Inference with arbitrarily quantified variables: preliminary report

Marco Alberti and Evelina Lamma

Dipartimento di Ingegneria
Università degli Studi di Ferrara
Via Saragat, 1
44100 Ferrara (Italy)
[malberti|elamma]@ing.unife.it

Abstract. First order reasoning requires to solve unification problems. Most first-order unification algorithms have been designed to solve existentially quantified sets of equations, but also more general settings where equations need to be solved under a mixed quantifier prefix have been considered in the literature [Mil92]. In this work, we discuss the case of inferences with arbitrarily quantified variables. We first introduce arbitrary quantifiers (possibly mixed) over variables as *intensional* notation standing for a corresponding *extensional*, semantical counterpart. Unification of variables (possibly differently quantified) is then mapped into corresponding operations over these *extensional* counterparts. In the paper, we also give soundness results for the proposed approach, and show how resolution can be covered in the framework.

1 Introduction

First order reasoning requires to solve unification problems. Most first-order unification algorithms have been designed to solve existentially quantified sets of equations, such that:

$$\exists x_1 \dots \exists x_n \; (t_1 = s_1 \wedge \dots \wedge t_m = s_m)$$

That is, the free variables of the terms $t_1, \dots, t_m, s_1, \dots s_m$ are interpreted as existentially quantified.

Nonetheless, the possibility of mixing different quantifiers in first-order expressions allows for a higher expressibility, but requires more complex unification techniques.

In [Mil92], for instance, Miller has tackled a more general situation where equations must be solved under a mixed quantifier prefix. That is, he considers unification problems of the form:

$$Q_1 x_1 \dots Q_n x_n \; (t_1 = s_1 \wedge \dots \wedge t_m = s_m)$$

where $Q_1 \dots Q_n$ are universal and existential quantifiers. The terms $t_1, \dots, t_m$, $s_1, \dots, s_m$ can be simply typed λ-terms and the variables $x_1, \dots, x_n$ can be of primitive or functional type.

In this work, we tackle the problem of first-order reasoning with arbitrary quantifications (among them, universal and existential quantification) possibly mixed as in [Mil92].

The problem arose from a very concrete context, developed within the UE IST-2001-32530 Project, where we have adopted a Computational Logic (CL) approach for the specification and verification of interaction protocols governing agent societies.

The CL-based approach has led to the definition of a language for specifying interaction protocols where both universal and existential quantifiers co-exists, and the operational counterpart for this language requires the definition of a proper unification mechanism for variables.

After introducing the language that we rely upon in Section 2, we present in Section 2.3 the notion of quantifier as *intensional* notation standing for a corresponding *extensional*, semantical counterpart. A set-based uniform treatment of quantifiers is then discussed in Section 3; in Section 4 we define unification of quantifiers, and use it to define an inference rule.

In the paper, we also give a soundness result for the proposed approach, and also show how resolution is covered in this framework (Section 5).

Notation

- Given a n-ary predicate symbol p/n and k tuples $a_1, \ldots, a_k$ such that the arity of a_i is ϕ_i for $i = 1, \ldots, k$ and $\sum_{i=1}^{k} \phi_i = n$, $p(a_1, \ldots, a_k)$ is sometimes used as short form for $p(a_{11}, \ldots, a_{1\phi_1}, \ldots, a_{k1}, \ldots, a_{k\phi_k})$.

  Example 1.1. If $\alpha = \langle X, Y \rangle$ and $\beta = \langle T, U \rangle$, $p(\alpha, \beta)$ indicates $p(X, Y, T, U)$.
- If i and j are integer numbers with $i < j$, $[i..j]$ will sometimes indicate the set $\{i, i+1, \ldots, j\}$
- Given an n-ary predicate p and a set A of n-ples of ground terms, we will call $\bigwedge_{a \in A} p(a)$ the *associated conjunction* of A w.r.t. p.

2 A Logic Language

In this section, we introduce the logic languages that we will refer to in the remainder of the paper. We define the languages starting with a core language of which we define syntax and semantics; then we introduce quantifiers as syntactical operators.

