ are sets of *function symbols* of arity i , and they will be referred as *functions*;
- $P_i = \{P^i, Q^i, R^i, P_0^i, Q_0^i, R_0^i, P_1^i, \dots\}$ for $i \in \mathbb{N}_0$ are sets of *predicate symbols* of arity i , and they will be referred as *predicates*;

- $C_i = \{c_0^i, c_1^i, c_2^i, \dots\}$ for $i \in \mathbb{N}$ are sets of *logical connective symbols* of arity i , and they will be referred as *connectives*; and
- $Q_i = \{q_0^i, q_1^i, q_2^i, \dots\}$ for $i \in \mathbb{N}$ are sets of *quantifier symbols* of arity i , and they will be referred as *quantifiers*.

We consider as *constant symbols* the functions symbols with arity 0, they will be referred simply as *constants* and will be denoted, unless other is mentioned, as $a, b, a_0, b_0, a_1, \dots$.

When the arity of a symbol is clear from the context the exponent of the symbol will not be written. Also, we will consider a fixed countable set of *quantification variable symbols* $\{v, x, y, v_0, x_0, y_0, v_1, \dots\}$ usually referred simply as *variables*.

Regarding metamathematical symbols we will consider fixed countable sets of

- *schema variables* $\{\mu, \mu_0, \mu_1, \mu_2, \dots\}$;
- *term schema variables* $\{\tau, \tau_0, \tau_1, \tau_2, \dots\}$ (referred as *term variables*);
- *formula schema variables* $\{\xi, \xi_0, \xi_1, \xi_2, \dots\}$ (referred as *formula variables*).

Definition 2.1.2 (Terms). The *set of terms*, usually called *terms*, induced by a signature Σ , is inductively defined as follows:

- every variable, schema variable, term variable and constant (of Σ) is a *term*;
- $f^n(\mathbf{t}_1, \dots, \mathbf{t}_n)$ is a *term*, for every $n \in \mathbb{N}$ such that $f^n \in F_n$ in Σ and \mathbf{t}_i ($i = 1, \dots, n$) is a term.

Terms that do not contain schema variables neither term variables occurrences will be called *regular terms*.

Unless otherwise is mentioned we will denote terms as $\mathbf{r}, \mathbf{s}, \mathbf{t}, \mathbf{r}_0, \mathbf{s}_0, \mathbf{t}_0, \mathbf{t}_1, \mathbf{s}_1, \dots$, and regular terms as $r, s, t, r_0, s_0, t_0, t_1, \dots$.

Definition 2.1.3 (Formulas). Consider a signature Σ . Then the set of *formulas* induced by Σ is inductively defined as follows:

1. $P^n(\mathbf{t}_1, \dots, \mathbf{t}_n)$ is a *formula* (called *atomic formula*), for any predicate $P^n \in P_n$ and terms \mathbf{t}_i ($i = 1, \dots, n$);
2. ξ is a *formula*, for any formula variable ξ ;

3. $c(\mathcal{A}_1, \dots, \mathcal{A}_n)$ is a *formula*, for all $n \in \mathbb{N}$, any $c \in C_n$ and formulas \mathcal{A}_i ($i = 1, \dots, n$); and
4. $qv(\mathcal{A}_1, \dots, \mathcal{A}_n)$ is also a *formula*, for all $n \in \mathbb{N}$, any $q \in Q_n$, formulas \mathcal{A}_i ($i = 1, \dots, n$), and variable or schema variable v .

We will denote the set of formulas induced by a Σ as $L(\Sigma)$.

If every term occurring in a formula \mathcal{A} is a regular term, and there are no formula variable occurrences in \mathcal{A} , then we say \mathcal{A} is a *regular formula*.

Unless otherwise is mentioned we will denote formulas as $\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{A}_0, \dots$ and regular formulas as A, B, C, A_0, \dots

2.2 Schematic systems

Definition 2.2.1 (Schematic sequent). Given a signature Σ , a *schematic sequent* or *sequent* over Σ is a pair $\langle \Gamma, \Delta \rangle$ written

$$\Gamma \vdash \Delta,$$

where both Γ and Δ can be either sets, sequences or multisets of formulas in $L(\Sigma)$, and they are named *antecedent* and *succedent*, respectively.

Consider a formula \mathcal{A} occurring in a schematic sequent, then we write either \mathcal{A}^1 or \mathcal{A}^2 in order to denote its position in the sequent, as antecedent or as succedent, respectively.

Definition 2.2.2 (Substitution). A *substitution* δ_0 is an assignment of variables to schema variables, of regular terms to term variables, and of regular formulas to formula variables.

The substitution δ_0 induces a map δ_1 from terms to regular terms inductively defined as follows:

$$\begin{aligned} \delta_1(\mu_i) &= \delta_0(\mu_i), \\ \delta_1(\tau_i) &= \delta_0(\tau_i), \\ \delta_1(f^n(\mathbf{t}_1, \dots, \mathbf{t}_n)) &= f^n(\delta_1(\mathbf{t}_1), \dots, \delta_1(\mathbf{t}_n)), \end{aligned}$$

for any $i, n \in \mathbb{N}_0$

δ_1 and δ_0 induce another map δ_2 on formulas inductively defined by

$$\begin{aligned} \delta_2(P^m(\mathbf{t}_1, \dots, \mathbf{t}_n)) &= P^m(\delta_1(\mathbf{t}_1), \dots, \delta_1(\mathbf{t}_n)), \\ \delta_2(\xi) &= \delta_0(\xi), \\ \delta_2(c^n(\mathcal{A}_1, \dots, \mathcal{A}_n)) &= c^n(\delta_2(\mathcal{A}_1), \dots, \delta_2(\mathcal{A}_n)), \\ \delta_2(q^n v(\mathcal{A}_1, \dots, \mathcal{A}_n)) &= q^n v(\delta_2(\mathcal{A}_1), \dots, \delta_2(\mathcal{A}_n)), \\ \delta_2(q^n \mu(\mathcal{A}_1, \dots, \mathcal{A}_n)) &= q^n \delta_1(\mu)(\delta_2(\mathcal{A}_1), \dots, \delta_2(\mathcal{A}_n)), \end{aligned}$$

for any $n \in \mathbb{N}$ and $m \in \mathbb{N}_0$. A *substitution* applied to a sequent $\mathcal{A}_1, \dots, \mathcal{A}_l \vdash \mathcal{A}_{l+1}, \dots, \mathcal{A}_k$ is defined as the sequent

$$\delta_2(\mathcal{A}_1), \dots, \delta_2(\mathcal{A}_l) \vdash \delta_2(\mathcal{A}_{l+1}), \dots, \delta_2(\mathcal{A}_k).$$

Moreover, we define

$$\delta_2(\{\mathcal{A}_1, \dots, \mathcal{A}_n\}) = \{\delta_2(\mathcal{A}_1), \dots, \delta_2(\mathcal{A}_n)\}.$$

In the sequel we will denote δ_0, δ_1 and δ_2 simply as δ .

Definition 2.2.3 (Proviso). Consider a variable v , a schematic variable μ , a term variable τ , formulas $\mathcal{A}_1, \mathcal{A}_2$, sequences (sets or multisets) of formulas Γ and Δ , and a sequent \mathcal{S} . A *proviso* has the form:

1. provided τ is free for μ (for v) in \mathcal{A}_1 (in \mathcal{S}); or
2. provided μ (v) is not free in \mathcal{A}_1 (in \mathcal{S}); or
3. provided μ (v) does not occur in \mathcal{A}_1 (in \mathcal{S}); or
4. Δ in $\Gamma \vdash \Delta$ contains at most only one formula; or
5. every predicate or formula variable in \mathcal{A}_1 also occurs in \mathcal{A}_2 .

Definition 2.2.4 (Rule). A *rule* is a tuple $\langle \mathcal{S}_0, \mathcal{W}, \mathcal{R} \rangle$, where \mathcal{S}_0 is a sequent, called the *consequence* or *lower sequent*, \mathcal{W} is a finite sequence (possibly empty) of sequents, called the *premises* or *upper sequent*, and \mathcal{R} is a finite set (possibly empty) of provisos.

The number of premises of a rule is said to be its *arity*.

Definition 2.2.5 (Schematic axiom). A *schematic axiom* or *axiom* is a rule $\langle \mathcal{S}_0, \mathcal{W}, \mathcal{R} \rangle$, where \mathcal{W} is empty and \mathcal{S}_0 has the form

$$\mathcal{A}_1, \Gamma \vdash \Delta, \mathcal{A}_2,$$

where \mathcal{A}_1 and \mathcal{A}_2 are called *distinguished occurrences* in the axiom, and Γ and Δ denote arbitrary collections of formulas called *side formulas* of the axiom. There must be at least one distinguished occurrence, and all predicates constants in one of the distinguished occurrences must be also in the other distinguished occurrence.

Note that the axioms have no premises.

Definition 2.2.6 (Derived axiom). We say an axiom $A_1, \Theta \vdash \Lambda, A_2$ is *derived* from a schematic axiom $\mathcal{A}_1, \Gamma \vdash \Delta, \mathcal{A}_2$ if there is a substitution δ such that $\delta(\mathcal{A}_1) = A_1$ and $\delta(\mathcal{A}_2) = A_2$, and Θ and Λ contains only regular formulas.

It should be stressed that axioms differing only in the side formulas are considered to be equivalent.

Definition 2.2.7 (Schematic rule). A *schematic rule* is a rule $\langle \mathcal{S}_0, \mathcal{W}, \mathcal{R} \rangle$, where \mathcal{W} is not empty. It is usually written as

$$\frac{\mathcal{S}_1[\mathcal{A}_{1,1} \dots \mathcal{A}_{1,n_1}], \dots, \mathcal{S}_k[\mathcal{A}_{k,1} \dots \mathcal{A}_{k,n_k}]}{\mathcal{S}_0[\mathcal{A}_0]}.$$

\mathcal{A}_0 is called the *main formula* of the rule, and $\mathcal{A}_{i,1}, \dots, \mathcal{A}_{i,n_i}$ are called the *auxiliary occurrences (formulas)* in the i -th sequent of the rule. A *side formula* of a rule is the one which is not a main one nor an auxiliary occurrence. If among the provisos there is one that forbids the occurrence of a variable or schema variable in the consequence, then we call such variable or schema variable an *eigenvariable*. Additionally the following conditions must be fulfilled

1. every predicate and formula variable occurring in the auxiliary formulas must also occur in the main formula, and
2. if a side formula occurs in the antecedent (succedent) in the premise of a rule, then it also occurs in the antecedent (succedent), as a side formula, in the consequent, and vice versa.

Condition 1 is called *weak embeddability*.

A *step of inference* or *inference* is defined from an schematic rule analogously as derived axiom is defined from schematic axiom.

Definition 2.2.8 (Schematic system). We define a *schematic system* as a tuple $\langle \Sigma, \Xi \rangle$, where Σ is a signature, Ξ is a set of rules, and every formula occurring in the rules of Ξ is induced by Σ .

Definition 2.2.9 (Deduction or derivation). A *deduction* or *derivation* in a schematic system \mathcal{Z} , also called a \mathcal{Z} -*deduction*, is a tree of sequents of *regular* formulas, where sequents at the top nodes (*initial sequents*) are derived axioms and all other sequents are consequences of steps of inference applied to their immediate ancestors.

The sequent in the bottom of a deduction is called the *end-sequent*. Suppose \mathcal{S} is the end-sequent of a deduction in a schematic system \mathcal{Z} , then we say \mathcal{S} is *derivable* in \mathcal{Z} (\mathcal{Z} -*derivable*).

Definition 2.2.10 (Height of a deduction). We inductively define the *height of a deduction* \mathcal{D} as follows:

If \mathcal{D} is an axiom, then the height of \mathcal{D} is 0.

If \mathcal{D} is a deduction whose root ends with a step of inference, then the height of \mathcal{D} is 1 plus the maximum of the heights of the deductions that make up the sub-trees of \mathcal{D} , corresponding to the premises.

Definition 2.2.11. Consider a schematic system \mathcal{Z} , a set of formulas Φ and a finite set (sequence or multiset) of regular formulas Γ_0 such that every formula in Γ_0 is derived from one in Φ . Then we define:

- If $\Gamma_0, \Gamma \vdash \Delta$ is derivable in \mathcal{Z} , then $\Gamma \vdash \Delta$ is said to be *derivable from* Φ in \mathcal{Z} , and it is written $\Phi, \Gamma \vdash_{\mathcal{Z}} \Delta$.
- Φ is *inconsistent* with respect to \mathcal{Z} if the empty sequent \vdash is derivable from Φ in \mathcal{Z} .
- If Φ is not consistent then we say it is *consistent*.
- If all functions and predicates in a regular formula A occur in Γ_0 , then A is said to be *dependant* on Φ .

2.2.1 Regular schematic systems

Definition 2.2.12 (Regularity). A pair of schematic rules with the forms

$$\frac{\mathcal{S}_1[\mathcal{A}_1] \dots \mathcal{S}_k[\mathcal{A}_k]}{\mathcal{S}_0[\mathcal{A}_0]} \quad \text{and} \quad \frac{\mathcal{S}[\mathcal{A}'_1, \dots, \mathcal{A}'_k]}{\mathcal{S}[\mathcal{A}_0]},$$

is said to be *k-regular* or simply *regular* if, for $i = 1, \dots, k$,

1. \mathcal{A}_i and \mathcal{A}'_i are atomic formulas or formula variables,
2. \mathcal{A}_i can be obtained from \mathcal{A}'_i by renaming some of the variables, term variables, schema variables, functions, constants, and/or the predicate or the formula variable,
3. \mathcal{A}_i and \mathcal{A}'_i are in opposite sides in \mathcal{S}_i and \mathcal{S} , respectively, it also holds for \mathcal{A}_0 , in \mathcal{S} and \mathcal{S}_0 ,
4. provisos 1, 2 and 3 of Definition 2.2.3 are not allowed, except in the case when there are quantifiers in \mathcal{A}_0 , in such a case, each \mathcal{A}_i is obtained from \mathcal{A}'_i by replacing variables and schema variables for eigenvariables, and eigenvariables for variables and schema variables, and
5. if \mathcal{A}_0 contains no quantifier, every predicate and formula variable in \mathcal{A}_0 occurs also in $\bigcup_{i=1}^k \mathcal{A}_i$, otherwise every predicate and formula variable in

\mathcal{A}_0 occurs also in $\bigcup_{i=1}^k \mathcal{A}_i''$, where \mathcal{A}_i'' is obtained from \mathcal{A}_i by replacing some variables and schema variables for eigenvariables, and eigenvariables for variables and schema variables.

Rules which belong to a regular pair are called *regular rules*, and they are the *dual* of each other.

Definition 2.2.13 (\mathcal{Z} -definability). A rule is \mathcal{Z} -definable if either it belongs to a schematic system \mathcal{Z} and it is regular or it is obtained by the successive application of regular rules of \mathcal{Z} .

