Deep Inference Proof Systems for Monotone Quantifiers

Loïc Allègre
LIRMM - University of Montpellier/CNRS, France

Abstract

Constructing reasonable proof systems in the presence of generalized quantifiers is challenging, in part due to the general lack of sound introduction/elimination rules for such quantifiers. We propose an alternative approach to the proof theory of generalized quantifiers, using deep inference methods to circumvent this problem. We show that the quantifiers suitable for a deep inference treatment are specifically the class of monotone quantifiers, and subsequently present a generic methodology for designing deep inference proof systems for logics with monotone quantifiers.

2012 ACM Subject Classification Theory of computation → Proof theory

Keywords and phrases Logic, Proof Theory, Generalized Quantifiers, Deep Inference

Category Work In Progress

1 Introduction

Generalized quantifiers, as originally introduced by Mostowski [12] and Lindström [11], are a well-known logical formalism used to express quantification beyond the usual quantifiers $\forall$ and $\exists$ of modern logic. While these quantifiers enjoy a relatively straightforward notion of models, their proof theory is considerably more challenging, and consequently much less studied. Typically, except for specific cases of particular quantifiers, there are no notions of proof systems for arbitrary generalized quantifiers *in general*.

This can be explained in part by the difficulty of finding sound inference rules for generalized quantifiers. Most traditional proof systems for logics with the usual quantifiers $\forall$ and $\exists$ make use of some form of *introduction/elimination* rules, allowing inferences between term-level statements and generic statements with quantifiers, such as:

$$\exists \frac{A(t)}{\exists x A(x)} \qquad \forall \frac{\forall x A(x)}{A(t)}$$

When dealing with generalized quantifiers, however, these kinds of rules fail in the overwhelming majority of cases. Consider for example the quantifier "at least κ": $Q_{\geq\kappa} = \{A \subseteq \mathcal{M} \mid |A| \geq \kappa\}$ for some cardinal κ. From $A(t)$, one cannot infer $Q_{\geq\kappa}xA(x)$, and from $Q_{\geq\kappa}xA(x)$ we cannot say anything about a particular term t. Thus, in the framework of a shallow proof system, when a formula starts with a quantifier that does not have proper introduction/elimination rules, no inference can be applied.

However, it is still possible to make inferences *directly within* the quantifiers. Consider the following valid scheme: $\forall x(A(x) \to B(x)) \to (Q_{\geq\kappa}xA(x) \to Q_{\geq\kappa}xB(x))$. It expresses that if B follows from A, then we can infer $Q_{\geq\kappa}xB(x)$ from $Q_{\geq\kappa}xA(x)$.

We argue that these kinds of properties advocate for a treatment of generalized quantifiers using *deep inference* methods. Deep inference, initially introduced by Guglielmi [5], is a formalism in which inference rules can be applied arbitrarily deep within formulae, instead of only at the top-level connector. In this ongoing work, we show that the quantifiers that are suitable for a deep inference approach are precisely *monotone* quantifiers, and propose a methodology for designing deep proof systems for logics with monotone quantifiers, circumventing the lack of sound shallow introduction/elimination rules.

2 Monotone Generalized Quantifiers

Generalized quantifiers were introduced by Mostowski [12], and later Lindström [11], as an extension of the concept of quantifiers to generic relations on relations in a model. We start by giving a brief reminder on the essential definitions and properties of generalized quantifiers that will be needed in the rest of this paper.

2.1 Basic Definitions

▶ **Definition 1** (Generalized Quantifier). *A* generalized quantifier of type $\langle k_1, ..., k_n \rangle$ *is a mapping* $Q(S)$ *from each non-empty set* S *to a subset of* $\mathcal{P}(S)^{k_1} \times ... \times \mathcal{P}(S)^{k_n}$. .

In other words, a quantifier on a set S is simply a n-ary relation over k_i-ary relations on S.

Examples:

1. The type $\langle 1 \rangle$ existential quantifier $\exists$ is the mapping:
$$\exists(M) = \{A \subseteq M \mid A \neq \emptyset\}$$
2. The type $\langle 1 \rangle$ quantifier "infinitely many" is:
$$Q^{\geq \omega}(M) = \{A \subseteq M \mid A \text{ is infinite}\}$$
3. The type $\langle 1, 1 \rangle$ quantifier "more ... than ..." is the binary relation:
$$More(M) = \{(A, B) \in \mathcal{P}(M) \times \mathcal{P}(M) \mid |A| > |B|\}$$

▶ **Definition 2** (Negations). *For any quantifier* Q*, we define*

- *the* outer negation *of* Q *as:*
$$(\neg Q)(M) = \{(A_1, ..., A_n) \in \mathcal{P}(M)^{k_1} \times ... \times \mathcal{P}(M)^{k_n} \mid (A_1, ..., A_n) \notin Q(M)\}$$
- *the* inner negation *of* Q *on the* i*-th argument as:*
$$(Q\neg_i) = \{(A_1, ..., A_i, ..., A_n) \in \mathcal{P}(M)^{k_1} \times ... \times \mathcal{P}(M)^{k_n} \mid (A_1, ..., \overline{A_i}, ..., A_n) \in Q(M)\}$$
where $\overline{A_i}$ *is the complement of* A_i *in* $\mathcal{P}(M)^{k_i}$.

The *dual* of Q on the i-th argument is defined as the quantifier $Q^{d_i} = (\neg Q)\neg_i = \neg(Q\neg_i)$. A n-ary quantifier therefore has n possibly different duals, and in case $n = 1$ we recover the usual notion of quantifier negation and duality.

2.2 First-order Logic with Generalized Quantifiers

The notion of generalized quantifier as defined above is a purely set-theoretic one. However, we can associate quite naturally to any quantifier a new corresponding logical operator extending a chosen logical language.