2.1 Core Languages Syntax

Alphabet. The *alphabet* of a language $\mathcal{L}$ is composed of:

- a finite set $\mathcal{H}$ of *constant* symbols;
- a set $\mathcal{V}$ of *variable* symbols;

- a set $\mathcal{P}$ of *predicate* symbols;
- the *logical connectors* $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$.

In order to have a finite term universe, we do not allow function symbols in our language. This restriction is necessary to avoid infinitely long formulae in the definition of quantifier (see Def. 2.3); however, we may release this restriction in future work.

Well Formed Formulae. A *well formed formula* of a language $\mathcal{L}$ is defined as follows:

- $p(x_1, \ldots, x_n)$ is a wff, if $p \in \mathcal{P}$ and $x_i \in \mathcal{H} \cup \mathcal{V}$ for $i = 1, \ldots, n$;
- if p and q are wffs, then $\neg p$, $p \wedge q$, $p \vee q$, $p \rightarrow q$ and $p \leftrightarrow q$ are wffs.

Example 2.1. Throughout this paper, we will show examples based on the language $\mathcal{L}_0$, where:

- $\mathcal{H} = \mathcal{H}_0 = \{1, 2, 3, 4, 5\}$
- $\mathcal{P} = \mathcal{P}_0 = \{p, q, r\}$

2.2 Semantics

The semantics of our languages is defined as usual for first order languages (see, for instance, [And86]), except for function symbols (which we do not have) and quantifier symbols (which we define, as syntactical operators, in Sect. 2.3).

2.3 Quantifiers

In this section, we extend the languages defined in Sect. 2.1 with quantifiers, defined as syntactical operators.

Definition 2.2. *Given a core language $\mathcal{L}$, $\mathcal{L}^{\mathcal{Q}}$ has the same alphabet of $\mathcal{L}$, plus a set $\mathcal{Q}$ of* quantifier *symbols. To each $\nabla \in \mathcal{Q}$ an integer number is associated by the ϕ function, named its* arity.

The following rule is added to those defining wffs:

- *if p is a wff in which n distinct variable symbols occur, $\nabla \in \mathcal{Q}$ and $\phi(\nabla) \geq n$, then ∇p is a wff.*

Definition 2.3 allows for transforming each wff of $\mathcal{L}^{\mathcal{Q}}$ into a formula of $\mathcal{L}$.

Definition 2.3. *(Quantifier)*

An m-ary quantifier ∇ is an operator which, given an injective map $\alpha : [1..m] \rightarrow [1..n]$ and applied to an n-ary predicate $p(X_1, \ldots, X_n)$ gives an $(n-m)$-ary predicate $\nabla_\alpha p$, defined as a formula in which only p/n appears as a predicate symbol and, in each occurrence of p/n, the α_i-th argument is replaced by a ground term, for $i = 1, \ldots, m$.

The semantics of a $\mathcal{L}^{\mathcal{Q}}$ wff is the semantics of the corresponding $\mathcal{L}$ wff.

2.4 Significant quantifiers

In the following, we give the definition in our framework of the two “classical” quantifiers: the *universal* ($\forall$) and the *existential* ($\exists$).

Definition 2.4. *(Unary universal quantifier)*
For a map $\alpha : \{1\} \to [1..n]$, $1 \mapsto k$, the unary universal quantifier $\forall$ *is defined by*

$$\forall_\alpha p(X_1, \ldots, X_{k-1}, X_{k+1}, \ldots, X_n) = \bigwedge_{a \in \mathcal{H}} p(X_1, \ldots, X_{k-1}, a, X_{k+1}, \ldots, X_n) \quad (1)$$

Definition 2.5. *(Unary existential quantifier)*
For a map $\alpha : \{1\} \to [1..n]$, $1 \mapsto k$, the unary universal quantifier $\exists$ *is defined by*

$$\exists_\alpha p(X_1, \ldots, X_{k-1}, X_{k+1}, \ldots, X_n) = \bigvee_{a \in \mathcal{H}} p(X_1, \ldots, X_{k-1}, a, X_{k+1}, \ldots, X_n) \quad (2)$$