Example 2.2.14. Consider the pairs of regular rules

$$\frac{\mathcal{A}_1, \mathcal{A}_2, \Gamma \vdash \Delta}{\mathcal{A}_1 \wedge \mathcal{A}_2, \Gamma \vdash \Delta}, \quad \frac{\Gamma \vdash \Delta, \mathcal{A}_1 \quad \Gamma \vdash \Delta, \mathcal{A}_2}{\Gamma \vdash \Delta, \mathcal{A}_1 \wedge \mathcal{A}_2},$$

$$\frac{\mathcal{A}_1, \Gamma \vdash \Delta}{\Gamma \vdash \Delta, \neg \mathcal{A}_1}, \quad \frac{\Gamma \vdash \Delta, \mathcal{A}_1}{\neg \mathcal{A}_1, \Gamma \vdash \Delta},$$

in a schematic system \mathcal{Z} , then

$$\frac{\Gamma \vdash \Delta, \mathcal{A}_1 \quad \Gamma \vdash \Delta, \mathcal{A}_2}{\neg(\mathcal{A}_1 \wedge \mathcal{A}_2), \Gamma \vdash \Delta} \quad \text{and} \quad \frac{\mathcal{A}_1, \mathcal{A}_2, \Gamma \vdash \Delta}{\Gamma \vdash \Delta, \neg(\mathcal{A}_1 \wedge \mathcal{A}_2)}$$

are \mathcal{Z} -definable.

Definition 2.2.15 (Regular schematic systems). A schematic system \mathcal{Z} is *regular* if

1. for all $k \in \{k_1, \dots, k_n\}$, where k_i is the arity of the i^{th} rule of \mathcal{Z} ($i = 1, \dots, n$), there is a \mathcal{Z} -definable k -regular pair such that the auxiliary and main formulas are all in the same side of its (their) respective sequent(s);
2. if a system admits provisos on eigenvariables for some of its rules, then for all n ($0 < n \leq m$) there are regular pairs of \mathcal{Z} -definable unary rules with auxiliary formulas $\mathcal{A}_1, \mathcal{A}_1'$ with m variables, and main formula \mathcal{A}_0 where n of the variables and schema variables occurring in $\mathcal{A}_1, \mathcal{A}_1'$ appear non free in \mathcal{A}_0 ;
3. for all $k \in \{k_1, \dots, k_n\}$, where k_i is the arity of the i^{th} rule of \mathcal{Z} ($i = 1, \dots, n$), there is a \mathcal{Z} -definable k -regular pair such that

$$\frac{\overbrace{\dots \mathcal{S}_i[A_i^j] \dots}^{k \text{ sequents}}}{\mathcal{S}_i[A_0^r]}, \quad \frac{\mathcal{S}_i[A_i^r]}{\mathcal{S}_i[A_0^j]},$$

where $j \neq r$, and A_i contains all predicates and formula variables (not including the special predicate symbols) that occur in A_0 .

2.2.2 Classical logic

Here, we introduce a formulation of the first order predicate calculus LK (logistischer klassischer Kalkül). This calculus was first presented by Gerhard Gentzen [17] as a deductive system for classical first order logic.

Definition 2.2.16 (LK). Consider the signature Σ , where $C_1 = \{\neg\}$, $C_2 = \{\vee, \wedge, \rightarrow\}$, $C_i = \emptyset$ ($i \in \mathbb{N} \setminus \{1, 2\}$), $Q_1 = \{\forall, \exists\}$ and $Q_i = \emptyset$ ($i \in \mathbb{N} \setminus \{1\}$). Also consider *sequences* of formulas Γ, Δ, Π and Λ . The system LK is a regular schematic system $\langle \Sigma, \Xi \rangle$ such that Ξ is composed by:

1. the *schematic axiom*¹

$$\xi \vdash \xi,$$

where ξ is a non regular atomic formula;

2. the *weakening rules, left and right*, respectively,

$$\frac{\Gamma \vdash \Delta}{\xi, \Gamma \vdash \Delta}, \quad \frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, \xi};$$

3. the *left and right contraction rules*, respectively,

$$\frac{\xi, \xi, \Gamma \vdash \Delta}{\xi, \Gamma \vdash \Delta}, \quad \frac{\Gamma \vdash \Delta, \xi, \xi}{\Gamma \vdash \Delta, \xi};$$

4. the *left and right exchange rules*, respectively,

$$\frac{\Gamma, \xi_1, \xi_2, \Pi \vdash \Delta}{\Gamma, \xi_2, \xi_1, \Pi \vdash \Delta}, \quad \frac{\Gamma \vdash \Delta, \xi_1, \xi_2, \Lambda}{\Gamma \vdash \Delta, \xi_2, \xi_1, \Lambda}.$$

5. the *cut rule*

$$\frac{\Gamma \vdash \Delta, \xi \quad \xi, \Pi \vdash \Lambda}{\Gamma, \Pi \vdash \Delta, \Lambda},$$

where the *instances* of ξ , in the steps of inference of the cut rule, are called the *cut formulas*;

6. the *left and right negation rules*, respectively,

$$\frac{\Gamma \vdash \Delta, \xi}{\neg \xi, \Gamma \vdash \Delta}, \quad \frac{\xi, \Gamma \vdash \Delta}{\Gamma \vdash \Delta, \neg \xi};$$

¹Axioms derived from the schematic axiom, in LK-proofs, have the *same atomic regular formula* as antecedent and succedent.

7. the *left conjunction rules*

$$\frac{\xi_1, \Gamma \vdash \Delta}{\xi_1 \wedge \xi_2, \Gamma \vdash \Delta}, \quad \frac{\xi_2, \Gamma \vdash \Delta}{\xi_1 \wedge \xi_2, \Gamma \vdash \Delta};$$

the *right conjunction rule*

$$\frac{\Gamma \vdash \Delta, \xi_1 \quad \Gamma \vdash \Delta, \xi_2}{\Gamma \vdash \Delta, \xi_1 \wedge \xi_2};$$

8. the *left disjunction rule*

$$\frac{\xi_1, \Gamma \vdash \Delta \quad \xi_2, \Gamma \vdash \Delta}{\xi_1 \vee \xi_2, \Gamma \vdash \Delta};$$

the *right disjunction rules*

$$\frac{\Gamma \vdash \Delta, \xi_1}{\Gamma \vdash \Delta, \xi_1 \vee \xi_2}, \quad \frac{\Gamma \vdash \Delta, \xi_2}{\Gamma \vdash \Delta, \xi_1 \vee \xi_2};$$

9. the *left implication rule*

$$\frac{\Gamma \vdash \Delta, \xi_1 \quad \xi_2, \Gamma \vdash \Lambda}{\xi_1 \rightarrow \xi_2, \Gamma \vdash \Delta, \Lambda};$$

the *right implication rule*

$$\frac{\xi_1, \Gamma \vdash \Delta, \xi_2}{\Gamma \vdash \Delta, \xi_1 \rightarrow \xi_2};$$

10. the *left universal quantification rule* is

$$\frac{\xi', \Gamma \vdash \Delta}{\forall \mu \xi, \Gamma \vdash \Delta}$$

where ξ' is obtained from ξ by replacing μ by a term variable;

the *right universal quantification rule* is

$$\frac{\Gamma \vdash \Delta, \xi'}{\Gamma \vdash \Delta, \forall \mu \xi}$$

where ξ' is obtained from ξ by replacing μ by a schema variable μ' with the proviso that μ' does not occur in the lower sequent;

11. the *left existential quantification rule* is

$$\frac{\xi', \Gamma \vdash \Delta}{\exists \mu \xi, \Gamma \vdash \Delta}$$

where ξ' is obtained from ξ by replacing μ by a schema variable μ' with the proviso that μ' does not occur in the lower sequent; and the *right existential quantification rule* is

$$\frac{\Gamma \vdash \Delta, \xi'}{\Gamma \vdash \Delta, \exists \mu \xi}$$

where ξ' is obtained from ξ by replacing μ by a term variable.

Weakening, contraction, exchange and cut rules form the so-called *structural rules*. The *logical rules* are the ones which are not structural.

Cut-elimination

A very important result about LK is the cut-elimination theorem, also known as Gentzen's Hauptsatz [17]. The cut rule can be seen as the use of lemmas in the proof of theorems. Then, Gentzen's Hauptsatz states that theorems can be proved without detours.

Theorem 2.2.17 (Gentzen's Hauptsatz [17]). *If a sequent is derivable, then it is also derivable without the use of the cut rule.*

In order to follow Gentzen's original proofs, notice that the following *mix rules* are LK-definable.

Definition 2.2.18 (Mix rules). The *mix rules* are defined by

$$\frac{\mathcal{S}[\overbrace{\xi^k, \dots, \xi^k}^{n \text{ times}}] \quad \mathcal{S}[\overbrace{\xi^{k'}, \dots, \xi^{k'}}^{n \text{ times}}]}{\mathcal{S}}$$

for every $n \in \mathbb{N}$, where k denotes the position (antecedent or succedent) of the formula ξ and $k \neq k'$. In a derived inference from the mix rule, the *instance* of ξ is called the *mix formula*.

Let's call LK* the system resulting from replacing the cut rule by the mix rules. It is easy to see that LK is equivalent to LK*, i.e., a formula in LK is derivable if and only if it is derivable in LK*.

Now we define two ways of measuring the complexity of a deduction.

Definition 2.2.19 (Grade). The *grade* of a formula is the number of logical symbols it has. The *grade* of a mix is the grade of its mix formula. The *grade* of a deduction, with a step of inference from a mix rule as its last inference, is the grade of its last step of inference.

Consider a deduction \mathcal{D} which contains a mix rule as its last inference named J . A *thread* in \mathcal{D} is a sequence of sequents $\mathcal{S}_1, \dots, \mathcal{S}_n$ such that \mathcal{S}_n is the end-sequent of \mathcal{D} , \mathcal{S}_1 is a initial sequent in \mathcal{D} , and \mathcal{D}_i is a premise of a step of inference in \mathcal{D} with consequence \mathcal{S}_j ($0 < i < j \leq n$). We call a thread in \mathcal{D} a *left (right) thread* if it contains the left (right) upper sequent of the mix J .

Definition 2.2.20 (Rank). The *rank* of a thread T in \mathcal{D} is defined as follow: if T is a left (right) thread, then the rank of T is the number of consecutive sequents, counting upward from the left (right) upper sequent of J , that contains the mix formula in its succedent (antecedent). The *left (right) rank* of \mathcal{D} is the maximum rank of the left (right) threads of \mathcal{D} . The *rank* of \mathcal{D} is the sum of its left and right ranks.

Gentzen's Hauptsatz is an easy consequence of the following lemma.

Lemma 2.2.21. *If \mathcal{D} is a deduction of \mathcal{S} in LK^* which contains only one mix rule, occurring as the last inference, then \mathcal{S} is derivable without a mix.*

Proof (sketch). By double induction over the grade and the rank of the deduction \mathcal{D} . ■

Definition 2.2.22 (Substitution on regular terms). Consider a map $[\]$ from regular terms to regular terms inductively defined by

$$\begin{aligned} [v_1]_t^{v_2} &= v_1; \\ [v_1]_t^{v_1} &= t; \\ [f(t_1, \dots, t_k)]_t^v &= f([t_1]_t^v, \dots, [t_k]_t^v). \end{aligned}$$

$[\]$ induces a map from regular formulas to regular formulas, which will also be called $[\]$ and it is inductively defined by

$$\begin{aligned} [P(t_1, \dots, t_n)]_t^v &= P([t_1]_t^v, \dots, [t_n]_t^v); \\ [c(A_1, \dots, A_n)]_t^v &= c([A_1]_t^v, \dots, [A_n]_t^v); \\ [qv'(A_1, \dots, A_n)]_t^v &= qv'([A_1]_t^v, \dots, [A_n]_t^v). \end{aligned}$$

By a *subformula* of a regular formula A we mean a formula used in building up A . More precisely:

Definition 2.2.23 (Subformula). The only subformula of an atomic regular formula is the formula itself. Considering a connective c , of arity n , then the subformulas of $c(A_1, \dots, A_n)$ are the subformulas of A_1, \dots, A_n and $c(A_1, \dots, A_n)$ itself. In case q is a quantifier of arity n , then the subformulas of $qv(A_1, \dots, A_n)$ are the subformulas of $[A_1]_v^t, \dots, [A_n]_v^t$ and $qv(A_1, \dots, A_n)$ itself.

Theorem 2.2.24 (Subformula property). *In a cut-free deduction in LK all the formulas which occur in it are subformulas of the formulas in the end-sequent.*

Proof (sketch). By mathematical induction on the height of the cut-free deduction. ■

So, cut-elimination tells us that if a formula is derivable in LK at all, it is derivable by use of its subformulas only.

The predicate calculus with equality

In the sequel we use the notation

$$\tau[\tau_1, \dots, \tau_n], \xi[\tau'_1, \dots, \tau'_n]$$

when there are substitutions δ, δ' , respectively, such that

$$\begin{aligned} \delta(\tau[\tau_1, \dots, \tau_n]) &= f(\delta(\tau_1), \dots, \delta(\tau_n)), \\ \delta'(\xi[\tau'_1, \dots, \tau'_n]) &= P(\delta'(\tau'_1), \dots, \delta'(\tau'_n)), \end{aligned}$$

for any function f and predicate P of arity n ($n \in \mathbb{N}$).

Definition 2.2.25 (LKe). The predicate calculus with equality LKe can be obtained from LK by specifying a predicate of two argument places ($=$) and adding the following sequents as additional axioms:

$$\begin{aligned} &\vdash \tau = \tau, \\ \tau_1 = \tau'_1, \dots, \tau_n = \tau'_n &\vdash \tau[\tau_1, \dots, \tau_n] = \tau[\tau'_1, \dots, \tau'_n], \\ \tau_1 = \tau'_1, \dots, \tau_n = \tau'_n, \xi[\tau_1, \dots, \tau_n] &\vdash \xi[\tau'_1, \dots, \tau'_n]. \end{aligned}$$

Proposition 2.2.26. *Consider an arbitrary regular formula $A(a_1, \dots, a_n)$. Then*

$$s_1 = t_1, \dots, s_n = t_n, A(s_1, \dots, s_n) \vdash A(t_1, \dots, t_n)$$

is derivable in LKe, for any regular terms s_i, t_i ($1 \leq i \leq n$). Furthermore, $s = t \rightarrow t = s$ and $s_1 = s_2, s_2 = s_3 \vdash s_1 = s_3$ are also derivable.

Definition 2.2.27. Denote by Γ_e the set formed by the following formulas:

$$\begin{aligned} & \forall \mu (\mu = \mu); \\ & \forall \mu_1 \dots \forall \mu_n \forall \mu'_1 \dots \forall \mu'_n \quad (\mu_1 = \mu'_1 \wedge \dots \wedge \mu_n = \mu'_n \rightarrow \\ & \quad \tau[\mu_1, \dots, \mu_n] = \tau[\mu'_1, \dots, \mu'_n]); \\ & \forall \mu_1 \dots \forall \mu_n \forall \mu'_1 \dots \forall \mu'_n \quad (\mu_1 = \mu'_1 \wedge \dots \wedge \mu_n = \mu'_n \wedge \\ & \quad \xi[\mu_1, \dots, \mu_n] \rightarrow \xi[\mu'_1, \dots, \mu'_n]). \end{aligned}$$

Proposition 2.2.28. *A sequent $\Gamma \vdash \Delta$ is derivable in LKe if and only if $\Gamma, \Gamma_e \vdash \Delta$ is derivable in LK.*

Proof (sketch). It is easy to see that all the axioms of LKe are derivable from Γ_e . Therefore the proposition is proved by mathematical induction on the derivation of $\Gamma \vdash \Delta$. In the other hand $\vdash \Gamma_e$ is derivable in LKe. ■

Definition 2.2.29 (Essential cut). If the cut formula of a cut in LKe is of the form $s = t$, then the cut is called *inessential*. It is called *essential* otherwise.

Theorem 2.2.30 (Cut-elimination for LKe [37, 40]). *If a sequent is derivable in LKe then it is also derivable without an essential cut.*

The same technique used for the LK case is applied for this proof. For details see [37, 40].