In the rest of this work, we place ourselves within the framework of a usual first-order language $\mathcal{L}$. The notion of model for $\mathcal{L}$ is that of an usual $\mathcal{L}$-structure $\mathcal{M} = (M, I)$ with M its universe and I an intepretation function of terms and formulae in $\mathcal{M}$. For a given model $\mathcal{M}$ and an assignment of free variables s, we note $\mathcal{M}, s \models \varphi$ to mean that φ is true in $\mathcal{M}$ with respect to s, and simply $\mathcal{M} \models \varphi$ when φ is true in $\mathcal{M}$ for every assignment. For an assignment s, we write $s[\overrightarrow{a}/\overrightarrow{x}]$ for the assignment that is equal to s, except that it assigns $\overrightarrow{a}$ to the variables $\overrightarrow{x}$.

▶ **Definition 3.** *For a type $\langle k_1, ..., k_n \rangle$ quantifier Q, we add to $\mathcal{L}$ a new quantifier symbol $\mathcal{Q}$, and a new formula formation rule: if $\overrightarrow{x_1}, ..., \overrightarrow{x_n}$ are respectively $k_1, ..., k_n$-tuples of variables and $\varphi_1, ..., \varphi_n$ are formulae, then:*

$$\mathcal{Q}\overrightarrow{x_1}, ..., \overrightarrow{x_n}[\varphi_1, ..., \varphi_n]$$

is a formula, in which for every i the variables $\overrightarrow{x_i}$ are bound in φ_i by $\mathcal{Q}$.

The semantics of this new logical symbol is then given by the corresponding quantifier Q, in that for a model $\mathcal{M}$ with domain M:

$$\mathcal{M}, s \models \mathcal{Q}\overrightarrow{x_1}, ..., \overrightarrow{x_n}[\varphi_1, ..., \varphi_n]$$

iff

$$(\{(\overrightarrow{a_1}) \in M^{k_1} \mid \mathcal{M}, s[\overrightarrow{a_1}/\overrightarrow{x_1}] \models \varphi_1\}, ..., \{(\overrightarrow{a_n}) \in M^{k_n} \mid \mathcal{M}, s[\overrightarrow{a_n}/\overrightarrow{x_n}] \models \varphi_n\}) \in Q(M)$$

We note $\mathcal{L}(\mathcal{Q})$ the language of first-order logic augmented with the symbol $\mathcal{Q}$ for the quantifier Q. $\mathcal{Q}$ is sometimes called a *global* quantifier, and $Q(M)$ a *local* quantifier. For a given, fixed Q, and for the sake of readability, we will allow ourselves to abuse notation and write $Q\overrightarrow{x}[...]$ for $\mathcal{Q}\overrightarrow{x}[...]$.

By this definition, it is also immediate to see that the definitions and properties of negation, inner negation and duals for local quantifiers naturally lift to the corresponding global quantifiers.

A remark must be made concerning the choice of automatically including dual quantifiers in the extension of the language. In general, there is no obligation that the language $\mathcal{L}(\mathcal{Q})$ contains the dual quantifiers of $\mathcal{Q}$, but more often than not it is practical to include them as well. In particular, this would allow one to recover the familiar duality equivalence: $\neg\mathcal{Q}\overrightarrow{x}[..., \varphi_i, ...] \equiv \mathcal{Q}^{d_i}\overrightarrow{x}[..., \neg\varphi_i, ...]$.

2.3 Monotonicity and Polarity

The concept of monotonicity of a generalized quantifier is due to Barwise and Cooper [3]. Informally, monotone quantifiers are those quantifiers that preserve inclusion relations (in one direction or the other) between subsets of the domain, meaning that from a inferential view, they are the quantifiers that preserve logical entailment, which is precisely what makes them suitable for a deep inference treatment.

▶ **Definition 4** (Monotone Quantifiers)**.** *We say that a quantifier Q on a model $\mathcal{M}$ is:*

1. Increasing *in the i-th argument if for all $(A_1, ..., A_n) \in \mathcal{P}(M)^{k_1} \times ... \times \mathcal{P}(M)^{k_n}$ and for all $B_i \supseteq A_i$, we have that $(A_1, ..., A_i, ..., A_n) \in Q(M)$ implies $(A_1, ..., B_i, ..., A_n) \in Q(M)$.*
2. Decreasing *in the i-th argument if for all $(A_1, ..., A_n) \in \mathcal{P}(M)^{k_1} \times ... \times \mathcal{P}(M)^{k_n}$ and for all $B_i \subseteq A_i$, we have that $(A_1, ..., A_i, ..., A_n) \in Q(M)$ implies $(A_1, ..., B_i, ..., A_n) \in Q(M)$.*

If Q is increasing (resp. decreasing) on every $\mathcal{M}$, we say that the global quantifier $\mathcal{Q}$ is increasing (resp. decreasing).

The definition of monotonicity and negation gives us the following immediate property:

▶ **Proposition 5.** *Q is i-increasing iff $\neg Q, Q\neg_i$ are i-decreasing, and Q is i-decreasing iff $\neg Q, Q\neg_i$ are i-increasing.*

In particular, this means that Q and Q^{d_i} have the same i-monotonicity, and that if Q is j-monotone, then Q^{d_i} with $i \neq j$ has opposed j-monotonicity.

In addition to monotonicity, we will need to introduce the notion of *polarity* of a quantifier, which essentially describes whether it accepts or rejects the empty set and/or the full domain M.

▶ **Definition 6** (Quantifier Polarity). *We say that a quantifier Q on a model $\mathcal{M}$ is, in the i-th argument:*

1. Positive *if for all A_j with $j \neq i$, $(..., A_{i-1}, M^{k_i}, A_{i+1}, ...) \in Q(M)$.*
2. Negative *if for all A_j with $j \neq i$, $(..., A_{i-1}, \emptyset, A_{i+1}, ...) \in Q(M)$.*
3. Anti-positive *if for all A_j with $j \neq i$, $(..., A_{i-1}, M^{k_i}, A_{i+1}, ...) \notin Q(M)$.*
4. Anti-negative *if for all A_j with $j \neq i$, $(..., A_{i-1}, \emptyset, A_{i+1}, ...) \notin Q(M)$.*

As before, if Q is one of the above on all models $\mathcal{M}$, we say that the global $\mathcal{Q}$ also has the corresponding polarity.