Example 2.6. Let us consider the $\mathcal{L}_0$ language, a map $\alpha : \{1\} \to \{1\}$ and a unary predicate p. In this case,

$$\forall_\alpha p = p(1) \wedge p(2) \wedge p(3) \wedge p(4) \wedge p(5)$$

and

$$\exists_\alpha p = p(1) \vee p(2) \vee p(3) \vee p(4) \vee p(5)$$

It can be easily seen that these definitions give the universal and existential quantifiers the same semantics that they have in classical first order languages.

Note 2.7. Without loss of generality, it can be assumed that $\alpha : [1..m] \to [1..n]$ is defined by $\alpha_i = i$ for $i = 1, \ldots, m$. We will do so in the following: ∇p will mean $\nabla_\alpha p$, where α is built as before.

Proof. For a generic α, $\nabla_\alpha p$ equates to $\nabla_\beta q$, with $\beta_i = i$ for $i = 1, \ldots, m$, by imposing $q(X_{\alpha_1}, \ldots, X_{\alpha_m}) = p(X_1, \ldots, X_n)$. □

Definition 2.8. *(Complete quantifier)*
Given an n-ary predicate p, an n-ary quantifier ∇ is complete *for p.*

In the remainder of the paper, quantifiers will be assumed to be complete, unless otherwise specified.

3 Set-based treatment of quantifiers

The following definitions introduce the concept of *extension* of a quantifier, which allows a uniform, set-based treatment of all quantifiers defined in our framework.

Definition 3.1. *(Instance of a quantifier)*
A set A of m-ples of ground terms is an instance *of an m-ary quantifier ∇ iff for each m-ary predicate p*

$$\bigwedge_{a \in A} p(a) \models \nabla p \tag{3}$$

Definition 3.2. *(Minimal instance of a quantifier)*
An instance A of a quantifier ∇ is minimal *if no proper subset of A is an instance of ∇.*

In the following, unless otherwise specified, we will always consider minimal instances.

3.1 Instances of universal and existential quantifiers

An instance of the universal quantifier $\forall$ is $\mathcal{H}$. This instance is also easily shown to be minimal and, thus, unique, since any other instance, being a set of ground terms, would have to be included in $\mathcal{H}$.

Any set A of ground terms such that $A \neq \emptyset$ and $A \subseteq \mathcal{H}$ is an instance of the existential quantifier $\exists$. An instance of $\exists$ is minimal if it contains only one element.

Definition 3.3. *(Extension of a quantifier)*
The extension *of a quantifier is the set of all its minimal instances.*

Example 3.4. (Extensions of universal and existential quantifiers)
The extension of $\forall$ is $\{\mathcal{H}\}$.
The extension of $\exists$ is $\{\{a\} : a \in \mathcal{H}\}$

The following proposition provides a link between the extension of a quantifier and its logical meaning.

Proposition 3.5. *Let $\mathcal{A}$ be the extension of a quantifier ∇. Then ∇p is logically equivalent to $\bigvee_{A \in \mathcal{A}} \bigwedge_{a \in A} p(a)$.*

Proof.
1. $\bigvee_{A \in \mathcal{A}} \bigwedge_{a \in A} p(a) \models \nabla p$
   A model of the disjunction is a model of at least one of the conjunctions, and thus of ∇p, by definition of instance.
2. $\nabla p \models \bigvee_{A \in \mathcal{A}} \bigwedge_{a \in A} p(a)$
   Let us write ∇p in disjunctive normal form, as $\bigvee_{B \in \mathcal{B}} \bigwedge_{b \in B} p(b)$
   Each B in $\mathcal{B}$ is obviously an instance of ∇. Thus, its associated conjunction entails the associated conjunction of a minimal instance (any one that is included in B) and, thus, it also entails $\bigvee_{A \in \mathcal{A}} \bigwedge_{a \in A}$.