2.3 Restricted schematic systems

In this section, we introduce a modified notion of schematic system, namely its restricted version. The restricted schematic systems are schematic systems with only one formula in the succedents of the schematic sequents. This modified version has the intention to be a generalization of logical systems for intuitionistic logic.

Definition 2.3.1 (Restricted schematic system). A *restricted schematic system* is a schematic system $\langle \Sigma, \Xi \rangle$, where each sequent of each rule in Ξ has the proviso that *at most only one formula* composes the succedent.

Definition 2.3.2 (Regular restricted schematic system). A *regular restricted schematic system* is a restricted schematic system that satisfy the conditions of regular schematic systems (Definition 2.2.15).

2.3.1 Intuitionistic and minimal logics

LJ (J stands for intuitionistic) was the first deductive system developed for intuitionistic logic. It was first introduced by Gentzen [17] as a subsystem of LK.

Minimal logic differs from intuitionistic logic by the rejection of the principle of explosion. The *ex falso sequitur quodlibet* principle or *principle of explosion* states that everything can follow from a contradiction.

Herein, we present restricted regular schematic systems for intuitionistic and minimal logics which are subsystems of LK. Also, we will show how to avoid structural rules. For such a purpose first we will present a system equivalent to LK.

Definition 2.3.3 (G1c system). The G1c system is a modification of LK, where multisets, instead of sequences, are considered in sequents in order to avoid exchange rules, an additional axiom is considered ($\perp \vdash$) in order to avoid the negation rules, and the left implication rule is modified to a shared version. Due to Gentzen's Hauptsatz, the cut rule is not considered for G1c. Moreover, the signature of G1c is not exactly the same as the one of LK. In particular, besides the consideration of \perp as a special predicate symbol, the negation symbol is no longer a connective.

It is not hard to see that G1c is equivalent to LK if we consider the equivalence between \perp and $\mathcal{A} \wedge \neg\mathcal{A}$, and $\neg\mathcal{A}$ with $\mathcal{A} \vdash \perp$.

Now, we introduce systems for intuitionistic and minimal logic as subsystems of G1c. Note that those systems are regular restricted schematic systems.

Definition 2.3.4 (G1i system). The regular restricted schematic system G1i system is obtained from G1c by considering only sequents with succedents of cardinality less or equal than 1, and by modifying the left implication rule to

$$\frac{\Gamma \vdash \xi_1 \quad \xi_2, \Gamma \vdash \Delta}{\xi_1 \rightarrow \xi_2, \Gamma \vdash \Delta}.$$

Definition 2.3.5 (G1m system). The G1m system is obtained from the G1i system by avoiding the axiom $\perp \vdash$.

Definition 2.3.6 (G2[mic] systems). The G2c system is obtained from the G1c system by avoiding the weakening rules. G2i is obtained from the G2c system by restricting to succedents of cardinality less or equal than 1. Finally, G2m is obtained from G2i by avoiding the axiom $\perp \vdash$.

Proposition 2.3.7 (Equivalence between G1[mic] and G2[mic][40]).

$$\Gamma \vdash_{G1[mic]} \Delta \quad \text{iff} \quad \Gamma \vdash_{G2[mic]} \Delta,$$

which means that weakening rules can be absorbed by logical ones in systems for classical, intuitionistic and minimal logic. Note Δ is composed by at most one formula in the cases for G1[mi] and G2[mi].

Definition 2.3.8 (G3 systems). The G3[mic] system, is obtained from G2[mic], by dropping out the contraction rules and modifying the axioms to

$$\begin{aligned} &\xi, \Gamma \vdash \Delta, \xi; \text{ and} \\ &\perp, \Gamma \vdash \Delta. \end{aligned}$$

Proposition 2.3.9 (Equivalence between $G2[mic]$ and $G3[mic]$ [40]).

$$\Gamma \vdash_{G2[mic]} \Delta \quad \text{iff} \quad \Gamma \vdash_{G3[mic]} \Delta,$$

which means that structural rules can be avoided in systems for classical, intuitionistic and minimal logic. Note Δ is at most one formula in the cases for $G2[mi]$ and $G3[mi]$.

Without essential changes the proof of cut-elimination for LK can serve also for G systems. Here we simply states the theorem. For details see [40].

Theorem 2.3.10 (Cut-elimination for G systems). *If a sequent is derivable in $G[123][mic]+cut$, then it is also derivable in $G[123][mic]$.*

Then we conclude that structural rules are not necessary in deductive systems for classical, intuitionistic and minimal logic.

Finally, we state the subformula property for G systems.

Theorem 2.3.11 (Subformula property for G systems). *In a cut-free deduction in $G[123][mic]$ all the formulas which occur in it are subformulas of the formulas in the end-sequent.*

Proof (sketch). By mathematical induction on the height of the cut-free deduction. ■

2.4 Logical graphs

In order to prove interpolation in Chapter 3 we will use the notion of logical flow graphs, as introduced by Buss [7].

Definition 2.4.1 (S-formula). An *s-formula* is an occurrence of a subformula of a formula in a deduction, as opposed to the subformula itself that can occur several times in the deduction.

Consider the derivation

$$\frac{\frac{\mathbf{A}_1 \vdash A_1 \quad \frac{\mathbf{A}_2 \vdash A_2 \quad \mathbf{A}_3 \vdash A_3}{A_2, A_2 \rightarrow A_3 \vdash A_3}}{A_1 \rightarrow A_2, A_2 \rightarrow A_3, A_1 \vdash A_3}}{\vdash (A_1 \rightarrow A_2) \rightarrow [(A_2 \rightarrow A_3) \rightarrow (A_1 \rightarrow A_3)]}$$

in LK. Note that the double line rules means that the lower sequent in the rule is the consequence of the upper sequent after the application of 1 or more rules. $\mathbf{A}_1, \mathbf{A}_2$ and \mathbf{A}_3 in the initial sequents are s-formulas of the subformulas A_1, A_2 and A_3 , respectively. Notice that the formulas A_1, A_2 , and A_3 occurring in other parts of the deduction are considered as different s-formulas.

Definition 2.4.2 (Variant). Let $\mathcal{A}(\mathfrak{t}_1, \dots, \mathfrak{t}_n)$ be a formula, then $\mathcal{A}(\mathfrak{t}'_1, \dots, \mathfrak{t}'_n)$ is considered a *variant* of it when at least one $\mathfrak{t}_i \neq \mathfrak{t}'_i$ ($i = 1, \dots, n$).

Definition 2.4.3 (Logical graph). A *logical graph* in a derivation is a graph obtained by tracing edges between the atomic s-formulas in the derivation such that:

1. For each axiom derived from a schematic axiom by means of a substitution δ , consider A_1 and A_2 as the two distinguished occurrences. Also consider B as an atomic s-formula in A_1 , and B_1, \dots, B_n as atomic s-formulas in A_2 . Now, suppose that either B_1, \dots, B_n are variants of B or B is a variant of each B_1, \dots, B_n , then there is an edge between each formula $\delta(B)$ and some (at least one) $\delta(B_i)$ ($i = 1, \dots, n$).
2. For each step of inference derived from a schematic rule by means of a substitution δ where a atomic s-formula B in a auxiliary formula have variants B_1, \dots, B_n in the main formula, there is an edge between the s-formula $\delta(B)$ and some (at least one) of its variants $\delta(B_i)$ ($i = 1, \dots, n$).
3. For each step of inference derived from a schematic rule by means of a substitution δ , there is an edge between each side formula $\delta(A)$ occurring in the upper sequent(s) and the side formula $\delta(A)$ occurring in the lower sequent.
4. For each pair of s-formulas A_1 and A_2 which are linked with by a logical edge, and for any subformulas B_1 and B_2 of A_1 and A_2 , respectively, there is a link between B_1 and B_2 .

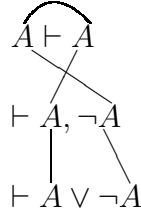
Definition 2.4.4 (Path). Consider a derivation \mathcal{D} . A sequence of consecutive edges of a logical graph in \mathcal{D} where no two edges cross the same axiom or the same step of inference is called a *path* in \mathcal{D} .

A *full path* in a derivation is a path which is maximal in such a derivation, and a *direct path* in a derivation is a path which does not cross any axiom in the derivation.

Example 2.4.5. Consider the following derivation

$$\frac{\frac{A \vdash A}{\vdash A, \neg A}}{A \vee \neg A}$$

A full path in the derivation is



Consider a derivation \mathcal{D} of $A_1 \vdash A_2$, and formula occurrence A_3 in \mathcal{D} . If all direct paths from atomic s-formulas of A_3 in \mathcal{D} go to A_1 (A_2) we say that A_3 goes to A_1 (A_2).

Proposition 2.4.6. Consider a derivation \mathcal{D} of $A_1 \vdash A_2$ with an atomic s-formula A_3 . Then, all full paths from A_3 in the logical graph of \mathcal{D} are either direct or they go back to a variant of A_3 in $A_1 \vdash A_2$.

Proof (sketch). Notice that if a path from A_3 crosses an axiom then it goes back to the end-sequent and in case it does not cross an axiom then it is a direct path. ■

Proposition 2.4.7. Consider a derivation \mathcal{D} of $A_1 \vdash A_2$ and a formula occurrence A_3 in \mathcal{D} . Then, either all direct paths from atomic s-formulas in A_3 down to the end-sequent go to A_1 or they all go to A_2 .

Proof (sketch). Each atomic s-formulas in the upper sequent of a rule can be linked to several s-formulas in the lower sequent, but all the linked s-formulas are in the same formula. ■

2.5 Structured sets

Definition 2.5.1 (Structured sets). A *structured set* is a bipartite collection

$$S = \langle s_1, \dots, s_k \mid s_{k+1}, \dots, s_n \rangle$$

of sets of points which may have a structure. It is written

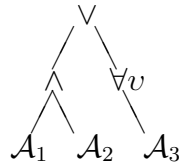
- S^1 or $\langle s_1^1, \dots, s_k^1 \rangle$ instead of $\langle s_1, \dots, s_k \rangle$, and
- S^2 or $\langle s_{k+1}^2, \dots, s_n^2 \rangle$ instead of $\langle s_{k+1}, \dots, s_n \rangle$.

Either, S^1 or S^2 may be empty. A *space* \mathcal{S} is a finite set of structured sets. Given a structured set S , if there is an embedding² between the members of S^1 and S^2 ($s_i \hookrightarrow s_j$), then S is called a *trivial structured set*. A *trivial space* is a space whose members are all trivial.

Example 2.5.2. Consider the first order formula

$$(\mathcal{A}_1 \wedge \mathcal{A}_2) \vee \forall v \mathcal{A}_3,$$

its tree structure can be graphically written as



Consider a sequent of first order formulas

$$\mathcal{A}_1, \dots, \mathcal{A}_k \vdash \mathcal{A}_{k+1}, \dots, \mathcal{A}_n,$$

which can be written as the structured set

$$\langle \mathcal{T}_1, \dots, \mathcal{T}_k \mid \mathcal{T}_{k+1}, \dots, \mathcal{T}_n \rangle,$$

where \mathcal{T}_i is a set of atomic formulas with the tree structure of \mathcal{A}_i ($i = 1, \dots, n$).

²It is called an *embedding* any pseudomap from X to Y everywhere defined on X . An embedding is not required to preserve any structure on X nor to be injective.

Definition 2.5.3 (Operator). An *operator* of arity l and subarity n , is a relation $R : \mathcal{S} \times \dots \times \mathcal{S}$ of arity $l + 1$, such that

- If $R(S_1, \dots, S_l, S_0)$ then there are a set of points s_0 and n sets of points $P = \{s_1, \dots, s_n\}$, such that $P \subset \bigcup S_i$ and $P \cap S_i \neq \emptyset$ ($i = 1, \dots, l$), so that S_0 is either

$$\langle \{s_0\} \cup S_*^1 \mid S_*^2 \rangle \text{ or } \langle S_*^1 \mid \{s_0\} \cup S_*^2 \rangle,$$

where

$$S_*^k = \bigcup_{j=1}^l S_j^k \setminus \{s_i \mid s_i \in S_j^k, 1 \leq i \leq n\}, \text{ for } k \in \{1, 2\};$$

and s_i is embedded into s_0 for $1 \leq i \leq n$.

- If $s_i \in S_j$ and also $s_i \in S'_k$ for $1 \leq i \leq n, 1 \leq \{j, k\} \leq l$, then $R(S'_1, \dots, S'_l, S_0)$.

The sets s_i are called the *arguments* of the operator, s_0 its *value* and S_0 its *output*.

Example 2.5.4. Consider the following rule of LK

$$\frac{\xi_1, \xi_2, \Gamma \vdash \Delta}{\xi_1 \wedge \xi_2, \Gamma \vdash \Delta},$$

whose corresponding operator is R such that

$$R(\{\xi_1, \xi_2, \Gamma, \Delta\}, \{\xi_1 \wedge \xi_2, \Gamma, \Delta\}).$$

Definition 2.5.5 (Surjective operator). A *surjective operator* is an operator whose all points in s_0 embed with at least one point in some s_i , where $i \in \{1, \dots, n\}$.

Definition 2.5.6 (Derived operator). Consider the operators R_l and R_k , of arity l and k respectively, where $l \leq k$. We say that R_l is *derived* from R_k if

- R_l operates over S_1, \dots, S_l ,
- there are sets $S_1 \subseteq S'_1, \dots, S_l \subseteq S'_l$, and S'_{l+1}, \dots, S'_k containing only empty sets,

- the arguments of R_l are the same of R_k together with all sets added to the S_i to get S'_i (with $j = \{1, \dots, l\}$), and all the empty sets s_j (with $j = \{l + 1, \dots, k\}$),
- and all the values and outputs correspond in the obvious way.

Definition 2.5.7 (Transformation). Consider a space $\mathcal{S} = \{S_1, \dots, S_k\}$, and an operator R acting over S_{i_1}, \dots, S_{i_l} and giving as a result S_0 , then a *transformation* on \mathcal{S} induced by R gives as a result the space $\mathcal{S}' = \{S_0\} \cup \mathcal{S} \setminus \{S_{i_1}, \dots, S_{i_l}\}$, such that, every $S \in \mathcal{S}$ and $S \neq S_{i_j}$ (with $j = \{1, \dots, l\}$) is embedded into a copy of itself in \mathcal{S}' .

Definition 2.5.8 (Regular pair). An unary operator acting over S_r , and another operator of arity l acting over S_1, \dots, S_l , are called a *regular pair* if

- exactly one element of each S_i (with $i \in \{1, \dots, l\}$), say s_i is embedded into the corresponding s_0 (in the context of the definition of operator),
- s_i appears in S_r but not in the same component where it appears in S_i ,
- the corresponding s_0 for the operators are the same in both, but it does not appear in the same component.

Two operators are called *duals* if both belong to the same regular pair.

Example 2.5.9. Consider the following rules of LK

$$\frac{\xi_1, \xi_2, \Gamma \vdash \Delta}{\xi_1 \wedge \xi_2, \Gamma \vdash \Delta},$$

$$\frac{\Gamma \vdash \Delta, \xi_1 \quad \Gamma \vdash \Delta, \xi_2}{\Gamma \vdash \Delta, \xi_1 \wedge \xi_2},$$

which represent a regular pair of operators, since ξ_1 and ξ_2 appear in different components in each rule, as well as $\xi_1 \wedge \xi_2$.

Definition 2.5.10 (Regular operator). A *regular operator* is an operator which belongs to a regular pair, and we say that it is *\mathcal{O} -regular*.

Definition 2.5.11 (Constructively operator). Let \mathcal{O} be a set of operators, then an operator is *constructively* from \mathcal{O} if either it belongs to or it is a composition of operators in it.