In the case Q is a unary type $\langle k \rangle$ quantifier, the characterization becomes much simpler since there is only one argument to consider, and thus Q is positive iff $M^k \in Q(M)$, negative iff $\emptyset \in Q(M)$, etc.

In terms of the global quantifier $\mathcal{Q}$, it is equivalent to the formulation: $\mathcal{Q}$ is positive iff for all $\mathcal{M}$, $\mathcal{M} \models Q\overrightarrow{x}[\top]$, negative iff $\mathcal{M} \models Q\overrightarrow{x}[\bot]$, and so on.

We also have a convenient result that links monotonicity and polarity for unary quantifiers.

▶ **Proposition 7.** *If Q is an increasing type $\langle k \rangle$ quantifier on $\mathcal{M}$ and $Q(M)$ is not trivial, i.e. $Q(M) \neq \mathcal{P}(M)^k$ and $Q(M) \neq \emptyset$, then Q is positive and anti-negative.*

If instead Q is decreasing and $Q(M)$ is not trivial, then Q is negative and anti-positive.

Proof. Immediate from the definition of monotonicity and non-triviality. ◀

Again, when this holds for all $\mathcal{M}$, the property naturally extends to the global quantifier $\mathcal{Q}$. However, it is important that non-triviality holds for all $\mathcal{M}$: consider the quantifier $Q^{\geq\omega}(M) = \{A \subseteq M \mid A \text{ is infinite}\}$, which is increasing in all $\mathcal{M}$ and thus globally increasing, but is trivial in all finite models, and therefore is not globally positive.[1]

In the case of n-ary quantifiers, as will be discussed later, an analoguous property holds, but it requires much stronger hypotheses on Q (in particular we need a stronger condition than simply non-triviality).

3 Proof Theory for Type $\langle k \rangle$ Monotone Quantifiers

The aim of this work is to propose a deep inference approach to the proof theory of generalized quantifiers. Deep inference methods are characterized by the application of inference rules within formulae instead of only at their root, so we need to study under which conditions rules can be applied inside the scope of a quantifier. That is, if $A \models B$, what properties are needed of a quantifier Q to ensure that $Q\overrightarrow{x}[A] \models Q\overrightarrow{x}[B]$? The answer is precisely *monotonicity*, since monotone quantifiers are those quantifiers that either preserve or reverse logical entailment relations.

Thus, we propose a general methodology for designing deep inference proof systems with monotone quantifiers. We only consider for now the case of unary type $\langle k \rangle$ quantifiers, as it

[1] It is positive with respect to the class of infinite models, however.

is easier to present: the generalization to arbitrary n-ary quantifiers follows quite naturally and without additional difficulty, and will be discussed later.

3.1 The Quantifier Calculus QC

We start by defining the *Quantifier Calculus QC*, which will serve as a general basis for designing more specific calculi. We choose to use a formalism based on *Open Deduction* [6], which we extend with rules for monotone generalized quantifiers.

We give ourselves the usual first-order language extended with any number of monotone quantifiers $\mathcal{L}(\mathcal{Q}_1, \mathcal{Q}_2, ...)$. For practical reasons, we will suppose that $\mathcal{L}(\mathcal{Q}_1, \mathcal{Q}_2, ...)$ is closed by quantifier duality, so each quantifier has a dual.

The *negation* $\overline{F}$ of a formula F is defined by De Morgan laws:

1. $\overline{\top} = \bot$
2. $\overline{\bot} = \top$
3. $\overline{p(t_1, ..., t_n)} = \overline{p}(t_1, ..., t_n)$
4. $\overline{A \wedge B} = \overline{A} \vee \overline{B}$
5. $\overline{A \vee B} = \overline{A} \wedge \overline{B}$
6. $\overline{\mathcal{Q}\vec{x}[A]} = \mathcal{Q}^d\vec{x}[\overline{A}]$

meaning we consider all formulae to be in *negation normal form*.

▶ **Definition 8** (Syntactic Equality). *We define* syntactic equality $=$ *as the smallest equivalence relation on formulae such that for all formulae* A, B, C*, and quantifier* Q*:*

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

$$A \wedge B = B \wedge A \qquad A \vee B = B \vee A$$

$$A \wedge (B \wedge C) = (A \wedge B) \wedge C \qquad A \vee (B \vee C) = (A \vee B) \vee C$$

$$\mathcal{Q}\vec{x}[\top] = \top \text{ if } Q \text{ is positive} \qquad \mathcal{Q}\vec{x}[\bot] = \top \text{ if } Q \text{ is negative}$$

$$\mathcal{Q}\vec{x}[\top] = \bot \text{ if } Q \text{ is anti-positive} \qquad \mathcal{Q}\vec{x}[\bot] = \bot \text{ if } Q \text{ is anti-negative}$$

▶ **Definition 9** (QC Inference Rules). *We define the set of* inference rules *of QC as follows:*

$$\mathsf{i}{\downarrow}\frac{\top}{A \vee \overline{A}} \qquad \mathsf{i}{\uparrow}\frac{A \wedge \overline{A}}{\bot} \qquad \mathsf{s}\frac{(A \vee B) \wedge C}{A \vee (B \wedge C)}$$

$$\mathsf{w}{\downarrow}\frac{\bot}{A} \qquad \mathsf{w}{\uparrow}\frac{A}{\top} \qquad \mathsf{qs}\frac{\mathcal{Q}\vec{x}[A(\vec{x}) \vee B] \wedge C}{\mathcal{Q}\vec{x}[A(\vec{x}) \wedge C] \vee B} \qquad =\frac{A}{B} \text{ if } A = B$$

$$\mathsf{c}{\downarrow}\frac{A \vee A}{A} \qquad \mathsf{c}{\uparrow}\frac{A}{A \wedge A} \qquad \mathsf{qmix}\frac{\mathcal{Q}\vec{x}[A(\vec{x}) \wedge B] \wedge \overline{C}}{\mathcal{Q}\vec{x}[A(\vec{x}) \vee C] \vee \overline{B}}$$