$\square$

This proposition shows that different quantifiers with the same extensions are equivalent at the semantical level. Formally, we can define an equivalence relation between quantifiers which holds between quantifiers with the same extension, and deal with the quotient set w.r.t. this relation. However, for the aim of this paper, it is sufficient to understand that quantifiers with the same extension are, in practice, the same quantifier.

The following definition allows the expression of quantification over subsets of the universe.

Definition 3.6. *(Restricted quantifiers)*

Given a quantifier ∇ with extension $\mathcal{A}$ and a set $S \in \mathcal{H}$, ∇^S ("∇ restricted to *S") is the quantifier whose extension is:*

$$\{B\colon A \in \mathcal{A}, B = A \cap S, B \neq \emptyset\} \tag{4}$$

Example 3.1. – Restricted unary universal quantifier: given $\mathcal{I} \subseteq \mathcal{H}$,

$$\forall^{\mathcal{I}} p = \bigwedge_{a \in \mathcal{I}} p(a) \tag{5}$$

– Restricted unary existential quantifier: given $\mathcal{I} \subseteq \mathcal{H}$,

$$\exists^{\mathcal{I}} p = \bigvee_{a \in \mathcal{I}} p(a) \tag{6}$$

3.2 Composing quantifiers

Def. 3.7 introduces *composed* quantifiers (such as $\forall\exists$), defined in terms of their components.

Definition 3.7. *(Composed quantifier)*

Let ∇_1 be an m_1-ary quantifier and ∇_2 be an m_2-ary quantifier.

Then the composition *of ∇_1 and ∇_2 is an $(m_1 + m_2)$-ary quantifier $(\nabla_1\nabla_2)$ defined by*

$$(\nabla_1\nabla_2)p = \nabla_1(\nabla_2 p) \tag{7}$$

By definition, the expression for the composed quantifier is:

$$(\nabla_1\nabla_2)p = \bigvee_{A \in \mathcal{A}} \bigwedge_{a \in A} \bigvee_{B \in \mathcal{B}} \bigwedge_{b \in B} p(a, b) \tag{8}$$

Prop. 3.9 will show another expression, which allows to single out the instances of the composed quantifier. The proof of Prop. 3.9 requires the following lemma:

Lemma 3.8.

$$\bigwedge_{a \in A} \bigvee_{b \in B} p(a, b) = \bigvee_{f \in \mathcal{F}_{A,B}} \bigwedge_{a \in A} p(a, f(a)) \tag{9}$$

where $\mathcal{F}_{A,B}$ is the set of all functions from A to B.

Proof. By distributing $\wedge$ over $\vee$, we obtain the disjunction of all the possible conjunctions which have one conjunct for each $a \in A$, of the form $p(a, b)$, where b is an element of B. The possible conjunctions are exactly the possible ways of associating an element of B to each element of A, i.e., $\mathcal{F}_{A,B}$.

$\square$

Proposition 3.9. *Let ∇_1 and ∇_2 be quantifiers, and $\mathcal{A}$ and $\mathcal{B}$ their extensions, respectively. Then*

$$(\nabla_1 \nabla_2)p = \bigvee_{A \in \mathcal{A}} \bigvee_{f \in \mathcal{F}_{A,\mathcal{B}}} \bigwedge_{a \in A} \bigwedge_{b \in f(a)} p(a, b) \tag{10}$$

Proof. Define $q_a(B) = \bigwedge_{b \in B} p(a, b)$. Then Eq. (8) can be written as:

$$(\nabla_1 \nabla_2)p = \bigvee_{A \in \mathcal{A}} (\bigwedge_{a \in A} \bigvee_{B \in \mathcal{B}} q_a(B)) \tag{11}$$

Applying Lemma 3.8 to the part of Eq. (11) in brackets, we can write:

$$(\nabla_1 \nabla_2)p = \bigvee_{A \in \mathcal{A}} (\bigvee_{f \in \mathcal{F}_{A,\mathcal{B}}} \bigwedge_{a \in A} q_a(f(a))) \tag{12}$$

and finally, by definition of $q_a(f(a))$, Eq. (10).