Definition 2.5.12 (\mathcal{O} -Definability). We will say that a space \mathcal{S} (equivalently its structured sets) is *\mathcal{O} -definable* if it is obtained by transforming a trivial space \mathcal{S}' with a finite number of applications of operators which either are constructible from \mathcal{O} or are derived from \mathcal{O} -regular operators.

Definition 2.5.13 (Closed under regularity). A set \mathcal{O} of operators is said to be *closed under regularity* whenever:

- there is an operator of arity l in \mathcal{O} acting over S_1, \dots, S_l ;
- there are sets of points $s_1^1, s_1^2, \dots, s_l^1, s_l^2$ in $S_1^1, S_1^2, \dots, S_l^1, S_l^2$, respectively;
- there is another structured set S_r containing s_i^k ($i \in \{1, \dots, l\}, k \in \{1, 2\}$), but $s_i^k \in S_r^{k'}$ and $k \neq k'$; and
- there is a regular pair of operators constructively from \mathcal{O} , such that, one operator is of arity l acting over S_1, \dots, S_l , the embedded set in S_1 of this operator is in the same component than the embedded set in S_1 of the operator we started with, and the unary operator acts over S_r .

Example 2.5.14. We can construct, from the disjunction and negation rules in LK, the following rules

$$\frac{\Gamma \vdash \Delta, \xi_1 \quad \Gamma \vdash \Delta, \xi_2}{\neg \xi_1 \vee \neg \xi_2, \Gamma \vdash \Delta},$$

$$\frac{\xi_1, \xi_2, \Gamma \vdash \Delta}{\Gamma \vdash \Delta, \neg \xi_1 \vee \neg \xi_2},$$

which form a regular pair.

It is no hard to see that LK is closed under regularity. Also notice this framework of structured sets as well as its *regularity* condition generalizes regular schematic systems.

Chapter 3

Interpolation

The *Craig interpolation theorem* [14] states that given a derivation of $A_1 \rightarrow A_2$, where A_1 and A_2 are formulas of classical first order logic, there are a formula A_3 , called the interpolant, and derivations of $A_1 \rightarrow A_3$ and $A_3 \rightarrow A_2$ such that A_3 contains only those predicates and free variables occurring in both A_1 and A_2 .

In the first section of this Chapter we describe the Craig interpolation theorem for LK as first presented by Craig in [14], also some variants of interpolation are described. It is followed by the description of interpolation in the context of structured sets, schematic systems and restricted schematic systems.

Finally, in the last section of this chapter we present an analysis of the *complexity (number of symbols) of the interpolant* in the propositional case as it is exposed in [11]. It is shown that minimal interpolants of non-linear size must have proofs with certain precise structural properties.

3.1 Craig interpolation

We first describe interpolation and some of its variants for LK, LKe and G systems. Then we describe some applications of interpolation such as *Beth definability* [1] and *Robinson's joint consistency* [31].

In order to understand the combinatorial nature of the interpolation property we describe such property for *structured sets* as was first introduced by Carbone in [10]. Sufficient conditions for interpolation in the context of schematic systems were first introduced in [9]. We present a refinement of such conditions. Then we introduce a formulation of the Craig interpolation theorem for *restricted schematic systems*.

3.1.1 Basic results

First we present the main technique used to prove interpolation, which was first introduced by Maehara [23]. It is worthwhile to mention that interpolants are actually built from the assumed deductions.

Lemma 3.1.1 (Maehara's lemma [23]). *Let $\Gamma \vdash \Delta$ be LK-derivable, and let (Γ_1, Γ_2) and (Δ_1, Δ_2) be arbitrary partitions of Γ and Δ respectively. Then there exists a formula A_0 (called the interpolant) such that:*

- $\Gamma_1 \vdash \Delta_1, A_0$ and $A_0, \Gamma_2 \vdash \Delta_2$ are both LK-derivable.
- All free variables, constants and predicates in A_0 occur both in $\Gamma_1 \cup \Delta_1$ and $\Gamma_2 \cup \Delta_2$.

PROOF. The proof runs by induction on the height k , of the proof \mathcal{D} of $\Gamma \vdash \Delta$.

- ($k = 0$) Then $\Gamma \vdash \Delta$ has the form $A \vdash A$. Consider the cases when the partition $(\Gamma_1; \Delta_1; \Gamma_2; \Delta_2)$ of $(\Gamma; \Delta)$ is defined as
 1. $(A; A; ;)$, then we define the interpolant as $\neg \top$;
 2. $(; ; A; A)$, then we define the interpolant as \top ;
 3. $(A; ; ; A)$, then we define the interpolant as A ;
 4. $(; A; A;)$, then we define the interpolant as $\neg A$.
- ($k > 0$) Now, consider the following cases regarding the last inference

1.

$$\frac{\Gamma \vdash \Delta, A_1 \quad \Gamma \vdash \Delta, A_2}{\Gamma \vdash \Delta, A_1 \wedge A_2}.$$

Let $(\Gamma; \Delta, A_1 \wedge A_2)$ be partitioned in

$$(\Gamma_1; \Gamma_2; \Delta_1, A_1 \wedge A_2; \Delta_2).$$

By induction hypothesis there are deductions $\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3$ and \mathcal{D}_4 for

$$\begin{aligned} & \Gamma_1 \vdash \Delta_1, A_1, A'_1; \\ & A'_1, \Gamma_2 \vdash \Delta_2; \\ & \Gamma_1 \vdash \Delta_1, A_2, A'_2; \\ & A'_2, \Gamma_2 \vdash \Delta_2; \end{aligned}$$

respectively, where A'_1 and A'_2 are interpolants. If we consider $A'_1 \vee A'_2$ as the interpolant, then we obtain

$$\frac{\mathcal{D}_1 \quad \Gamma_1 \vdash \Delta_1, A_1, A'_1}{\Gamma_1 \vdash \Delta_1, A_1, A'_1 \vee A'_2}$$

$$\frac{\mathcal{D}_3 \quad \Gamma_1 \vdash \Delta_1, A_2, A'_2}{\Gamma_1 \vdash \Delta_1, A_2, A'_1 \vee A'_2}$$

then, clearly

$$\frac{\Gamma_1 \vdash \Delta_1, A_1, A_2, A'_1 \vee A'_2}{\Gamma_1 \vdash \Delta_1, A_1 \wedge A_2, A'_1 \vee A'_2}$$

and

$$\frac{\mathcal{D}_2 \quad \mathcal{D}_4 \quad A'_1, \Gamma_2 \vdash \Delta_2 \quad A'_2, \Gamma_2 \vdash \Delta_2}{A'_1 \vee A'_2, \Gamma_2 \vdash \Delta_2}$$

The cases for the other partitions of $(\Gamma; \Delta, A_1 \wedge A_2)$ run similarly, so we omit them.

2.

$$\frac{A(t), \Gamma \vdash \Delta}{\forall v_1 A(v_1), \Gamma \vdash \Delta}$$

Let $(\forall v_1 A(v_1), \Gamma; \Delta)$ be partitioned in

$$(\Gamma_1, \forall v_1 A(v_1); \Gamma_2; \Delta_1; \Delta_2).$$

By induction hypothesis we have deductions \mathcal{D}_1 and \mathcal{D}_2 for

$$\begin{aligned} &A(t), \Gamma_1 \vdash \Delta_1, A'(\vec{b}); \\ &A'(\vec{b}), \Gamma_2 \vdash \Delta_2; \end{aligned}$$

respectively, where $A'(\vec{b})$ is an interpolant and $\vec{b} = b_1, \dots, b_n$ are all the free variables and constants which occur in t .

Assume \vec{b}'' are the variables and constants in \vec{b} which do not occur in $A(v_1), \Delta_1, \Gamma_1$, and $\vec{b} = \vec{b}', \vec{b}''$. Now, consider the interpolant $\forall \vec{v}_2 A'(\vec{b}', \vec{v}_2)$, then we get

$$\frac{\mathcal{D}_1 \quad \frac{A(t), \Gamma_1 \vdash \Delta_1, A'(\vec{b}', \vec{b}'')}{A(t), \Gamma_1 \vdash \Delta_1, \forall \vec{v}_2 A'(\vec{b}', \vec{v}_2)}}{\forall v_1 A(v_1), \Gamma_1 \vdash \Delta_1, \forall \vec{v}_2 A'(\vec{b}', \vec{v}_2)}$$

and

$$\frac{\mathcal{D}_2}{\frac{A'(\vec{b}', \vec{b}''), \Gamma_2 \vdash \Delta_2}{\forall \vec{v}_2 A'(\vec{b}', \vec{v}_2), \Gamma_2 \vdash \Delta_2}}$$

The cases for the other partitions of $(\forall v_1 A(v_1), \Gamma; \Delta)$ run similarly, so we omit them.

3.

$$\frac{\Gamma \vdash \Delta, A(v_2)}{\Gamma \vdash \Delta, \forall v_1 A(v_1)}$$

Let $(\Gamma; \Delta, \forall v_1 A(v_1))$ be partitioned in

$$(\Gamma_1; \Gamma_2; \Delta_1, \forall v_1 A(v_1); \Delta_2).$$

By induction hypothesis we know that there are deductions D_1 and D_2 for

$$\begin{aligned} &\Gamma_1 \vdash \Delta_1, A(v_2), A'; \\ &A', \Gamma_2 \vdash \Delta_2; \end{aligned}$$

respectively, where A' is an interpolant. Then, clearly

$$\Gamma_1 \vdash \Delta_1, \forall v_1 A(v_1), A'$$

so, A' is also the new interpolant. The cases for the other partitions of $(\Gamma; \Delta, \forall v_1 A(v_1))$ run similarly, so we omit them.

The remaining cases run similarly. ■

Theorem 3.1.2 (Craig's interpolation [14]). *Consider that $\vdash A_1 \rightarrow A_2$ is LK-derivable. If A_1 and A_2 have a least one predicate in common, then there exists a formula A_3 , called the interpolant of $A_1 \rightarrow A_2$, such that A_3 contains only those constants, predicates and free variables that occur in both A_1 and A_2 , and such that $\vdash A_1 \rightarrow A_3$ and $\vdash A_3 \rightarrow A_2$ are both LK-derivable. If A_1 and A_2 have no predicate in common, then either $A_1 \vdash$ or $\vdash A_2$ is LK-derivable.*

PROOF. By Maehara's lemma we know there is an interpolant A_3 , such that $A_1 \vdash A_3$ and $A_3 \vdash A_2$ are derivable. There are two cases:

- Consider the case where A_1 and A_2 have predicates in common. Let P be a predicate in A_1 and A_2 , and let A'' be $\forall \vec{v}P(\vec{v})$. If we transform A_3 into A'_3 by replacing \top with $A'' \rightarrow A''$, we get the desired interpolant A'_3 .
- Consider the case when A_1 and A_2 have no predicates in common. Notice that A_3 is only composed by \top and logical symbols, hence either $\vdash A_3$ or $A_3 \vdash$ is derivable. Therefore, either $A_1 \vdash$ or $\vdash A_2$ is derivable.

■

Another direct consequence of Maehara's lemma is the consideration of positive and negative occurrences of predicate symbols in the interpolant. This property is called Lyndon's interpolation [22].

Definition 3.1.3. An occurrence π of an atomic formula in a regular formula A is inductively defined as follows

- if A is atomic, then π is positive,
- if π is positive in A , then it is also positive in $A \vee A'$ and $\forall vA$, but it is negative in $\neg A$.

If π is a positive/negative occurrence of A in A , then every variable, constant or function in A occurs positively/negatively in A .

Theorem 3.1.4 (Lyndon's interpolation [22]). *If $\vdash A \rightarrow B$ is LK-derivable, then there is an interpolant C , such that $\vdash A \rightarrow C$ and $\vdash C \rightarrow B$ are both LK-derivable and additionally, if a predicate appears positively/negatively in C then it also appears positively/negatively in A and B .*

Nagashima in [28] extended interpolation for languages with the equality symbol ($=$). Here it is described that result as in [40].

Theorem 3.1.5 (Interpolation for languages with equality [28]). *If $\Gamma_e \vdash A_1 \rightarrow A_2$ is LK-derivable, then there is an interpolant A_3 , such that $\Gamma_e \vdash A_1 \rightarrow A_3$ and $\Gamma_e \vdash A_3 \rightarrow A_2$ are both derivable.*

In order to extend the Maehara's lemma proof to get interpolation in languages with equality, we represent each function f by a new predicate $A(\vec{v}_1, v_2)$ such that $f(\vec{v}_1) = v_2$, and whenever two variables v_1 and v_2 $v_1 = v_2$, we introduce a new predicate $E(v_1, v_2)$. See [40] for a detailed proof.

Notice that it is not possible to extend Lyndon's interpolation Theorem for constants and variables in the presence of equality as it is shown by the sequent

$$\exists v(v = b \wedge \neg A(v)) \vdash \neg A(b),$$

where b occurs positively in the antecedent and negatively in the succedent.

Following Maehara's technique for proving interpolation for languages with equality, no essential changes are needed to state the interpolation property of G systems.

Theorem 3.1.6 (Interpolation for G systems [40]). *Consider a deductive system $G_{[123][mic]}$ with equality. If*

$$\Gamma_e, \Gamma \vdash_{G_{[123][mic]}} \Delta$$

then there is formula A_0 , called the interpolant, such that

- $\Gamma_e, \Gamma \vdash_{G_{[123][mic]}} A_0$;
- $\Gamma_e, A_0 \vdash_{G_{[123][mic]}} \Delta$;
- *every constant, free variable, predicate (except \perp) and function in A_0 also occurs in both Γ and Δ .*

3.1.2 Some applications

We write the symbol \rightarrow above logical symbols (e.g. $\vec{P}, \vec{v}, \vec{c}$) in order to represent a finite sequence of symbols of the same type. In order to denote the occurrence of predicate symbols P_1, \dots, P_n in a formula \mathcal{A} we write $\mathcal{A}(P_1, \dots, P_n)$.

Definition 3.1.7 (Implicit and explicit definitions). Consider

- the predicates P, P', \vec{P} ,
- the formula $\forall \vec{v}(P(\vec{v}) \equiv P'(\vec{v}))$, named A ,
- the formula $B(P, \vec{P}) \wedge B(P', \vec{P})$, named C .