▶ **Definition 10** (Derivations). *We define* derivations *with premise* A *and conclusion* B*, written* $\begin{array}{c} A \\ \| \\ B \end{array}$*, as follows:*

1. *If F is a formula, then F is a derivation with premise and conclusion F.*
2. *If $\begin{array}{c} A \\ \| \\ B \end{array}$ and $\begin{array}{c} C \\ \| \\ D \end{array}$ are derivations, then $\boxed{\begin{array}{c} A \\ \| \\ B \end{array}} \star \boxed{\begin{array}{c} C \\ \| \\ D \end{array}}$ is a derivation with premise $A \star C$ and conclusion $B \star D$, for $\star \in \{\vee, \wedge\}$.*
3. *If $\begin{array}{c} A \\ \| \\ B \end{array}$ and $\begin{array}{c} C \\ \| \\ D \end{array}$ are derivations, and if $\rho \dfrac{B}{C}$ is an instance of an inference rule, then $\rho \dfrac{\boxed{\begin{array}{c} A \\ \| \\ B \end{array}}}{\boxed{\begin{array}{c} C \\ \| \\ D \end{array}}}$ is a derivation with premise A and conclusion D.*
4. *If $\mathcal{Q}^{\uparrow}$ is an increasing quantifier and $\begin{array}{c} A \\ \| \\ B \end{array}$ is a derivation, then $\mathcal{Q}^{\uparrow}\overrightarrow{x} \begin{bmatrix} A \\ \| \\ B \end{bmatrix}$ is a derivation with premise $\mathcal{Q}^{\uparrow}\overrightarrow{x}[A]$ and conclusion $\mathcal{Q}^{\uparrow}\overrightarrow{x}[B]$.*
5. *If $\mathcal{Q}^{\downarrow}$ is a decreasing quantifier and $\begin{array}{c} A \\ \| \\ B \end{array}$ is a derivation, then $\mathcal{Q}^{\downarrow}\overrightarrow{x} \begin{bmatrix} A \\ \| \\ B \end{bmatrix}$ is a derivation with premise $\mathcal{Q}^{\downarrow}\overrightarrow{x}[B]$ and conclusion $\mathcal{Q}^{\downarrow}\overrightarrow{x}[A]$.*

One must be careful with the case of composition by a decreasing quantifier: since they reverse logical entailment, the premise and the conclusion are swapped by the quantifier, which is unusual compared to more traditional proof systems.

When there is a derivation $\begin{array}{c} \top \\ \| \\ A \end{array}$ with conclusion A, we say it is a *proof* of A, noted $\begin{array}{c} \mathbb{T} \\ \| \\ A \end{array}$.

3.2 Soundness

We show that QC is sound, i.e. that derivations with a valid premise have a valid conclusion, in particular that there is no proof of $\bot$. The propositional fragment is more or less the same as usual deep inference systems, but the introduction of generalized quantifiers requires some careful verification.

▶ **Lemma 11.** *If $A \models B$, then for every n and n-tuple of variables $\overrightarrow{x}$ that do not appear bound in A or B, and for every model $\mathcal{M}$, we have:*

$$\{\overrightarrow{a} \in M^n \mid \mathcal{M}, s[\overrightarrow{a}/\overrightarrow{x}] \models A\} \subseteq \{\overrightarrow{a} \in M^n \mid \mathcal{M}, s[\overrightarrow{a}/\overrightarrow{x}] \models B\}$$

This lemma is the crucial property guaranteeing that we will be able to leverage the monotonicity of quantifiers to apply inference rules inside their scope without losing validity.

Proof. Suppose that $A \models B$, and let $\mathcal{M}$ be a model.

For any $\overrightarrow{a} \in M^n$, if $\mathcal{M}, s[\overrightarrow{a}/\overrightarrow{x}]$ is a model of A, then since $A \models B$, $\mathcal{M}, s[\overrightarrow{a}/\overrightarrow{x}]$ is also a model of B.

Thus $\{\overrightarrow{a} \in M^n \mid \mathcal{M}, s[\overrightarrow{a}/\overrightarrow{x}] \models A\} \subseteq \{\overrightarrow{a} \in M^n \mid \mathcal{M}, s[\overrightarrow{a}/\overrightarrow{x}] \models B\}$.

◀

▶ **Theorem 12** (Soundness). *QC is sound, i.e. if* $\begin{array}{c}A\\ \|\\ B\end{array}$*, then* $A \models B$.

Proof. We proceed by induction on the size of a derivation.

The base case are formulae F, which are derivations from F to F. They are trivally sound derivations, since $F \models F$ always.

For the inductive cases, we need to show that soundness is preserved by composition of derivations. Let δ be a derivation of size n, and suppose that all derivations of size less than n are sound.

We have the following cases:

1. Composition by $\vee, \wedge$: suppose $\delta = \begin{array}{c}\boxed{\begin{array}{c}A\\ \|\\ B\end{array}}\\ A\end{array} \vee \begin{array}{c}\boxed{\begin{array}{c}C\\ \|\\ D\end{array}}\\ C\end{array}$, and let $\mathcal{M}$ be a model such that $M \models A \vee C$.
   Then by induction hypothesis $\begin{array}{c}A\\ \|\\ B\end{array}$ and $\begin{array}{c}C\\ \|\\ D\end{array}$ are sound derivations. If $\mathcal{M} \models A$, then $\mathcal{M} \models B$, and if $\mathcal{M} \not\models A$ then necessarily $\mathcal{M} \models C$, which entails that $\mathcal{M} \models D$. In either case, we have $\mathcal{M} \models B \vee D$, and thus δ is sound.
   The case for $\wedge$ is similar.