$\square$

Eq. (10) is useful because it shows explicitly the instances of the composed quantifiers (the sets $\{\langle a, b \rangle : a \in A, b \in f(a)\}$, each determined by the choice of an $A \in \mathcal{A}$ and an $f \in \mathcal{F}_{A,\mathcal{B}}$).

Example 3.10. Let $\nabla_1 = \forall^{\{1,2\}}$ and $\nabla_2 = \exists^{\{3,4\}}$

– $\nabla_1 \nabla_2$
  Here, $\mathcal{A} = \{\{1, 2\}\}$, so the only possible choice for $A \in \mathcal{A}$ is $A = \{1, 2\}$. There are four possible choices for f:
  - $f(1) = \{3\}, f(2) = \{3\}$
  - $f(1) = \{3\}, f(2) = \{4\}$
  - $f(1) = \{4\}, f(2) = \{3\}$
  - $f(1) = \{4\}, f(2) = \{4\}$

  The instances of $\nabla_1 \nabla_2$ are thus the following four:
  - $\{\langle 1, 3\rangle, \langle 2, 3\rangle\}$
  - $\{\langle 1, 3\rangle, \langle 2, 4\rangle\}$
  - $\{\langle 1, 4\rangle, \langle 2, 3\rangle\}$
  - $\{\langle 1, 4\rangle, \langle 2, 4\rangle\}$

– $\nabla_2 \nabla_1$
  In this case, $\mathcal{A} = \{\{3\}, \{4\}\}$ and the possible choices for $A \in \mathcal{A}$ are $A = \{3\}$ and $A = \{4\}$. Chosen A, there is only one choice for f: $f(a) = \{1, 2\}$, where a is the only element of A. Thus, the instances of $\nabla_2 \nabla_1$ are:
  - $\{\langle 3, 1\rangle \langle 3, 2\rangle\}$
  - $\{\langle 4, 1\rangle \langle 4, 2\rangle\}$

3.3 An order relation over quantifiers

In the following, we define a set-based partial order relation which semantically maps into entailment.

Definition 3.11. *(Weaker quantifier)*
$\nabla_1 \preceq \nabla_2$ *iff each instance of* ∇_2 *has a non-empty subset that is an instance of* ∇_1.

Proposition 3.12. *If* $\nabla_1 \succeq \nabla_2$*, then for any predicate* p $\nabla_1 p \models \nabla_2 p$

Proof. A model of $\nabla_1 p$ is also a model of the associated conjunction of an instance (say A) of ∇_1, because of Prop. 3.5. Since $\nabla_1 \succeq \nabla_2$, there exists a subset of A that is an instance of ∇_2, and whose associated conjunction entails $\nabla_2 p$. □

4 An inference rule

In this section, we propose (and prove sound) an inference rule which can be applied, under the conditions of Def. 4.1, to predicates with different quantifications.

Definition 4.1. *(Unifiable quantifiers)*
Two quantifiers ∇_1 *(whose extension is* $\mathcal{A}$*) and* ∇_2 *(whose extension is* $\mathcal{B}$*) are said to be* unifiable *iff*

$$A \in \mathcal{A}, B \in \mathcal{B} \rightarrow A \cap B \neq \emptyset \tag{13}$$

Example 4.2. $\forall^{[1..4]}$ *and* $\forall^{[2..5]}$ *are unifiable;* $\exists^{[1..4]}$ *and* $\exists^{[2..5]}$ *are not.*

Definition 4.3. *(Unification of quantifiers)*
Let ∇_1 *and* ∇_2 *be two unifiable quantifiers. Their* unification *is a quantifier* $(\nabla_1 \cup \nabla_2)$ *whose extension is given by:*

$$\{C : A \in \mathcal{A}, B \in \mathcal{B}, C = A \cap B\} \tag{14}$$

Example 4.4. Let $\nabla_1 = \forall^{[1..3]}$ and $\nabla_2 = \exists^{[2..3]}$. The extension of ∇_1 is $\{\{1,2,3\}\}$ and the extension of ∇_2 is $\{\{2\},\{3\}\}$: ∇_1 and ∇_2 are unifiable and the extension of their unification is $\{\{2\},\{3\}\}$, which is the extension of $\exists^{[2..3]}$. Thus,

$$\forall^{[1..3]} \sqcap \exists^{[2..3]} = \exists^{[2..3]} \tag{15}$$

The following proposition proves sound an inference rule which, given two predicates with (possibly) different quantifications, derives their conjunction, quantified with the unification of the original quantifiers.