Then,

- $B(P, \vec{P})$ defines P *implicitly* in terms of \vec{P} if

$$\vdash_{LK} C \rightarrow A,$$

and

- $B(P, \vec{P})$ defines P explicitly in terms of \vec{P} and the constants in $B(P, \vec{P})$ if there exists a formula $A_0(\vec{b})$ containing only the predicates \vec{P} and the constants in $B(P, \vec{P})$ such that

$$\vdash_{LK} B(P, \vec{P}) \rightarrow \forall \vec{v}(P(\vec{v}) \equiv A_0(\vec{v})).$$

Theorem 3.1.8 (Beth's definability [1]). *If P is implicitly defined in terms of \vec{P} by $B(P, \vec{P})$, then P is also explicitly defined in terms of \vec{P} and the constants in $B(P, \vec{P})$.*

Proof (sketch). Let \vec{v} be free variables not occurring in B . Then, by assumption

$$B(P, \vec{P}), B(P', \vec{P}) \vdash P(\vec{b}) \equiv P'(\vec{b})$$

and hence

$$B(P, \vec{P}) \wedge P(\vec{b}) \vdash B(P', \vec{P}) \rightarrow P'(\vec{b}). \quad (3.1)$$

Now, we only apply interpolation to 3.1. ■

Proposition 3.1.9 (Robinson's joint consistency [31]). *Assume that the language contains no function symbols. Let Φ_1 and Φ_2 be sets of formulas. Suppose that for any regular formula A which is dependent on Φ_1 and Φ_2 it is not the case that $\Phi_1 \vdash A$ and $\Phi_2 \vdash \neg A$ (or $\Phi_2 \vdash A$ and $\Phi_1 \vdash \neg A$) are both derivable in LK ($G[123][mic]$). Then $\Phi_1 \cup \Phi_2$ is consistent with respect to LK ($G[123][mic]$).*

Proof (sketch). Suppose $\Phi_1 \cup \Phi_2$ is not consistent. Then $\Phi_1, \Phi_2 \vdash$ is derivable. Since Φ_1 and Φ_2 are both consistent, neither Φ_1 nor Φ_2 is empty. Applying Maehara's lemma (3.1.1) we get deductions of $\Gamma_1 \vdash A$ and $A, \Gamma_2 \vdash$. Contradiction. ■

3.1.3 Structured sets

Theorem 3.1.10 (Interpolation for structured sets [10]). *Let $S = \langle A \mid B \rangle$ be a structured set obtained from some trivial space \mathcal{S}' by applying a finite number of transformations based on a set of operators \mathcal{O} which is closed under regularity. Then there are two structured sets $S^A = \langle A \mid I \rangle$, and*

$S^B = \langle I \mid B \rangle$ (or $\langle A, I \mid \rangle$, and $\langle I, B \mid \rangle$) (where I is called the interpolant set of A, B) which are both \mathcal{O} -definable. Moreover, if the regular operators are surjective operators and the trivial structured sets in \mathcal{S}' are defined by surjective embeddings, then for all points $z \in I$ there are points $x \in A$ and $y \in B$ such that z is mapped into x and into y by the transformations defining S^A and S^B , and such that the points x and y are mapped into each other by the transformations defining S .

If I does not contain any point and no structured set in \mathcal{S}' is trivial because of empty embeddings, then either $S^A = \langle A \mid \rangle$ or $S^B = \langle \mid B \rangle$ can be built from trivial spaces using only operators in \mathcal{O} .

Proof (sketch). The proof is by induction on the steps of transformation from the trivial space $\mathcal{S}_0 = \{S_{0,1}, \dots, S_{0,h}\}$ to the space $\mathcal{S} = \{S\}$. Let's say that \mathcal{S}_{p+1} is the space obtained from \mathcal{S}_p by the $p + 1$ -th step of transformation from \mathcal{S}_0 to \mathcal{S} . The technique used in this proof is to build the interpolant set from the partition spaces of \mathcal{S} . In particular, at step p

1. there are two spaces $\mathcal{S}_p^A = \{S_{p,1}^A, \dots, S_{p,i_r}^A\}$ and $\mathcal{S}_p^B = \{S_{p,1}^B, \dots, S_{p,i_l}^B\}$, where $l, r \leq k$ and
 - (a) if $p = 0$ then \mathcal{S}_0^A and \mathcal{S}_0^B are trivial, otherwise they are \mathcal{O} -definable,
 - (b) every member in \mathcal{S}_p^A , as well as the members in \mathcal{S}_p^B , is associated with an element in \mathcal{S}_p , and each member of \mathcal{S}_p has a counterpart, but no more than one, in \mathcal{S}_p^A or \mathcal{S}_p^B , and
 - (c) if $S_{p,i}$ (in \mathcal{S}_p) contains sets that are embedded into A , then $S_{p,i}$ is associated to a member of \mathcal{S}_p^A which contains these same sets, and in the same components; if $S_{p,i}$ is associated to a structured set in \mathcal{S}_p^A , then the later contains at most one other set of points beyond the ones just mentioned; the same holds for \mathcal{S}_p^B , and
2. the interpolant set I is build from some special sets (I -sets) such that
 - (a) if $S_{p,i}$ (in \mathcal{S}_p) is associated to members of $\mathcal{S}_p^A, \mathcal{S}_p^B$, respectively, then each of these sets has an I -set (both the same set) in opposite components; in case there is no such association, then there are no such I -sets, and
 - (b) if the regular operators are surjective and the sets in \mathcal{S}_0 are defined by surjective embeddings, for any S_{p,i_j}^A and S_{p,i_s}^B associated to the same $S_{p,i}$, and all points z in their I -set there are points x and y in some set of S_{p,i_j}^A and S_{p,i_s}^B , respectively, such that z is mapped

into x and y by those embeddings and their inverses defining \mathcal{S}_p^A and \mathcal{S}_p^B , and such that the points x and y are mapped into each other by the transformations defining S .

■

Now, in order to consider also the LJ system, it is necessary to restrict our definition of structured sets.

Definition 3.1.11 (Restricted structured sets). A *restricted structured set* is a structured set where its second component contains at most one set and copies of it.

Definition 3.1.12 (Restricted transformation). A transformation which involves only restricted structured sets is called a *restricted transformation*.

The interpolation theorem also holds when only restricted transformations are involved, see [10].

Example 3.1.13 (Interpolation for LJ). Consider the logical system LJ, then it is not hard to see that restricted structured sets can be interpreted as sequents in LJ, hence the interpolation property is a particular case of interpolation in restricted structured sets.

The key point to ensure interpolation in structured sets is to do the transformation using a set of operators closed under regularity. In the case for logical systems, this observation was used to prove interpolation in some fragments of linear and classical logic by Carbone [11]. It is worthwhile also to notice that the well known subformula property [37] is not required in the examples mentioned above in order to ensure interpolation, it is only necessary some kind of embedding, for example analytical cuts.

3.1.4 Regular schematic systems

Definition 3.1.14. The set $\mathcal{L}(\mathcal{A})$, where \mathcal{A} is a formula, is defined as follow:

1. $\mathcal{L}(\mathcal{A}(\mathfrak{t}_1, \dots, \mathfrak{t}_n)) = \{\mathcal{A}\}$, where \mathcal{A} is an atomic formula and $\mathfrak{t}_1, \dots, \mathfrak{t}_n$ are constant terms or variable terms occurring in \mathcal{A} .
2. $\mathcal{L}(c(\mathcal{A}_1, \dots, \mathcal{A}_n)) = \bigcup_{i=1}^n \mathcal{L}(\mathcal{A}_i)$, where c is a connective of arity n and $\mathcal{A}_1, \dots, \mathcal{A}_n$ are formulas.

3. $\mathcal{L}(qv(\mathcal{A})) = \mathcal{L}(\mathcal{A})$, where q is a quantifier, v is a variable or schema variable and \mathcal{A} is a formula.

Theorem 3.1.15 (Interpolation [9]). *Let \mathcal{Z} be a regular schematic system. Let A and B be two regular formulas such that the sequent $A \vdash B$ is derivable in \mathcal{Z} , and that A and B have at least one predicate in common. Then there exists a formula C containing only those predicates that occur in both A and B and possibly special predicates (\perp, \top), such that $A \vdash C$ and $C \vdash B$ are derivable in \mathcal{Z} .*

PROOF. Let \mathcal{D}_* be a derivation of $A \vdash B$, then the objective is to show that there are derivations \mathcal{D}_*^A and \mathcal{D}_*^B of $A \vdash C_*$ and $C_* \vdash B$, respectively, such that $\mathcal{L}(C_*) \subset (\mathcal{L}(A) \cap \mathcal{L}(B)) \cup \{\perp, \top\}$.

The proof is by induction on the height of the subtrees \mathcal{D} , of \mathcal{D}_* , with end-sequent $\Gamma \vdash \Delta$. We will show that there exist subcollections Γ^A, Γ^B and Δ^A, Δ^B of Γ and Δ , respectively, and a formula $C_{\mathcal{D}}$ such that

1. $\Gamma^A \vdash \Delta^A, C_{\mathcal{D}}$,
2. $C_{\mathcal{D}}, \Gamma^B \vdash \Delta^B$,
3. there is a path of a logical graph in \mathcal{D}_* from each formula occurrence in Γ^A, Δ^A and Γ^B, Δ^B to A and B , respectively, and
4. $\mathcal{L}(C_{\mathcal{D}}) \subset (\mathcal{L}(A) \cap \mathcal{L}(B)) \cup \{\perp, \top\}$.

For the base case suppose that \mathcal{D} is an axiom. Then we must consider three cases depending whether there are two distinguished occurrences or only one (as antecedent or succedent).

When \mathcal{D} has the form $D, \Theta \vdash \Lambda, D'$, then it must be taken into account where each of the occurrences D and D' go (A or B).

If D goes to A and D' goes to B , we define the tuple $(\Gamma^A, \Delta^A; C_{\mathcal{D}}; \Gamma^B, \Delta^B)$ as

$$(D, \Theta^A; \Lambda^A; D; \Theta^B; \Lambda^B, D'),$$

when $\mathcal{L}(D) \subset \mathcal{L}(D') \cup \{\perp, \top\}$, and

$$(D, \Theta^A; \Lambda^A; D'; \Theta^B; \Lambda^B, D'),$$

if $\mathcal{L}(D') \subset \mathcal{L}(D) \cup \{\perp, \top\}$.

If D goes to B , D' goes to A and $\mathcal{L}(D) \subset \mathcal{L}(D') \cup \{\perp, \top\}$, then we define the tuple $(\Gamma^A; \Delta^A; C_{\mathcal{D}}; \Gamma^B; \Delta^B)$ as

$$(\Theta^A; \Lambda^A, D'; D_1; D, \Theta^B; \Lambda^B),$$

such that there is \mathcal{Z} -definable (condition 3 of Definition 2.2.15) k -regular pair (for some $k > 0$)

$$\frac{\overbrace{D, \Gamma^A \vdash \Delta^A \quad \dots \quad \mathcal{S}_{k-1}}^{k \text{ sequents}}}{\Gamma^A \vdash \Delta^A, D_1}, \quad \frac{\Gamma^B \vdash \Delta^B, D}{D_1, \Gamma^B \vdash \Delta^B} \quad \text{or}$$

$$\frac{D, \Gamma^A \vdash \Delta^A}{\Gamma^A \vdash \Delta^A, D_1}, \quad \frac{\overbrace{\Gamma^B \vdash \Delta^B, D \quad \dots \quad \mathcal{S}'_{k-1}}^{k \text{ sequents}}}{D_1, \Gamma^B \vdash \Delta^B},$$

where each predicate occurring in D_1 also occurs in D .

In case D goes to B and D' goes to A , but $\mathcal{L}(D') \subset \mathcal{L}(D) \cup \{\perp, \top\}$, then we define $(\Gamma^A; \Delta^A; C_{\mathcal{D}}; \Gamma^B; \Delta^B)$, similarly to the previous case, as

$$(\Theta^A; \Lambda^A, D'; D'_1; D, \Theta^B; \Lambda^B).$$

If both D and D' go to A , we define $(\Gamma^A; \Delta^A; C_{\mathcal{D}}; \Gamma^B; \Delta^B)$ as

$$(D, \Theta^A; \Lambda^A, D'; \perp; \Theta^B; \Lambda^B),$$

if D and D' go to B it is defined as

$$(\Theta^A; \Lambda^A; \top; D, \Theta^B; \Lambda^B, D').$$

For the case when \mathcal{D} is of the form $\Theta \vdash \Lambda, D'$, then either D' goes to A or goes to B , for the first case define $(\Gamma^A; \Delta^A; C_{\mathcal{D}}; \Gamma^B; \Delta^B)$ as

$$(\Theta^A; \Lambda^A, D'; \perp; \Theta^B; \Lambda^B),$$

and for the second as

$$(\Theta^A; \Lambda^A; \top; \Theta^B; \Lambda^B, D').$$

If the form of \mathcal{D} is $D, \Theta \vdash \Lambda$, then define $(\Gamma^A; \Delta^A; C_{\mathcal{D}}; \Gamma^B; \Delta^B)$ as

$$(D, \Theta^A; \Lambda^A; \perp; \Theta^B; \Lambda^B)$$

when D goes to A , and

$$(\Theta^A; \Lambda^A; \top; D, \Theta^B; \Lambda^B)$$

when D goes to B .

For the case when the height of the subtree \mathcal{D} is greater than 1, suppose that the last rule of inference of \mathcal{D} has the form:

$$\frac{\mathcal{D}_1 \quad \dots \quad \mathcal{D}_k}{\mathcal{S}_0[A_0]} \frac{\mathcal{S}_1[A_{1,1}, \dots, A_{1,n_1}] \quad \dots \quad \mathcal{S}_k[A_{k,1}, \dots, A_{k,n_k}]}{(\alpha)}$$

where $\mathcal{D}_1, \dots, \mathcal{D}_k$ ($k \geq 1$) are subtrees. Then, by induction hypothesis there are interpolants $C_{\mathcal{D}_i}$ and derivations $\mathcal{D}_i^A, \mathcal{D}_i^B$ such that $\mathcal{L}(C_{\mathcal{D}_i}) \subset (\mathcal{L}(A) \cap \mathcal{L}(B)) \cup \{\perp, \top\}$ ($i = 1, \dots, k$).

We have to consider two cases according to whether A_0 goes to A or to B , and two subcases according to whether α contains or not a proviso over eigenvariables. In case A_0 goes to A and α contains a proviso let \mathcal{D}^A be the derivation

$$\frac{\mathcal{D}_1^A \quad \dots \quad \mathcal{D}_k^A}{\mathcal{S}_0^A[A_0, C_0]} \frac{\frac{\mathcal{S}_1^A[A_{1,1}, \dots, A_{1,n_1}, C_{\mathcal{D}_1}]}{\mathcal{S}_1^A[A_{1,1}, \dots, A_{1,n_1}, C'_{\mathcal{D}_1}]} (\eta) \quad \dots \quad \frac{\mathcal{S}_k^A[A_{k,1}, \dots, A_{k,n_k}, C_{\mathcal{D}_k}]}{\mathcal{S}_k^A[A_{k,1}, \dots, A_{k,n_k}, C'_{\mathcal{D}_k}]} (\eta)}{\mathcal{S}_0^A[A_0, C'_{\mathcal{D}_1}, \dots, C'_{\mathcal{D}_k}]} (\beta) \quad (\alpha)$$

where $C_{\mathcal{D}}$ is C_0 . Note the eigenvariables of $C_{\mathcal{D}_i}$ ($i = 1, \dots, k$) are not free in $C'_{\mathcal{D}_i}$ due to the application of (η) which exists in the calculus due to condition 2 of Definition 2.2.15. In case there are no eigenvariables in $C_{\mathcal{D}_i}$ we define \mathcal{D}^A by avoiding the (η) rule in the obvious way. (β) is ensured by condition 1 of Definition 2.2.15. Let \mathcal{D}^B be the derivation

$$\frac{\mathcal{D}_1^B \quad \dots \quad \mathcal{D}_k^B}{\mathcal{S}_0^B[C_0]} \frac{\frac{\mathcal{S}_1^B[C_{\mathcal{D}_1}]}{\mathcal{S}_1^B[C'_{\mathcal{D}_1}]} (\eta') \quad \dots \quad \frac{\mathcal{S}_k^B[C_{\mathcal{D}_k}]}{\mathcal{S}_k^B[C'_{\mathcal{D}_k}]} (\eta')}{(\beta')}$$

where (η') and (β') are the duals of (η) and (β) , respectively. In case $C_{\mathcal{D}_i}$ has no eigenvariables, \mathcal{D}^B is defined as expected.