2. Composition by inference: we first need to prove that all inference rules of QC are sound. The soundness of the propositional rules $\{\mathsf{i}\uparrow, \mathsf{i}\downarrow, \mathsf{w}\uparrow, \mathsf{w}\downarrow, \mathsf{c}\uparrow, \mathsf{c}\downarrow, \mathsf{s}, =\}$ is immediate (they are the usual propositional open deduction rules). The soundness of $=$ for the quantifier equalities is also immediate, by definition of quantifier polarities.
   The only non-trivial cases are the quantifier switch qs and the quantifier mix qmix:

   a. qs: Suppose that $\mathcal{M} \models \mathcal{Q}\overrightarrow{x}[A(\overrightarrow{x}) \vee B] \wedge C$. If $\mathcal{M} \models B$, immediately $\mathcal{M} \models \mathcal{Q}\overrightarrow{x}[A(\overrightarrow{x}) \wedge C] \vee B$.
   If $\mathcal{M} \not\models B$, then $\{\overrightarrow{x} \in M^n \mid \mathcal{M} \models A(\overrightarrow{x}) \vee B\} = \{\overrightarrow{x} \in M^n \mid \mathcal{M} \models A(\overrightarrow{x})\}$. And since $\mathcal{M} \models C$, we have also $\{\overrightarrow{x} \in M^n \mid \mathcal{M} \models A(\overrightarrow{x})\} = \{\overrightarrow{x} \in M^n \mid \mathcal{M} \models A(\overrightarrow{x}) \wedge C\}$. Thus $\mathcal{M} \models \mathcal{Q}\overrightarrow{x}[A(\overrightarrow{x}) \wedge C] \vee B$.

   b. qmix: Suppose that $\mathcal{M} \models \mathcal{Q}\overrightarrow{x}[A(\overrightarrow{x}) \wedge B] \wedge \overline{C}$. If $M \models \overline{B}$, then $\mathcal{M} \models \mathcal{Q}\overrightarrow{x}[A(\overrightarrow{x}) \vee C] \vee \overline{B}$.
   If instead $\mathcal{M} \models B$, then $\{\overrightarrow{x} \in M^n \mid \mathcal{M} \models A(\overrightarrow{x}) \wedge B\} = \{\overrightarrow{x} \in M^n \mid \mathcal{M} \models A(\overrightarrow{x})\}$. And since $\mathcal{M} \models \overline{C}$, we have also $\{\overrightarrow{x} \in M^n \mid \mathcal{M} \models A(\overrightarrow{x})\} = \{\overrightarrow{x} \in M^n \mid \mathcal{M} \models A(\overrightarrow{x}) \vee C\}$. Thus $\mathcal{M} \models \mathcal{Q}\overrightarrow{x}[A(\overrightarrow{x}) \vee C] \vee \overline{B}$.

   Then the soundness of composition by inference follows immediately: if $\delta = \rho \dfrac{\boxed{\begin{array}{c}A\\ \|\\ B\end{array}}}{\boxed{\begin{array}{c}C\\ \|\\ D\end{array}}}$, then by induction hypothesis $\begin{array}{c}A\\ \|\\ B\end{array}$ and $\begin{array}{c}C\\ \|\\ D\end{array}$ are sound. Thus, if $\mathcal{M} \models A$, then $\mathcal{M} \models B$, which means by soundness of ρ that $\mathcal{M} \models C$, and consequently $\mathcal{M} \models D$.

3. Composition by an increasing quantifier $\mathcal{Q}^{\uparrow}$:
If $\delta = \mathcal{Q}^{\uparrow}\overrightarrow{x}\begin{bmatrix} A \\ \| \\ B \end{bmatrix}$, then by hypothesis $\begin{matrix} A \\ \| \\ B \end{matrix}$ is sound and $A \models B$. Therefore, by Lemma 11, we have for any model $\mathcal{M}$:

$$\{\overrightarrow{a} \in M^n \mid \mathcal{M}, s[\overrightarrow{a}/\overrightarrow{x}] \models A\} \subseteq \{\overrightarrow{a} \in M^n \mid \mathcal{M}, s[\overrightarrow{a}/\overrightarrow{x}] \models B\}$$

since the $\overrightarrow{x}$ do not appear bound in A, B.[2]
Since $\mathcal{Q}^{\uparrow}$ is increasing, if follows from the definition that if $\mathcal{M} \models \mathcal{Q}^{\uparrow}\overrightarrow{x}[A]$ then $\mathcal{M} \models \mathcal{Q}^{\uparrow}\overrightarrow{x}[B]$.

4. Composition by a decreasing quantifier $\mathcal{Q}^{\downarrow}$:
If $\delta = \mathcal{Q}^{\downarrow}\overrightarrow{x}\begin{bmatrix} A \\ \| \\ B \end{bmatrix}$, then by hypothesis $\begin{matrix} A \\ \| \\ B \end{matrix}$ is sound and $A \models B$. Therefore, by Lemma 11, we have for any model $\mathcal{M}$:

$$\{\overrightarrow{a} \in M^n \mid \mathcal{M}, s[\overrightarrow{a}/\overrightarrow{x}] \models A\} \subseteq \{\overrightarrow{a} \in M^n \mid \mathcal{M}, s[\overrightarrow{a}/\overrightarrow{x}] \models B\}$$

since the $\overrightarrow{x}$ do not appear bound in A, B
Since $\mathcal{Q}^{\downarrow}$ is decreasing, we have that if $\mathcal{M} \models \mathcal{Q}^{\downarrow}\overrightarrow{x}[B]$ then $\mathcal{M} \models \mathcal{Q}^{\downarrow}\overrightarrow{x}[A]$, and thus δ is a sound derivation $\begin{matrix} \mathcal{Q}^{\downarrow}\overrightarrow{x}[B] \\ \| \\ \mathcal{Q}^{\downarrow}\overrightarrow{x}[A] \end{matrix}$[3].

We conclude by induction that for all formulae, if there is a derivation $\begin{matrix} A \\ \| \\ B \end{matrix}$ then $A \models B$, in particular that there is no proof of $\bot$.

◀

3.3 Examples and properties

To illustrate the behaviour of the calculus, we give some examples of derivations, and some derived rules that are admissible in the calculus QC.