Proposition 4.5. *Let* ∇_1 *and* ∇_2 *be two unifiable quantifiers. Then*

$$\nabla_1 p \wedge \nabla_2 q \models (\nabla_1 \sqcap \nabla_2)(p \wedge q) \tag{16}$$

∇_1	∇_2	Unifiable?	$\nabla_1 \sqcap \nabla_2$
$\forall$	$\forall$	yes	$\forall$
$\forall$	$\exists$	yes	$\exists$
$\exists$	$\exists$	no	-
$\forall^{S_1}$	$\forall^{S_2}$	if $S_1 \cap S_2 \neq \emptyset$	$\forall^{S_1 \cap S_2}$
$\forall^{S_1}$	$\exists^{S_2}$	if $S_2 \subseteq S_1$	$\exists^{S_2}$

Table 1. Unification of particular quantifiers

Proof. A model of $\nabla_1 p \wedge \nabla_2 q$ is a model of the associated conjunction of at least one instance of ∇_1 (name it A) w.r.t. p and of the associated conjunction of one instance of ∇_2 (name it B) w.r.t. q. Therefore, it is the model of the associated conjunction of $A \cap B$ w.r.t $p \wedge q$. As $A \cap B$ is an instance of $(\nabla_1 \sqcap \nabla_2)$, a model of its associated conjunction w.r.t $p \wedge q$ is also a model of $(\nabla_1 \sqcap \nabla_2)(p \wedge q)$.

□

Obviously, in practice, we do not want to compute the unification of two quantifiers directly by the definition each time we need to make an inference. The following proposition provides some ready-to-use results.

Proposition 4.6. *Tab. 1 summarises particular cases of unification between quantifiers.*

Proof. By definition of unification.

The following proposition shows that the unification of composed quantifiers equates to the composition of the unification of the components. Together with Prop. 4.6, it allows for building the unification of composed quantifiers without considering their extensions.

Proposition 4.7.

$$(\nabla_1 \nabla_2) \sqcap (\diamondsuit_1 \diamondsuit_2) = (\nabla_1 \sqcap \diamondsuit_1)(\nabla_2 \sqcap \diamondsuit_2) \tag{17}$$

Proof. We will prove that $(\nabla_1 \nabla_2) \sqcap (\diamondsuit_1 \diamondsuit_2)$ and $(\nabla_1 \sqcap \diamondsuit_1)(\nabla_2 \sqcap \diamondsuit_2)$ have the same instances.

Let $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{B}_1$ and $\mathcal{B}_2$ be the extensions of ∇_1, ∇_2, $\diamondsuit_1$ and $\diamondsuit_2$, respectively.

– An instance of $(\nabla_1 \nabla_2)$ is determined by an $A_1 \in \mathcal{A}_1$ and by an $f \in \mathcal{F}_{A_1, \mathcal{A}_2}$, and has the form:

$$\{\langle a_1, a_2 \rangle : a_1 \in A_1, a_2 \in f(a_1)\} \tag{18}$$

Similarly, an instance of $(\diamondsuit_1 \diamondsuit_2)$ is defined by a $B_1 \in \mathcal{B}_1$ and a $g \in \mathcal{F}_{B_1, \mathcal{B}_2}$, and has the form:

$$\{\langle b_1, b_2 \rangle : b_1 \in B_1, b_2 \in g(b_1)\} \tag{19}$$

Each instance of $(\nabla_1 \nabla_2) \sqcap (\diamondsuit_1 \diamondsuit_2)$ is the intersection of an instance of $(\nabla_1 \nabla_2)$ and an instance of $(\diamondsuit_1 \diamondsuit_2)$. Thus, it is determined by an $A_1 \in \mathcal{A}_1$, a $B_1 \in \mathcal{B}_1$, an $f \in \mathcal{F}_{A_1, \mathcal{A}_2}$, and a $g \in \mathcal{F}_{B_1, \mathcal{B}_2}$, and has the form:

$$\{\langle c_1, c_2 \rangle : c_1 \in (A_1 \cap B_1), c_2 \in (f(c_1) \cap g(c_1))\} \tag{20}$$

– Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be the extensions of $(\nabla_1 \sqcap \diamondsuit_1)$ and $(\nabla_2 \sqcap \diamondsuit_2)$, respectively. An instance of $(\nabla_1 \sqcap \diamondsuit_1)(\nabla_2 \sqcap \diamondsuit_2)$ is determined by a $C_1 \in \mathcal{C}_1$ and a $h \in \mathcal{F}_{C_1,\mathcal{C}_2}$, and has the form:

$$\{\langle c_1, c_2\rangle \colon c_1 \in C_1, c_2 \in h(c_1)\} \tag{21}$$

However, the extensions of $(\nabla_1 \sqcap \diamondsuit_1)$ and $(\nabla_2 \sqcap \diamondsuit_2)$ have the form:

$$\begin{aligned} \mathcal{C}_1 =& \{C_1 \colon A_1 \in \mathcal{A}_1, B_1 \in \mathcal{B}_1, C_1 = A_1 \cap B_1\} \\ \mathcal{C}_2 =& \{C_2 \colon A_2 \in \mathcal{A}_2, B_2 \in \mathcal{B}_2, C_2 = A_2 \cap B_2\} \end{aligned} \tag{22}$$

Eq. (22) shows that each $C_1 \in \mathcal{C}_1$ is determined by an $A_1 \in \mathcal{A}_1$ and a $B_1 \in \mathcal{B}_1$: those (possibly non unique) such that $A_1 \cap B_1 = C_1$. Each $h \in \mathcal{F}_{C_1,\mathcal{C}_2}$ is, in turn, determined by an $f \in \mathcal{F}_{A_1,\mathcal{A}_2}$ and a $g \in \mathcal{F}_{B_1,\mathcal{B}_2}$: namely, those (possibly non unique) that respectively map each $c_1 \in C_1$ to an $A_2 \in \mathcal{A}_2$ and a $B_2 \in \mathcal{B}_2$ such that $A_2 \cap B_2 = h(c_1)$.
In conclusion, each instance of $(\nabla_1 \sqcap \diamondsuit_1)(\nabla_2 \sqcap \diamondsuit_2)$ is determined by an $A_1 \in \mathcal{A}_1$, a $B_1 \in \mathcal{B}_1$, an $f \in \mathcal{F}_{A_1,\mathcal{A}_2}$, and a $g \in \mathcal{F}_{B_1,\mathcal{B}_2}$, and has the form:

$$\{\langle c_1, c_2\rangle \colon c_1 \in (A_1 \cap B_1), c_2 \in (f(c_1) \cap g(c_1))\} \tag{23}$$

Summarizing, each instance of one member of Eq. (17) is determined by an $A_1 \in \mathcal{A}_1$, a $B_1 \in \mathcal{B}_1$, an $f \in \mathcal{F}_{A_1,\mathcal{A}_2}$, and a $g \in \mathcal{F}_{B_1,\mathcal{B}_2}$ that also determines an instance of the other member, in the same way (shown by Eq. (20) and Eq. (23)). In other words, the two members of Eq. (17), having the same instances, are the same quantifier.

$\square$

5 Application to resolution

In this section, we apply the inference rule introduced in Sect. 4 to a kind of resolution that operates on disjunctions with arbitrary quantfications.