In case A_0 goes to B and $C_{\mathcal{D}_i}$ contains eigenvariables of \mathcal{D} , let \mathcal{D}^A be the derivation

$$\frac{\mathcal{D}_1^A \quad \dots \quad \mathcal{D}_k^A}{\mathcal{S}_0^A[C_0]} \frac{\frac{\mathcal{S}_1^A[C_{\mathcal{D}_1}]}{\mathcal{S}_1^A[C'_{\mathcal{D}_1}]} (\eta') \quad \dots \quad \frac{\mathcal{S}_k^A[C_{\mathcal{D}_k}]}{\mathcal{S}_k^A[C'_{\mathcal{D}_k}]} (\eta')}{(\beta')}$$

and \mathcal{D}^B be

$$\frac{\frac{\mathcal{D}_1^B}{\mathcal{S}_1^B[A_{1,1}, \dots, A_{1,n_1}, C_{\mathcal{D}_1}]} (\eta) \quad \dots \quad \frac{\mathcal{D}_k^B}{\mathcal{S}_k^B[A_{k,1}, \dots, A_{k,n_k}, C_{\mathcal{D}_k}]} (\eta)}{\frac{\mathcal{S}_1^B[A_{1,1}, \dots, A_{1,n_1}, C'_{\mathcal{D}_1}]} (\alpha) \quad \dots \quad \frac{\mathcal{S}_k^B[A_{k,1}, \dots, A_{k,n_k}, C'_{\mathcal{D}_k}]} (\alpha)}{\frac{\mathcal{S}_0^B[A_0, C'_{\mathcal{D}_1}, \dots, C'_{\mathcal{D}_k}]} (\beta)}{\mathcal{S}_0^B[A_0, C_0]} (\beta)}$$

where $C_{\mathcal{D}}$ is C_0 . For the case when $C_{\mathcal{D}_i}$ has no eigenvariables \mathcal{D}^A and \mathcal{D}^B are defined in the obvious way. ■

3.1.5 Restricted regular schematic systems

In this section, we prove a different version of Craig interpolation in regular restricted schematic systems. This version differs from Craig interpolation, in the fact that given a deduction \mathcal{D} of $A \vdash B$, we present deductions of $A \vdash C$ and, depending on the structure of \mathcal{D} , either $C \vdash B$ or $C \vdash B''$, where C is an interpolant and, in case $C \vdash B''$, there is a deduction

$$\frac{\frac{B \vdash B}{B, B' \vdash}}{B \vdash B''}$$

The technique used to prove this version of interpolation relies on the conditions presented in Definition 2.2.15. This technique constructs the deductions \mathcal{D}^A and \mathcal{D}^B of $A \vdash C$ and either $C \vdash B$ or $C \vdash B''$, respectively, directly from \mathcal{D} . When constructing \mathcal{D}^A and \mathcal{D}^B , since in restricted schematic systems the sequents are allowed to have at most one formula in the succedent, we change of position the formulas appearing in the succedents of some sequents in \mathcal{D} to the antecedent, and the other way around. This process of "changing positions of formulas" is the cause why in some cases we obtain $C \vdash B''$ instead of $C \vdash B$. Although in regular schematic systems, B is logically equivalent with B'' , this is no longer the case for regular restricted schematic systems. It should be notice that following a different technique in the construction of \mathcal{D}^A and \mathcal{D}^B in order to get Craig interpolation in the traditional sense, a different notion of regular restricted schematic system would be necessary.

Theorem 3.1.16. *Let \mathcal{Z} be a regular restricted schematic system. If there is a derivation in \mathcal{Z} of $A \vdash B$, then there are also derivations in \mathcal{Z} of $A \vdash C$ and either $C \vdash B$ or $C \vdash B''$, such that*

- $\mathcal{L}(C) \subset (\mathcal{L}(A) \cap \mathcal{L}(B)) \cup \{\perp, \top\}$; and
- if $C \vdash B''$, then there is a deduction

$$\frac{\frac{B \vdash B}{B, B' \vdash}}{B \vdash B''}$$

PROOF. Consider a proof \mathcal{D}_* of $A \vdash B$. The proof runs by induction over the height of subtrees \mathcal{D} (of \mathcal{D}_*). Suppose \mathcal{D} has end-sequent $\Gamma \vdash \Delta$, where Δ can be either empty or be a singleton. If Δ is not empty, then we call E the formula in it. We will show that there are collections of formulas $\Gamma^A, \Gamma^B, \Delta^B$, and a formula $C_{\mathcal{D}}$ such that

1. $\Gamma^A \vdash C_{\mathcal{D}}$;
2. $C_{\mathcal{D}}, \Gamma^B \vdash \Delta^B$;
3. there is a path from each formula occurrence in Γ^A and Γ^B, Δ^B to A and B , respectively;
4. if Δ is empty, then Δ^B is empty;
5. if E goes to B , then
 - each formula in Γ that goes to A is in Γ^A ;
 - each formula in Γ that goes to B is in Γ^B ;
 - either E or a formula E'' is in Δ^B such that there is a deduction

$$\frac{\frac{E \vdash E}{E, E' \vdash}}{E \vdash E''}$$

6. if E goes to A , then there exist collections of formulas Γ_0^A and Γ_0^B that contains each formula in Γ that goes to A and B , respectively, and a formula $C_{\mathcal{D}_0}$, such that there are deductions whose last steps of inference, constructed from \mathcal{Z} -definable k -regular pairs whose existence is ensured by condition 3 of Definition 2.2.15, are

$$\frac{\overbrace{C_{\mathcal{D}_0}, \Gamma_0^A \vdash \Delta \quad \dots \quad \mathcal{S}_{k-1}^A}^{k \text{ sequents}}}{\Gamma^A \vdash C_{\mathcal{D}}}, \quad \frac{\Gamma_0^B \vdash C_{\mathcal{D}_0}}{C_{\mathcal{D}}, \Gamma^B \vdash \Delta^B} \text{ or}$$

$$\frac{C_{\mathcal{D}_0}, \Gamma_0^A \vdash \Delta}{\Gamma^A \vdash C_{\mathcal{D}}}, \quad \frac{\overbrace{\Gamma_0^B \vdash C_{\mathcal{D}_0} \quad \dots \quad S_{k-1}^B}^{k \text{ sequents}}}{C_{\mathcal{D}}, \Gamma^B \vdash \Delta^B},$$

where $\mathcal{L}(C_{\mathcal{D}}) \subset \mathcal{D}(C_{\mathcal{D}_0}) \cup \{\perp, \top\}$, $\mathcal{L}(\Gamma_0^A, \Delta) = \mathcal{L}(\Gamma^A)$ and $\mathcal{D}(\Gamma_0^B) = \mathcal{L}(\Gamma^B, \Delta^B)$, and

7. $\mathcal{L}(C_{\mathcal{D}}) \subset (\mathcal{L}(A) \cap \mathcal{L}(B)) \cup \{\perp, \top\}$.

Now, consider the base case.

Assuming \mathcal{D} has the form $D, \Theta \vdash D'$, where D and D' are the two distinguished occurrences, the occurrence of D goes to A , and the occurrence of D' goes to B , then the definition of $(\Gamma^A; C_{\mathcal{D}}; \Gamma^B; \Delta^B)$ is either

$$(D, \Theta^A; D; \Theta^B; D'),$$

when $\mathcal{L}(D) \subset \mathcal{L}(D') \cup \{\perp, \top\}$, or

$$(D, \Theta^A; D'; \Theta^B; D')$$

when $\mathcal{L}(D') \subset \mathcal{L}(D) \cup \{\perp, \top\}$.

Now, if we consider the case when D is going to B and D' is going to A , we define the tuple $(\Gamma_0^B; C_{\mathcal{D}_0}; \Gamma_0^A)$ as

$$(D, \Theta^B; D; \Theta^A)$$

when $\mathcal{L}(D) \subset \mathcal{L}(D') \cup \{\perp, \top\}$. When $\mathcal{L}(D') \subset \mathcal{L}(D) \cup \{\perp, \top\}$, $(\Gamma_0^B; C_{\mathcal{D}_0}; \Gamma_0^A)$ is

$$(D, \Theta^B; D'; \Theta^A).$$

Also, it could be the case that both D and D' go to A , in such a case we define the tuple $(\Gamma_0^B; C_{\mathcal{D}_0}; \Gamma_0^A)$ as

$$(\Theta^B; \top; D, \Theta^A).$$

When both D and D' go to B , then we define the tuple $(\Gamma^A; C_{\mathcal{D}}; \Gamma^B; \Delta^B)$ as

$$(\Theta^A; \top; D, \Theta^B; D').$$

Now, when \mathcal{D} is of the form $D, \Theta \vdash E$, where D is its only distinguished occurrence, we define $(\Gamma^A; C_{\mathcal{D}}; \Gamma^B; \Delta^B)$ as

$$(D, \Theta^A; \perp; \Theta^B, E)$$

in case D goes to A and E goes to B . When both D and E go to A , then $(\Gamma_0^B; C_{\mathcal{D}_0}; \Gamma_0^A)$ is

$$(\Theta^B; \top; D, \Theta^A).$$

If D goes to B then $(\Gamma^A; C_{\mathcal{D}}; \Gamma^B; \Delta^B)$ is

$$(\Theta^A; \top; D, \Theta^B; E)$$

when E goes to B , and $(\Gamma_0^B; C_{\mathcal{D}_0}; \Gamma_0^A)$ is

$$(D, \Theta^B; \perp; \Theta^A)$$

when E goes to A .

When \mathcal{D} has the form $\Theta \vdash E$ where E is the only distinguished occurrence define $(\Gamma^A; C_{\mathcal{D}}; \Gamma^B; \Delta^B)$ as

$$(\Theta^A; \top; \Theta^B; E)$$

when E goes to B , and $(\Gamma_0^B; C_{\mathcal{D}_0}; \Gamma_0^A)$ as

$$(\Theta^B; \top; \Theta^A),$$

when E goes to A .

If \mathcal{D} has the form $D, \Theta \vdash$, then we define $(\Gamma^A; C_{\mathcal{D}}; \Gamma^B)$ as

$$(D, \Theta^A; \perp; \Theta^B)$$

when D goes to A , and

$$(\Theta^A; \top; D, \Theta^B)$$

when D goes to B .

Assuming the height of \mathcal{D} is greater than 1. Suppose it has the form

$$\frac{\mathcal{D}_1 \quad \dots \quad \mathcal{D}_k}{\mathcal{S}_0[A_0]} (\alpha)$$

We must distinguish two main cases, whether A_0 goes to A or to B . First, we consider the case when A_0 goes to A .

If the succedent of \mathcal{S}_0 is not empty, then we denote the formula there as E_0 .

The premises \mathcal{S}_i ($i = 1, \dots, k$) can have the following forms:

1. $\Gamma_i^A, A_{i,1}, \dots, A_{i,n_i}, \Gamma_i^B \vdash E_0$, or
2. $\Gamma_i^A, A_{i,1}, \dots, A_{i,n_i}, \Gamma_i^B \vdash$, or
3. $\Gamma_i^A, A_{i,1}, \dots, A_{i,n_i}, \Gamma_i^B \vdash A_{i,r}$.

If E_0 is absent, then only the last two forms are possible.

For each $i = 1, \dots, k$, by induction hypothesis we have formulas $C_{\mathcal{D}_i}$ whose predicates occur in both A and B , and derivations \mathcal{D}_i^A and \mathcal{D}_i^B with end-sequents \mathcal{S}_i^A and \mathcal{S}_i^B , such that:

- if E_0 goes to B , then \mathcal{S}_i^A and \mathcal{S}_i^B have the following forms:
 1. $\Gamma_i^A, A_{i,1}, \dots, A_{i,n_i} \vdash C_{\mathcal{D}_i}$ and $C_{\mathcal{D}_i}, \Gamma_i^B \vdash E_0$, or
 2. $\Gamma_i^A, A_{i,1}, \dots, A_{i,n_i} \vdash C_{\mathcal{D}_i}$ and $C_{\mathcal{D}_i}, \Gamma_i^B \vdash E_0''$, where $E_0 \vdash E_0''$, or
 3. $\Gamma_i^A, A_{i,1}, \dots, A_{i,n_i} \vdash C_{\mathcal{D}_i}$ and $C_{\mathcal{D}_i}, \Gamma_i^B \vdash$, or
 4. $C_{\mathcal{D}_i}, \Gamma_i^A, A_{i,1}, \dots, A_{i,n_i} \vdash A_{i,r}$ and $\Gamma_i^B \vdash C_{\mathcal{D}_i}$,
- if E_0 goes to A , then \mathcal{S}_i^A and \mathcal{S}_i^B have the following forms:
 1. $C_{\mathcal{D}_i}, \Gamma_i^A, A_{i,1}, \dots, A_{i,n_i} \vdash E_0$ and $\Gamma_i^B \vdash C_{\mathcal{D}_i}$, or
 2. $\Gamma_i^A, A_{i,1}, \dots, A_{i,n_i} \vdash C_{\mathcal{D}_i}$ and $C_{\mathcal{D}_i}, \Gamma_i^B \vdash$, or
 3. $C_{\mathcal{D}_i}, \Gamma_i^A, A_{i,1}, \dots, A_{i,n_i} \vdash A_{i,r}$ and $\Gamma_i^B \vdash C_{\mathcal{D}_i}$.

Consider the case when E_0 goes to B , (α) is unary and contains an eigenvariable proviso. Then let $\mathcal{D}_{1,*}^A$ be the derivation

$$\frac{\mathcal{D}_1^A}{\frac{\mathcal{S}_1^A[C_{\mathcal{D}_1}]}{\mathcal{S}_1^A[C_{\mathcal{D}_1}]} (\beta)}$$

where the (β) rule, whose existence is ensured by condition 2 of Definition 2.2.15, is used to eliminate all the eigenvariables occurring in $C_{\mathcal{D}_1}$. In case (α) contains no eigenvariable proviso, then we avoid (β) , define $\mathcal{D}_{1,*}^A$ simply as \mathcal{D}_1^A . Now, consider the derivation \mathcal{D}^A

$$\frac{\mathcal{D}_{1,*}^A}{\frac{\mathcal{S}_1^A[A_{1,1}, \dots, A_{1,n_1}]}{\mathcal{S}_1^A[A_0]} (\alpha)}$$

Now, let \mathcal{D}^B be the derivation

$$\frac{\mathcal{D}_1^B}{\frac{\mathcal{S}_1^B[C_{\mathcal{D}_1}]}{\mathcal{S}_1^B[C_{\mathcal{D}_1}]} (\beta')}$$

where (β') is the dual of (β) . In case (α) has no eigenvariable proviso, \mathcal{D}^B is defined as expected. Hence, $C_{\mathcal{D}}$ is defined as $C'_{\mathcal{D}_1}$.

Consider the case when (α) is not unary. For each $i = 1, \dots, k$, if $C_{\mathcal{D}_i}$ is in the succedent of \mathcal{S}_i^A and (α) contains a eigenvariable proviso, then consider $\mathcal{D}_{i,*}^A$ as

$$\frac{\mathcal{D}_i^A}{\frac{\mathcal{S}_i^A[C_{\mathcal{D}_i}]}{\frac{\mathcal{S}_i^A[C'_{\mathcal{D}_i}]}{\mathcal{S}_i^A[C''_{\mathcal{D}_i}]}}(\beta)} \cdots \frac{\mathcal{D}_{i,k-1}}{\mathcal{S}_{i,k-1}}(\eta) \quad \text{or} \quad \frac{\mathcal{D}_i^A}{\frac{\mathcal{S}_i^A[C_{\mathcal{D}_i}]}{\frac{\mathcal{S}_i^A[C'_{\mathcal{D}_i}]}{\mathcal{S}_i^A[C''_{\mathcal{D}_i}]}}(\beta)}(\eta)$$

The (η) rule is used to change the position of $C_{\mathcal{D}_i}$, i.e., $C'_{\mathcal{D}_i}$ is in the antecedent and contains each predicate occurring in $C_{\mathcal{D}_i}$. The existence of (η) is ensured by condition 3 of Definition 2.2.15. The (β) rule is used to eliminate all the eigenvariables occurring in $C_{\mathcal{D}_i}$. If $C_{\mathcal{D}_i}$ is in the antecedent or (α) has no eigenvariable proviso, then we do not need to apply (η) or (β) , respectively, and $\mathcal{D}_{i,*}^A$ is defined as expected.