1. For any *increasing* quantifier $\mathcal{Q}^{\uparrow}$, the rule $\mathsf{qc}{\uparrow}\dfrac{\mathcal{Q}^{\uparrow}\overrightarrow{x}[A(\overrightarrow{x}) \wedge B(\overrightarrow{x})]}{\mathcal{Q}^{\uparrow}\overrightarrow{x}[A(\overrightarrow{x})] \wedge \mathcal{Q}^{\uparrow}\overrightarrow{x}[B(\overrightarrow{x})]}$ is admissible:

$$\mathsf{c}{\uparrow}\dfrac{\mathcal{Q}\overrightarrow{x}[A(\overrightarrow{x}) \wedge B(\overrightarrow{x})]}{\mathcal{Q}^{\uparrow}\overrightarrow{x}\boxed{\mathsf{w}{\uparrow}\dfrac{A(\overrightarrow{x}) \wedge B(\overrightarrow{x})}{A(\overrightarrow{x})}} \wedge \mathcal{Q}^{\uparrow}\overrightarrow{x}\boxed{\mathsf{w}{\uparrow}\dfrac{A(\overrightarrow{x}) \wedge B(\overrightarrow{x})}{B(\overrightarrow{x})}}}$$

2. By taking the dual derivation, we have the admissiblity of the rule $\mathsf{qc}{\downarrow}\dfrac{\mathcal{Q}^{\uparrow}\overrightarrow{x}[A(\overrightarrow{x})] \vee \mathcal{Q}^{\uparrow}\overrightarrow{x}[B(\overrightarrow{x})]}{\mathcal{Q}\overrightarrow{x}[A(\overrightarrow{x}) \vee B(\overrightarrow{x})]}$.

---

[2] Otherwise the formulae would not be well-formed.
[3] Recall that decreasing quantifiers switch premise and conclusion.

3. For any *decreasing* quantifier $\mathcal{Q}^{\downarrow}$, the rule $\mathsf{dqc}{\downarrow}\dfrac{\mathcal{Q}^{\downarrow}\overrightarrow{x}[A(\overrightarrow{x}) \vee B(\overrightarrow{x})]}{\mathcal{Q}^{\downarrow}\overrightarrow{x}[A(\overrightarrow{x})] \wedge \mathcal{Q}^{\downarrow}\overrightarrow{x}[B(\overrightarrow{x})]}$ is admissible:

$$\mathsf{c}{\uparrow}\dfrac{\mathcal{Q}^{\downarrow}\overrightarrow{x}[A(\overrightarrow{x}) \vee B(\overrightarrow{x})]}{\mathcal{Q}^{\downarrow}\overrightarrow{x}\boxed{\mathsf{w}{\uparrow}\dfrac{A(\overrightarrow{x})}{A(\overrightarrow{x}) \vee B(\overrightarrow{x})}} \wedge \mathcal{Q}^{\downarrow}\overrightarrow{x}\boxed{\mathsf{w}{\uparrow}\dfrac{B(\overrightarrow{x})}{A(\overrightarrow{x}) \vee B(\overrightarrow{x})}}}$$

Remark that the conclusion is a conjunction, whereas the premise has a disjunction within the quantifier: this may seem odd, but is due the decreasing nature of the quantifier[4]. Recall also that since $\mathcal{Q}^{\downarrow}$ is decreasing, $\mathcal{Q}^{\downarrow}\overrightarrow{x}\,\mathsf{w}{\uparrow}\dfrac{A(\overrightarrow{x})}{A(\overrightarrow{x}) \vee B(\overrightarrow{x})}$ is a valid derivation with *premise* $\mathcal{Q}^{\downarrow}\overrightarrow{x}[A(\overrightarrow{x}) \vee B(\overrightarrow{x})]$ and *conclusion* $\mathcal{Q}^{\downarrow}\overrightarrow{x}[A(\overrightarrow{x})]$.

4. By duality, we also have the admissibility of the rule $\mathsf{dqc}{\uparrow}\dfrac{\mathcal{Q}^{\downarrow}\overrightarrow{x}[A(\overrightarrow{x})] \vee \mathcal{Q}^{\downarrow}\overrightarrow{x}[B(\overrightarrow{x})]}{\mathcal{Q}^{\downarrow}\overrightarrow{x}[A(\overrightarrow{x}) \wedge B(\overrightarrow{x})]}$

4 Extending QC with specific quantifier rules

We presented system QC with the aim to provide a framework as general as possible for dealing with monotone quantifiers, and contains only the rules that are valid for *all* monotone quantifiers. In that sense, QC is to be seen as more of a "system template" rather than one effective proof system. To begin with, it is not very reasonable to consider a system containing *all* monotone quantifiers that can be defined. Typical use cases consider a first-order language extended with only a few particular quantifiers (or a specific family of quantifiers), and QC provides sound rule blueprints to design proof systems.

Depending on the case, one might want to add specific inference rules to QC to account for properties of specific quantifiers, for example to reintroduce into the calculus the usual introduction and elimination rules of $\forall, \exists$. In this section, we discuss how additional rules can be integrated into QC.

In fact, since monotone quantifiers preserve entailment, the only requirement to ensure that adding a rule $\rho\dfrac{A}{B}$ does nor break quantifier rules is that it must be *locally sound*, meaning its premise must logically entails its conclusion, i.e. that $A \models B$. This constraint is required for Lemma 11 to hold true, and without this lemma, the soundness of rules applications within quantifiers fails, since they cannot be justified then by the monotonicity of quantifiers.

▶ **Theorem 13.** *Let ρ be a locally sound inference rule. Then $QC + \{\rho\}$ is sound.*

Proof. The proof is the same as the proof of Theorem 12, except there is an additional base case for ρ, which is covered by the fact that ρ is locally sound. The soundness of composition of derivations by ρ follows. ◀

One example of such a rule that is *not* locally sound and thus cannot be added to QC is typically the $\forall$ introduction : $\forall\dfrac{A(x)}{\forall y A(y)}$ (with x eigenvariable), precisely *because* of

[4] To convice yourself of the soundness of this statement, consider the fact that "There are no (A or B)" implies that "There are no A" *and* "There are no B".

the eigenvariable condition. This rule is *generally sound*: if the premise is valid, then the conclusion is valid, but we do not have $A(x) \models \forall y A(y)$.[5]

As an illustration, we will show how we can extend QC to recover usual rules for $\forall$ and $\exists$. Being both increasing and positive quantifiers, they of course already enjoy the general quantifier rules of QC, but it would be nice if we could also use their usual rules.