Now consider the derivation \mathcal{D}_*^A

$$\frac{\mathcal{D}_{1,*}^A \quad \cdots \quad \mathcal{D}_{k,*}^A}{\frac{\mathcal{S}_1^A[A_{1,1}, \dots, A_{1,n_1}, C''_{\mathcal{D}_1}] \quad \cdots \quad \mathcal{S}_k^A[A_{k,1}, \dots, A_{k,n_k}, C''_{\mathcal{D}_k}]}{\mathcal{S}_0^A[A_0, C''_{\mathcal{D}_1}, \dots, C''_{\mathcal{D}_k}]}(\alpha)}(\zeta)$$

where $C_{\mathcal{D}_i}$ ($i = 1, \dots, k$) are side formulas in (α) , and A_0 is side in (ζ) . By the first condition in Definition 2.2.15, (ζ) exists.

In the sequel, we assume both (η) and its dual (η') are unary. The generalization to arity $k > 1$ is trivial.

Notice that if E_0 goes to B , then \mathcal{S}_0^A contains no formula in the succedent, then we define \mathcal{D}^A as

$$\frac{\mathcal{D}_*^A}{\frac{\mathcal{S}_0^A[C_0]}{\mathcal{S}_0^A[C'_0]}} \eta'$$

If E_0 goes to A , then it is in the succedent of \mathcal{S}_0^A . In such case we define \mathcal{D}^A as

$$\frac{\mathcal{D}_*^A}{\frac{\mathcal{S}_0^A[C_0, E_0]}{\mathcal{S}_0^A[C_0, E'_0]}(\eta)}(\eta')$$

where C_0 is a side formula in (η) and E'_0 is a side formula in (η') .

In the sequel we will write E_* in order to denote E_0 or E'_0 .

When E_* is in the succedent of \mathcal{S}_i^B , $C_{\mathcal{D}_i}$ is in the antecedent, and (α) contains a eigenvariable proviso, let $\mathcal{D}_{i,*}^B$ be the derivation

$$\begin{array}{c} \mathcal{D}_i^B \\ \frac{\mathcal{S}_i^B[C_{\mathcal{D}_i}, E_*]}{\mathcal{S}_i^B[C_{\mathcal{D}_i}, E'_*]} (\eta) \\ \frac{\mathcal{S}_i^B[C_{\mathcal{D}_i}, E'_*]}{\mathcal{S}_i^B[C'_{\mathcal{D}_i}, E'_*]} (\eta') \\ \frac{\mathcal{S}_i^B[C'_{\mathcal{D}_i}, E'_*]}{\mathcal{S}_i^B[C''_{\mathcal{D}_i}, E'_*]} (\beta') \end{array}$$

In case E_* does not occur in \mathcal{S}_i^B , $C_{\mathcal{D}_i}$ is already in the succedent or (α) does not have a eigenvariable proviso, $\mathcal{D}_{i,*}^B$ is defined as expected, respectively.

Consider the derivation \mathcal{D}_*^B

$$\frac{\mathcal{D}_{1,*}^B \quad \cdots \quad \mathcal{D}_{k,*}^B}{\mathcal{S}_0^B[C_0]} (\zeta')$$

where (ζ') is the dual of (ζ) .

If E_0 goes to B , notice E'_* is in the antecedent of \mathcal{S}_0^B , then we define \mathcal{D}^B as

$$\begin{array}{c} \mathcal{D}_*^B \\ \frac{\mathcal{S}_0^B[C_0, E'_*]}{\mathcal{S}_0^B[C'_0, E'_*]} (\eta) \\ \frac{\mathcal{S}_0^B[C'_0, E'_*]}{\mathcal{S}_0^B[C'_0, E''_*]} (\eta') \end{array}$$

where E'_* is a side formula in (η) , and C'_0 is side in (η') . If E_0 goes to A or is absent in \mathcal{S}_0 , we define \mathcal{D}^B as

$$\frac{\mathcal{D}_*^B}{\mathcal{S}_0^B[C'_0]} (\eta)$$

Therefore $C_{\mathcal{D}}$ is C'_0 .

Finally, if A_0 goes to B , the definitions of $C_{\mathcal{D}_i}$, \mathcal{D}^A and \mathcal{D}^B are analogous to the case when A_0 goes to B . ■

3.2 Interpolant complexity

In the sequel we consider a system resulting from LK by avoiding the rules that involve quantifiers. This system is for classical propositional logic. Consider a language $L \subset \{0, 1\}^*$ and $L \in NP \cap co-NP$. Hence the complement \bar{L} of L is in NP . Let's consider $A_n(p_1, \dots, p_n, q_1, \dots, q_m)$ and $B_n(p_1, \dots, p_n, r_1, \dots, r_s)$ as two propositional formulas where p_i, q_j, r_k ($1 \leq i \leq n, 1 \leq j \leq m, 1 \leq k \leq s$) are the only propositional symbols occurring in A_n and B_n , respectively. We will refer to the number of symbols occurring in a formula A as $|A|$. Now, it is well-known [27] that there are formulas A_n and B_n such that $|A_n|, |B_n| \leq n^{\mathcal{O}(1)}$ and for all $n \in \mathbb{N}$

$$\begin{aligned} L &= \{(p_1, \dots, p_n) \in \{0, 1\}^n \mid A_n(p_1, \dots, p_n, y_1, \dots, y_m) \text{ holds} \\ &\quad \text{for some } (y_1, \dots, y_m) \in \{0, 1\}^m\}, \\ \bar{L} &= \{(p_1, \dots, p_n) \in \{0, 1\}^n \mid B_n(p_1, \dots, p_n, z_1, \dots, z_s) \text{ holds} \\ &\quad \text{for some } (z_1, \dots, z_s) \in \{0, 1\}^s\}. \end{aligned}$$

Then, for all n we have a derivation \mathcal{D} of $A_n \vdash \neg B_n$, and by Craig interpolation there is an interpolant I_n of \mathcal{D} . Now, if there is a function f computable in polynomial time such that $|I_n| \leq f(|A_n| + |B_n|)$ for all n , then $L \in P/poly$. In other words, if the interpolant complexity of a sequent $A \vdash B$ is polynomially bounded by the size of the derivation of $A \vdash B$, then $NP \cap co-NP \subset P/poly$.

In this section we show, as described in [11], that the interpolant complexity of a sequent $A \rightarrow B$ in a deductive system for classical propositional logic depends on non trivial structures of A and B .

Corollaries 3.2.1, 3.2.3, 3.2.4, 3.2.5 and 3.2.7 are direct consequences of Theorem 3.1.15.

Corollary 3.2.1 (Propositional interpolation [11]). *Let \mathcal{D} be a derivation of $A \vdash B$. If A and B have at least one propositional symbol in common, then there exist a formula C containing only those propositional symbols that occur in both A and B , such that*

1. *there exist derivations \mathcal{D}^A and \mathcal{D}^B of $A \vdash C$ and $C \vdash B$, respectively;*
2. *if p occurs in C then there is a p^A in A and a p^B in B such that \mathcal{D}^A (\mathcal{D}^B) contains a logical path between p and p^A (p^B) and \mathcal{D} contains a logical path between p^A and p^B ;*

3. if \mathcal{D}^A (\mathcal{D}^B) contains a logical path between two occurrences p^1 and p^2 in A (B) then \mathcal{D} contains a logical path between p^1 and p^2 .

If A and B have no propositional symbol in common, then either $A \vdash$ or $\vdash B$ is derivable.

Definition 3.2.2 (Bridge). Any logical path (Definition 2.4.4), of a derivation with end-sequent \mathcal{S} , starting and ending with two distinct s-formulas occurring in \mathcal{S} is a *bridge*.

Consider \mathcal{D}_* as a derivation of $A \vdash B$ in the logical system for classical propositional logic, C an interpolant of \mathcal{D}_* , and \mathcal{D}^A and \mathcal{D}^B derivations of $A \vdash C$ and $C \vdash B$, respectively.

Corollary 3.2.3 (Inclusion-Exclusion Formula). *The number of axioms in \mathcal{D}_* is exactly the sum of the number of axioms in \mathcal{D}^A and the number of axioms in \mathcal{D}^B , minus the number of axioms where bridges connecting A and B pass through.*

Proof (sketch). The number of axioms in \mathcal{D}^A (\mathcal{D}^B) is the number of bridges from A to C (C to B) plus the number of bridges from A to A (B to B), and the number of bridges from A to C (C to B) is indeed the number of bridges from A to B in \mathcal{D}_* . ■

Corollary 3.2.4. *If the length of \mathcal{D}_* is k then the length of \mathcal{D}^A and \mathcal{D}^B is at most $3k$.*

Corollary 3.2.5. *There are no contraction rules in $\mathcal{D}^A, \mathcal{D}^B$ where the direct paths from C pass through.*

Definition 3.2.6 (Size). The size of a formula \mathcal{A} is the number of symbols occurring in it, and we write it as $|\mathcal{A}|$. The size of a sequent is the sum of the sizes of the formulas occurring in it. The size of a derivation is the sum of the sizes of the sequents occurring in it.

We write $\#lines(\mathcal{D})$ in order to denote the height of a derivation \mathcal{D} .

Corollary 3.2.7. *The size of the interpolant C is exactly the number of axioms in \mathcal{D}_* where bridges connecting A to B pass through.*

Proof (sketch). From 3.2.4 and 3.2.5. ■

Notice the relation of an interpolant and a cut-free derivation of a tautology $A \rightarrow B$ suggested by Corollaries 3.2.1 and 3.2.3. The following results¹ suggest that the complexity of the interpolant depends on the *logical relations* of the s-formulas in A and B .

Definition 3.2.8. Let f be a logical graph of a derivation \mathcal{D} . If all s-formulas in \mathcal{D} are connected to the end-sequent, we say f is *compact*. A subgraph f' of f is said to be compact if any variant in f connected to a variant in f' belongs to f' , any edge in f connecting two variants in f' is in f' , and any variant in f' has a connected variant in the end-sequent of \mathcal{D} .

Due to the inversion property [11] we will assume the logical graphs of derivations to be compact.

Lemma 3.2.9. *Let \mathcal{D} be a derivation of $A \vdash B$. Then, there is a derivation \mathcal{D}' of $A \vdash B$ whose logical graph is compact and $\#lines(\mathcal{D}') \leq \#lines(\mathcal{D})$. If \mathcal{D} does not have any bridge between A and B , \mathcal{D}' does not have any bridge between A and B either.*

Definition 3.2.10. Consider a derivation \mathcal{D} of $A \vdash B$, those axioms in \mathcal{D} whose distinguished formulas are connected to variants in A (B) are named A -axioms (B -axioms). In case the distinguished formulas of the axioms are connected to both A and B , then we define those axioms as AB -axioms.

Corollary 3.2.11. *Consider a derivation \mathcal{D} of $A \vdash B$, then there is also a derivation \mathcal{D}' of $A \vdash B$ whose axioms are either A -axioms or B -axioms or AB -axioms, and $\#lines(\mathcal{D}') \leq \#lines(\mathcal{D})$. If there are no bridges between A and B , then axioms in \mathcal{D}' are either A -axioms or B -axioms.*

Definition 3.2.12. Consider a derivation \mathcal{D} with a logical graph f . Let f' be a subgraph of f , then the *forgetful* map $H_{f'}$ from occurrences of formulas in

¹For detailed proofs please see [11]

\mathcal{D} to formulas (possibly none) is inductively defined

$$H_{f'}(A) = \begin{cases} A & \text{if } A \text{ is atomic and a node in } f' \\ \emptyset & \text{otherwise} \end{cases} \quad (3.2)$$

$$H_{f'}(\neg A) = \begin{cases} \neg H_{f'}(A) & \text{if } H_{f'}(A) \neq \emptyset \\ \emptyset & \text{otherwise} \end{cases} \quad (3.3)$$

$$H_{f'}(\mathcal{O}(A, B)) = \begin{cases} \mathcal{O}(H_{f'}(A), H_{f'}(B)) & \text{if } H_{f'}(A), H_{f'}(B) \neq \emptyset \\ H_{f'}(A) & \text{if } H_{f'}(B) = \emptyset \\ H_{f'}(B) & \text{if } H_{f'}(A) = \emptyset \\ \emptyset & \text{if } H_{f'}(A), H_{f'}(B) = \emptyset \end{cases} \quad (3.4)$$

where \mathcal{O} is either \vee or \wedge .

Definition 3.2.13 (Weak formulas). Consider a derived axiom $A, \Gamma \vdash \Delta, A$. Formulas occurring in Γ, Δ are called *weak formulas*. Consider a deduction \mathcal{D} of $A \vdash B$, then a weak formula C in \mathcal{D} is said to be *A-weak (B-weak)* if each *s-formula* in C has at least one connected variant in A (B).

Definition 3.2.14 (Base). Consider a logical graph f in a derivation of

$$A_1, \dots, A_m \vdash B_1, \dots, B_n$$

Let f' be a subgraph of f , then a *base* of f' is defined as

$$H_{f'}(A_1, \dots, A_m \vdash B_1, \dots, B_n) = \\ H_{f'}(A_1), \dots, H_{f'}(A_m) \vdash H_{f'}(B_1), \dots, H_{f'}(B_n).$$

Consider a formula occurrence A in \mathcal{S} , if $H_{f'}(A) = \emptyset$ then A does not belong to the base of \mathcal{S} induced by f' .

Theorem 3.2.15. *Consider a derivation \mathcal{D} of $A \vdash B$ with no bridges between A and B . If all *s-formulas* occurring in weak formulas of \mathcal{D} and lying in the compact subgraph with base $A \vdash (\vdash B)$, occur in A -axioms (B -axioms), then there is a derivation \mathcal{D}' of $A \vdash (\vdash B)$ such that $\#lines(\mathcal{D}') \leq \#lines(\mathcal{D})$.*

In other other words, this theorem states that there is a derivation of $A \vdash$ or $\vdash B$ complexity bounded for the original derivation when $\mathcal{L}(A) \cap \mathcal{L}(B) = \emptyset$ and weak formulas are properly distributed in the original derivation.

Corollary 3.2.16. *Consider a derivation \mathcal{D} of $A \vdash B$ with no bridges between A and B . If all weak occurrences in \mathcal{D} have either direct paths to A or to B and they are used as auxiliary formulas only by unary rules, then there exists a derivation \mathcal{D}' of $A \vdash$ or $\vdash B$ such that $\#lines(\mathcal{D}') \leq \#lines(\mathcal{D})$.*

Proof (sketch). If the weak occurrences of a derivation go to either A or B , we can rearrange the weak formulas in order to find a derivation of the same height whose weak occurrences directly linked to A (B) lie in $A - axioms$ ($B - axioms$), and then use theorem 3.2.15. \blacksquare

Theorem 3.2.17. *Consider a derivation \mathcal{D} of*

$$A(p_1, \dots, p_n, q_1, \dots, q_m) \rightarrow B(p_1, \dots, p_n, r_1, \dots, r_s),$$

where p_i, q_j, r_k ($1 \leq i \leq n, 1 \leq j \leq m, 1 \leq k \leq s$) are the occurrences of propositional variables in A and B , respectively. If $\#lines(\mathcal{D}) = k$ and A -weak (B -weak) formulas only occur in $A - axioms$ ($B - axioms$), then there is a derivation \mathcal{D}' of $A'(q_1, \dots, q_m) \vdash (\vdash B'(r_1, \dots, r_s))$, where A' (B') is obtained from A (B) by replacing p_i ($i = 1, \dots, n$) for either \perp or \top , such that $\#lines(\mathcal{D}') \leq k + 2|A| + 1$ ($\#lines(\mathcal{D}') \leq k + 2|B| + 1$).