Existential instantiation can be added as is, since we have $A(t) \models \exists x A(x)$ for any term t:

$$\mathsf{n}{\downarrow}\,\frac{A(t)}{\exists x A(x)}$$

As mentioned above however, universal introduction cannot. Instead, in a similar fashion to [4] or [1], we can add *retract* rules :

$$\mathsf{r}{\downarrow}\,\frac{\forall x A(x) \vee B}{\forall x (A(x) \vee B)} \qquad \mathsf{r}{\uparrow}\,\frac{\forall x (A(x) \wedge B)}{\forall x A(x) \wedge B}$$

We give some examples of derivations in $QC + \{\mathsf{n}{\downarrow}, \mathsf{r}{\downarrow}, \mathsf{r}{\uparrow}\}$:

1. For any increasing, positive $\mathcal{Q}^{\uparrow}$, $\forall x A(x) \to \mathcal{Q}^{\uparrow} y A(y)$ is provable :

$$\dfrac{\dfrac{}{\top}\,{=}}{\dfrac{\mathcal{Q}^{\uparrow} y \left[\dfrac{\dfrac{}{\top}\,\mathsf{i}{\downarrow}}{A(y) \vee \left[\dfrac{\overline{A}(y)}{\exists x \overline{A}(x)}\,\mathsf{n}{\downarrow}\right]}\right]}{\exists x \overline{A}(x) \vee \mathcal{Q}^{\uparrow} y [A(y)]}\,\mathsf{qs}}$$

2. For any decreasing, negative $\mathcal{Q}^{\downarrow}$, $\exists x A(x) \vee \mathcal{Q}^{\downarrow} y A(y)$ is provable:

$$\dfrac{\dfrac{}{\top}\,{=}}{\dfrac{Q^{\downarrow} y \left[\dfrac{\overline{\exists x A(x)} \wedge \left[\dfrac{A(y)}{\exists x A(x)}\,\mathsf{n}{\downarrow}\right]}{\bot}\,\mathsf{i}{\uparrow}\right]}{\exists x A(x) \vee Q^{\downarrow} y [A(y)]}\,\mathsf{qmix}}$$

5 Proof Theory for Type $\langle k_1, ..., k_n \rangle$ Monotone Quantifiers

The properties and rules presented above for unary type $\langle k \rangle$ quantifiers extend very naturally to n-ary quantifiers. In general, by definition of monotonicity in the i-th argument of a quantifier, inference rules can be applied within the scope the i-th position in the quantifier, independently of the other arguments (provided that the inference rules only act on one argument of the quantifier, not multiple). Of course, if a n-ary quantifier $\mathcal{Q}$ is monotone only in some and not all of its arguments, rules cannot be applied within the non-monotone scopes of $\mathcal{Q}$.

What is however lost in the generalization to n-ary quantifiers is the direct link between monotonicity and polarity, in particular Proposition 7 does not hold as is. This means

[5] Consider a model $\mathcal{M}$ in which the free variable x is assigned to an a such that $\mathcal{M}, s \models A(a)$, but in which $\mathcal{M}, s \not\models \forall y A(y)$.

axiom/equality rules become less general and more dependent on particular quantifiers. But this is not necessarily a problem.

To illustrate this point, consider the $\langle 1,1\rangle$ quantifier "strictly more": $More[A,B](M) = \{(A,B) \subseteq \mathcal{M} \times \mathcal{M} \mid |A| > |B|\}$. It is increasing in the first argument and decreasing in the second, thus the corresponding rules are applicable. However, it is not positive in the first argument, since in any $\mathcal{M}$, $\mathcal{M} \not\models More_x[\top,\top]$, and it is not negative in the second since $\mathcal{M} \not\models More_x[\bot,\bot]$.

This means the equalities $More_x[\top, B(x)] = \top$ and $More_x[A(\overrightarrow{x}), \bot] = \top$ do not hold for generic A, B. However, we can still have axiom rules for this quantifier, since the equalities $More_x[\top,\bot] = \top$ and $More_x[\bot,\top] = \bot$ hold, but we do not have generic axioms for monotone n-ary quantifiers, and we need to add specific axioms for specific quantifiers on top of inference rules.

6 Non-monotone Quantifiers

The deep inference methodology we introduced here is only applicable to monotone quantifiers. In fact, monotonicity is precisely the property ensuring that inferences within a quantifier scope are sound. While the class of monotone quantifier encompasses a significant number of common quantifiers, there are still some interesting cases of non-monotone quantifiers, that fall outside the scope of this approach. An immediate example are all the quantifiers that accept sets of an *exact* cardinality, such as "exactly n" : $\exists_{=n}(M) = \{A \subseteq M \mid |A| = n\}$. This quantifier is neither increasing nor decreasing, thus inference schemata of the following form are unsound:

$$\exists_{=n} x \left[\frac{A(x)}{A(x) \vee \overline{A}(x)}\right] \qquad \exists_{=n} x \left[\frac{A(x)}{\top}\right]$$

In fact, the only inference rules that remain applicable are the ones where the premise and conclusion are semantically equivalent, i.e. $\{=, \mathsf{i}\uparrow, \mathsf{i}\downarrow, \mathsf{c}\uparrow, \mathsf{c}\downarrow\}$, since in any model, the premise and conclusion are interpreted by the same subset, and thus these rules are sound for *any* quantifier.