Definition 3.2.18 (Variation). A variation \mathcal{B}^* of a formula \mathcal{B} is obtained by substituting in \mathcal{B} none, one or more of the s-formulas occurring in it with \perp or \top as follow

- if \mathcal{C} is an s-formula that occurs positively in \mathcal{B} then \mathcal{C} is replaced by \top ;
- if \mathcal{C} is an s-formula that occurs negatively in \mathcal{B} then \mathcal{C} is replace by \perp .

$^*\mathcal{B}$ is obtained from \mathcal{B}^* by replacing every occurrence of \top and \perp by \perp and \top , respectively.

In order to proof Theorem 3.2.22, we present Lemmas 3.2.19, 3.2.20, and 3.2.21 without proving them. Please see [11] for the proofs.

Lemma 3.2.19. *For all variations A^* of a regular formula A , $A \vdash A^*$ and $^*A \vdash A$ are derivable in no longer than $2|A|$ lines of derivation.*

Lemma 3.2.20. *Consider the variation A^* of a regular formula A made up of \perp, \top and logical symbols only, then $\vdash A^*$ and $A^* \vdash$ are derivable in $|A|$ steps.*

Lemma 3.2.21. *Consider a derivation \mathcal{D} of \mathcal{S} and a s-formula A in it. Then either there is a bridge between A and a variant of it in \mathcal{S} or there is a derivation \mathcal{D}' of \mathcal{S}' where \mathcal{S}' is obtained by replacing A with $\top(\perp)$ and $\#lines(\mathcal{D}') \leq \#lines(\mathcal{D})$.*

Finally, we present the main result of this section (Theorem 3.2.22), it says that when A (respectively, B) does not contain pairs of subformulas linked by a bridge one to the other, then there is an interpolant with size linearly bounded by the size of A (respectively, B).

Theorem 3.2.22 (Interpolants polynomially bounded [11]). *Let \mathcal{D} be a derivation of $A \vdash B$ with no bridges from A back to A (B back to B , respectively). If there is a bridge between A and B , then there is an interpolant C for $A \vdash B$ such that*

1. $|C| \leq 4|A|$ ($|C| \leq 4|B|$, respectively), and
2. $\#lines(\mathcal{D}^A) \leq 5|A|$ and $\#lines(\mathcal{D}^B) \leq 3\#lines(\mathcal{D})$, where \mathcal{D}^A and \mathcal{D}^B are derivations of $A \vdash C$ and $C \vdash B$, respectively (and symmetrically in the other case).

If there is no bridge between A and B , then either there is a derivation \mathcal{D}' of $A \vdash$ or $\vdash B$ such that $\#lines(\mathcal{D}') \leq \#lines(\mathcal{D})$.

Proof (sketch). Consider the case when there is bridge between A and B , and R is a propositional variable present in both A and B .

From Lemma 3.2.21 we can have a derivation \mathcal{D}' of $A^* \vdash B$ from \mathcal{D} such that A^* is obtained by replacing the s-formulas occurring positively in A by \top and the s-formulas occurring negatively by \perp . Now, please notice $A \vdash A^*$ (Lemma 3.2.20). Consider a formula C obtained from A^* by replacing the symbols \perp and \top by $R \wedge \neg R$ and $R \vee \neg R$, respectively. Since $|A| = |A^*|$, the axioms of the form $\Gamma \vdash \Delta, \top$ and $\top, \Gamma \vdash \Delta$ must be replaced in the deduction by 3 lines corresponding to the application of 2 additional steps of inference \vee , and by Lemma 3.2.20, $|C| \leq 4|A|$, $\#lines(\mathcal{D}^A) \leq 5|A|$, and $\#lines(\mathcal{D}^B) \leq 3\#lines(\mathcal{D})$.

When there are no bridges between A and B , then all s-formulas in A^* are weak. Since the weak occurrences and \perp, \top can be properly eliminated without increasing the length of the derivation, then we get $\vdash B$. ■

The following Corollary suggest that when investigating the complexity of the interpolant we should not consider standard quantities, like the height or number of symbols of a proof, but instead look at measurement of structure, such as the number of bridges from A to A and from B to B .

Corollary 3.2.23. *Consider a tautology $A \vdash B$ of size $m \geq 5$, and suppose that the minimal interpolant of $A \vdash B$ has size at least m^2 , then all derivations of $A \vdash B$ must contain bridges from A back to A , and bridges from B back to B .*

Chapter 4

Conclusion

In this chapter we summarise the work introduced in this dissertation and also we pointed out some further directions of investigation.

4.1 Results

A refinement of the sufficient conditions for ensure interpolation in schematic systems, originally introduced by Carbone in [9], is established. It differs from the Carbone approach by the introduction of Condition 3 in Definition 2.2.15. We now motivate the condition:

Consider a deduction \mathcal{D} of $A \vdash B$, where one of its axioms is

$$D, \Gamma \vdash \Delta, D'$$

such that there is path of a logical graph in \mathcal{D} from D to B , another one from D' to A , and $\mathcal{L}(D) \subset \mathcal{L}(D') \cup \{\perp, \top\}$. In other words, there are two Z -definable rules such that

$$\frac{D, \Gamma^A \vdash \Delta^A, D' \quad \dots \quad \mathcal{S}_{k-1}}{\Gamma_1^A \vdash \Delta_1^A, D_1} \quad \frac{D, \Gamma^B \vdash \Delta^B, D' \quad \dots \quad \mathcal{S}'_{k'-1}}{D'_1, \Gamma_1^B \vdash \Delta_1^B}$$

where Γ^A, Γ^B and Δ^A, Δ^B are subcollections of Γ and Δ , respectively, with a path of a logical graph from each of its occurrences to A and B , respectively, and $\mathcal{S}_i, \mathcal{S}'_j$ ($i = 1, \dots, k-1$ and $j = 1, \dots, k'-1$) are axioms. Notice there is no guarantee of $\mathcal{L}(D_1) = \mathcal{L}(D'_1)$. The motivation of condition 3 of Definition 2.2.15 is precisely to guarantee that $\mathcal{L}(D_1) = \mathcal{L}(D'_1)$, and in consequence to be the desired interpolant.

A proof for interpolation in schematic systems by means of logical graphs was also presented by Carbone in [9] where it is pointed out as further investigation the interpolation property of intuitionistic schematic systems. In this Thesis, an intuitionistic deductive systems in the context of schematic systems is described and also it is proved its interpolation property by means of logical graphs.

4.2 Further directions

Girard was the first in borrow ideas of geometry and topology to bring new insights in the nature of proofs [19]. The idea of using the flow of occurrences to study the structure of proofs is fundamental in the work of Girard where graphs called proof nets are associates to linear logic proofs. Following this direction, Guglielmi introduced a new methodology [21], called deep inference, in order to find cut-free analytical systems in some logics that traditional formalisms were unable to do [21], actually, it results that the formalism derived from this new methodology have provided new proof-theoretical properties, that have been to date not expressible in traditional formalisms [39]. The calculus of structures is the simplest formalism in the scope of deep inference that generalizes most traditional formalisms, in particular the sequent calculus, and besides its success in the development of cut-free systems for "problematic" logics, there are already important results in another logics, such as, linear logic [34, 35, 36], modal logics [5, 33], and Yetter's non-commutative logic [18]. Regarding classical logics, the calculus of structures has achieved important results revealing completely new techniques for cut-elimination and interpolation [4, 6, 3, 2].

Operations on logics, such as product, fusion, temporalization, probalization and fibring [16, 38, 15, 32], is about composing and decomposing logics from the application of operators to other logics. The preservation of logical properties, such as completeness, interpolation and decidability, in the composition and decomposition of logics is the main advantage of operating over logics. Quantum, non-truth-functional and modal logics have been provided by this kind of enrichment mechanisms in the scope of Hilbert, Gentzen and labeled deductive systems [8, 25, 24]. Moreover, preservation of interpolation by fibring has been studied in [12].

Some further investigation can be done in the development of new insights, techniques and results regarding cut-elimination and interpolation, in particular

- the formulation of a purely syntactical proof of the cut-elimination and interpolation property in the propositional and predicate cases for classical and non-classical logic in the scope of deep inference;
- the formulation of sufficient conditions for a logic presented in deep inference to enjoy interpolation and cut-elimination;
- study of combination mechanisms of logics like fibring and probabilization in the scope of deep inference and obtaining sufficient conditions for the preservation of cut-elimination and the interpolation property;
- the formulation of new results regarding the proof complexity of the interpolants.

Bibliography

- [1] E. Beth. On Padoa's method in the theory of definition. *Proceedings of the Nederlandse Akadademie van Wetenschappen, Series A*, 56:330–339, 1953.
- [2] K. 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.
- [3] K. Brünnler. Two restrictions on contraction. *Logic Journal of the IGPL*, 11(5):525–529, 2003.
- [4] K. Brünnler. Cut elimination inside a deep inference system for classical predicate logic. *Studia Logica*, 82(1):51–71, 2006.
- [5] K. 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.
- [6] K. Brünnler and A. Guglielmi. A first order system with finite choice of premises. In Vincent Hendricks, Fabian Neuhaus, Stig Andur Pedersen, Uwe Scheffler, and Heinrich Wansing, editors, *First-Order Logic Revisited*, pages 59–74. Logos Verlag, 2004.
- [7] S. Buss. The undecidability of k-provability. *Annals of Pure and Applied Logic*, 53:72–102, 1991.
- [8] C. Caleiro, W. A. Carnielli, M. E. Coniglio, A. Sernadas, and C. Sernadas. Fibring non-truth-functional logics: Completeness preservation. *Journal of Logic, Language and Information*, 12(2):183–211, 2003.
- [9] A. Carbone. The Craig interpolation theorem for schematic systems. In *Colloquium Logicum*, volume 2 of *Annals of the Kurt Gödel Society*, pages 87–100. Springer-Verlag, 1996.
- [10] A. Carbone. Some combinatorics behind proofs. *Manuscript*, 1996.

- [11] A. Carbone. Interpolants, cut elimination and flow graphs for the propositional calculus. *Annals of Pure and Applied Logic*, 83:249–299, 1997.
- [12] W. A. Carnielli, J. Rasga, and C. Sernadas. Preservation of interpolation features by fibring. *Journal of Logic and Computation*. In print.
- [13] S. A. Cook and R. Reckhow. The relative efficiency of propositional proof systems. *The Journal of Symbolic Logic*, 44:36–50, 1979.
- [14] W. Craig. Linear reasoning. A new form of the Herbrand-Gentzen theorem. *The Journal of Symbolic Logic*, 3:250–268, 1957.
- [15] M. Finger and D. Gabbay. Adding a temporal dimension to a logic system. *Journal of Logic, Language and Information*, 1(3):203–233, 1992.
- [16] D. Gabbay and V. Shehtman. Products of modal logics i. *Logic Journal of the IGPL*, 6(1):73–146, 1998.
- [17] G. Gentzen. Untersuchungen über das logische schliessen. *Mathematische Zeitschrift*, 39:176–210 and 405–431, 1935. English translation: Investigations into logical deduction, in *The Collected Papers of Gerhard Gentzen*, edited by M. E. Szabo, pp 68-131, North-Holland, 1969.
- [18] P. 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.
- [19] J.-Y. Girard. Linear logic. *Journal of Theoretical Computer Science*, 50:1–102, 1987.
- [20] J.-Y. Girard. *Proof Theory and Logical Complexity*. Studies in Proof Theory. Bibliopolis, Napoli, 1987.
- [21] A. Guglielmi. A system of interaction and structure. *ACM Transaction on Computational Logic*, 8(1):1–64, 2007.
- [22] R. C. Lyndon. An interpolation theorem in the predicate calculus. *Pacific Journal of Mathematics*, 9:129–142, 1959.
- [23] S. Maehara. On the interpolation theorem of Craig. *Sugaku*, 12:235–237, 1960.
- [24] P. Mateus, J. Rasga, and C. Sernadas. Modal sequent calculi labelled with truth values: Cut elimination. *Logic Journal of the IGPL*, 13(2):173–199, 2005.

- [25] P. Mateus and A. Sernadas. Exogenous quantum logic. In W. A. Carnielli, F. M. Dionísio, and P. Mateus, editors, *CombLog'04-Workshop on Combination of Logics: Theory and Applications*, pages 141–150, Lisbon, 2004. IST Press.
- [26] K. McMillan. Applications of Craig interpolants in model checking. In Nicolas Halbwachs and Lenore D. Zuck, editors, *TACAS*, volume 3440 of *Lecture Notes in Computer Science*, pages 1–12. Springer, 2005.
- [27] D. Mundici. A quadratic lower bound for the complexity of boolean interpolants. *Technical report, Department of Computer Science, University of Milan*, 1993.
- [28] T. Nagashima. An extension of the Craig-Schütte interpolation theorem. *Annals of the Japan Association for Philosophy of Science*, 2:12–18, 1966.
- [29] R. Parikh. Some results on the length of proofs. *Transaction of the American Mathematical Society*, 177:29–36, 1973.
- [30] P. Pudlák. The length of proofs. In S. Buss, editor, *Handbook of Proof Theory*. North-Holland, 1995.
- [31] A. Robinson. A result on consistency and its application to the theory of definition. *Proceedings of the Nederlandse Akademie van Wetenschappen, Series A*, 59:47–58, 1956.
- [32] A. Sernadas, C. Sernadas, and C. Caleiro. Fibring logics as a categorical construction. *Journal of Logic and Computation*, 9(2):149–179, 1999.
- [33] C. Stewart and P. Stouppa. A systematic proof theory for several modal logics. In Renate Schmidt, Ian Pratt-Hartmann, Mark Reynolds, and Heinrich Wansing, editors, *Advances in Modal Logic*, volume 5, pages 309–333. King's College Publications, 2005.
- [34] L. Straßburger. *Linear Logic and Noncommutativity in the Calculus of Structures*. PhD thesis, Technische Universität Dresden, 2003.
- [35] L. Straßburger. MELL in the calculus of structures. *Theoretical Computer Science*, 309:213–285, 2003.
- [36] L. Straßburger. System NEL is undecidable. In Ruy De Queiroz, Elaine Pimentel, and Lucília Figueiredo, editors, *10th Workshop on Logic, Language, Information and Computation (WoLLIC)*, volume 84 of *Electronic Notes in Theoretical Computer Science*. Elsevier, 2003.

- [37] G. Takeuti. *Proof Theory*. North-Holland, Amsterdam, second edition, 1987.
- [38] R. Thomason. Combinations of tense and modality. In *Handbook of philosophical logic, Vol. II*, volume 165 of *Synthese Library*, pages 135–165. Reidel, Dordrecht, 1984.
- [39] A. Tiu. A system of interaction and structure II: The need for deep inference. *Logical Methods in Computer Science*, 2(2):4:1–24, 2006.
- [40] A. S. Troelstra and H. Schwichtenberg. *Basic Proof Theory*. Cambridge Tracts in Theoretical Computer Science (No. 43). Cambridge University Press, second edition, June 2000.