One possibility to circumvent this problem would be to express arbitrary quantifiers as conjunctions/intersections of monotone quantifiers. For example, "exactly n" is equivalent to "at least n and at most n": $\exists_{=n} x A(x) \equiv \exists_{\geq n} x A(x) \wedge \exists_{\leq n} x A(x)$. However, this approach fails in the general case, since the resulting conjunctions/intersections may not be finite. Consider for example the quantifier "an even number of", equivalent to "exactly zero *or* exactly two *or* ...", which would yield an infinite formula:

$$\exists_{2\mathbb{N}} \equiv \bigwedge_{n \in 2\mathbb{N}} (\exists_{\leq n} \wedge \exists_{\geq n})$$

7 Conclusion and Future Works

We showed in this ongoing work that deep inference methods can be used to propose proof systems for generalized quantifiers, circumventing the notorious challenges of designing sound shallow proof systems for these quantifiers. We established that quantifier monotonicity is a sufficient condition to enable a deep inference treatment of generalized quantifiers, and subsequently proposed a base generic calculus of quantifiers QC, as well as a methodology to extend it with more specific quantifiers and/or rules.

We conclude by giving an overview of the main questions we plan to study in the direct continuation of this work.

7.1 Normalization and Decomposition

Deep inference proof systems for classical first-order logic have ben proven to enjoy forms of normalization/cut-elimination [4]. Our next immediate concern is to study in which capacity these normalization methods can be applied to QC and its extensions. We expect that it will depend on the chosen extensions of QC with particular quantifiers, but we aim to give criteria on which rules can be added to the calculus while preserving normalization/cut-elimination theorems.

The question of obtaining decomposition theorems in QC, analoguous to the results in [13] (for linear logic) or [8] (for first-order logic), is closely related, and would also be of particular interest.

7.2 Completeness

The completeness of a proof system dealing with generalized quantifiers is a difficult question: it is very dependent on the specific choice of quantifiers we add to the language, and thus would have to be studied on a case-by-case basis.

In addition, most non-trivial extensions of first-order logic with generalized quantifiers are at least as expressive as incomplete fragments of second-order logic [6], so we have little hope of having completeness with regards to the usual notion of semantics. One possibility would be to use the notion of semantics and completeness of second-order logic, with partial semantics and Henkin models.

Some very restricted extensions of first-order logic with generalized quantifiers, however, do enjoy completeness: for example, first-order logic with the quantifier "uncountably many" has been shown to be complete by Keisler [9]. We can hope to propose complete proof systems based on QC for such logics.

7.3 Quantifier Calculi for well-known quantifier classes

The Quantifier Calculus QC serves as a basis for proof systems with monotone quantifiers. Many well-known quantifier classes fall in this category.

One class of particular interest is the class of Henkin Quantifiers [7], which correspond to partially-ordered prefixes of first-order quantifiers. As established in [2], there are no locally sound analytical sequent calculi for any reasonable logic with Henkin quantifier, however sequent calculi that are globally sound but locally unsound can be constructed. Henkin Quantifiers as well as their duals are monotone increasing, positive and anti-negative[7], as was first shown in [10], meaning they are suitable for the methods presented in this work. We believe it could be relevant to study if deep inference methods could provide an alternative approach to the proof theory of Henkin Quantifiers.

Another field of application is the study of the class of generalized quantifiers that correspond to natural language determiners and noun phrases. These quantifiers are almost always monotone in some capacity, or can be expressed as conjunctions of monotone quantifiers [3].

---

[6] For example, any quantifiers that can express notions of (infinite) cardinality.
[7] This is not surprising, as they are defined by means of $\forall, \exists$.

References

1 Cameron Allett. Non-Elementary Compression of First-Order Proofs in Deep Inference Using Epsilon-Terms. In *Proceedings of the 39th Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 1–14, Tallinn Estonia, July 2024. ACM. doi:10.1145/3661814.3662101.

2 Matthias Baaz and Anela Lolić. Towards a proof theory for Henkin quantifiers. *Journal of Logic and Computation*, 31(1):40–66, January 2021. doi:10.1093/logcom/exaa071.

3 Jon Barwise and Robin Cooper. Generalized quantifiers and natural language. *Linguistics and Philosophy*, 4(2):159–219, June 1981. doi:10.1007/BF00350139.

4 Kai Brünnler. Cut Elimination inside a Deep Inference System for Classical Predicate Logic. *Studia Logica*, 82(1):51–71, February 2006. doi:10.1007/s11225-006-6605-4.

5 Alessio Guglielmi. A system of interaction and structure. *ACM Trans. Comput. Logic*, 8(1), January 2007. doi:10.1145/1182613.1182614.

6 Alessio Guglielmi, Tom Gundersen, and Michel Parigot. A Proof Calculus Which Reduces Syntactic Bureaucracy. In Christopher Lynch, editor, *Proceedings of the 21st International Conference on Rewriting Techniques and Applications*, volume 6 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 135–150, Dagstuhl, Germany, 2010. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.RTA.2010.135.

7 Leon Henkin. Some remarks on infinitely long formulas. In *Infinitistic Methods: Proceedings of the Symposium on Foundations of Mathematics, Warsaw, 1959*, pages 167–183, Warsaw, 1961. Panstwowe Wydawnictwo Naukowe / Pergamon Press.

8 Dominic J. D. Hughes, Lutz Straßburger, and Jui-Hsuan Wu. Combinatorial proofs and decomposition theorems for first-order logic. In *2021 36th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–13, 2021. doi:10.1109/LICS52264.2021.9470579.

9 H. Jerome Keisler. Logic with the quantifier "there exists uncountably many". *Annals of Mathematical Logic*, 1(1):1–93, 1970. doi:10.1016/S0003-4843(70)80005-5.

10 Leszek Aleksander Kołodziejczyk. The Expressive Power of Henkin Quantifiers with Dualization. Master's thesis, Instytut Filozofii, Uniwersytet Warszawski, 2002.

11 Per Lindström. First order predicate logic with generalized quantifiers. *Theoria*, 32:186–195, 1966.

12 Andrzej Mostowski. On a generalization of quantifiers. *Fundamenta Mathematicae*, 44:12–36, 1957.

13 Lutz Straßburger. MELL in the calculus of structures. *Theoretical Computer Science*, 309(1-3):213–285, December 2003. doi:10.1016/S0304-3975(03)00240-8.