Proposition 5.1. *Let ∇_1 and ∇_2 be unifiable. Then*

$$\nabla_1(\neg p \vee q) \wedge \nabla_2(p \vee r) \models (\nabla_1 \sqcap \nabla_2)(q \vee r) \tag{24}$$

Proof.

$$\nabla_1(\neg p \vee q) \wedge \nabla_2(p \vee r) \models (\nabla_1 \sqcap \nabla_2)((\neg p \vee q) \wedge (p \vee r))$$

because of Prop. 4.5. Let $\mathcal{A}$ be the extension of $(\nabla_1 \sqcap \nabla_2)$; then, because of Prop. 3.5,

$$(\nabla_1 \sqcap \nabla_2)((\neg p \vee q) \wedge (p \vee r)) \leftrightarrow \bigvee_{A \in \mathcal{A}} \bigwedge_{a \in A} ((\neg p(a) \vee q(a)) \wedge (p(a) \vee r(a)))$$

By applying propositional resolution to each conjunct,

$$\bigvee_{A\in\mathcal{A}}\bigwedge_{a\in A}((\neg p(a)\vee q(a))\wedge(p(a)\vee r(a)))\models\bigvee_{A\in\mathcal{A}}\bigwedge_{a\in A}(q(a)\vee r(a))$$

which, using Prop. 3.5 again, can be rewritten as

$$(\nabla_1\sqcap\nabla_2)(q\vee r)$$

□

Example 5.2. Let us consider the following two formulae:

$$\forall^{[1..4]}\forall^{[1..3]}(\neg p\vee q) \tag{25}$$

$$\forall^{[2..5]}\exists^{[2..3]}p \tag{26}$$

By Tab. 1, $\forall^{[1..4]}\sqcap\forall^{[2..5]}=\forall^{[2..4]}$ and $\forall^{[1..3]}\sqcap\exists^{[2..3]}=\exists^{[2..3]}$. Thus (Eq. (17)) $\forall^{[1..4]}\forall^{[1..3]}\sqcap\forall^{[2..5]}\exists^{[2..3]}=\forall^{[2..4]}\exists^{[2..3]}$ and Eqs. (25) and (26), because of Prop. 4.5, entail

$$\forall^{[2..4]}\exists^{[2..3]}((\neg p\vee q)\wedge p) \tag{27}$$

and finally

$$\forall^{[2..4]}\exists^{[2..3]}q \tag{28}$$

6 Discussion

The work presented in this paper is still at an early stage, and several points remain to be defined. In particular, we believe that our work, in its current status, has two main limitations, that we are currently trying to tackle.

- First of all, the inference rule presented in Sect. 4 can be applied only if a strong condition holds: the two original quantifiers have to be unifiable. However, it seems possible to perform reasoning steps even in many cases where the quantifiers are not unifiable. Consider, for instance, the case of two existential quantifiers[1]:

  $$\exists X p(X)\wedge\exists Y q(Y) \tag{29}$$

  the natural step would be to perform a *case analysis*, i.e., to branch the reasoning into two cases:
  - in one case, the arguments of the predicates are constrained to be equal ($X=Y$), which makes their quantifier unifiable, and possible to infer $\exists X p(X)\wedge q(X)$;
  - in the other case, where $X\neq Y$, no inference is performed.

  We are currently working on formalizing and dealing in a uniform way with such cases.

[1] In the following formulae, we express variable quantification explicitely.

– We have not yet defined how we deal with quantifiers with different arity, as in the following case:

$$\begin{aligned} &\forall X \exists Y p(X) \rightarrow q(Y) \\ &\exists Z p(Z) \end{aligned} \tag{30}$$

where we would obviously like to infer $\exists T q(T)$. One possible way would be to extend the quantification of the second formula with a "dummy" universally quantified variable, to obtain $\exists Z \forall U p(Z)$, which would work in this particular case because $\forall\exists$ is unifiable with $\exists\forall$ and their unification results in $\exists\exists$.
We are currently researching a result that generalizes this kind of reasoning.

7 Conclusions

In this paper, we have presented a logic language where quantifiers can be arbitrary and are defined as syntactical operators. We have introduced a set-based theory of quantifiers which allows performing inference over sets of formulae with arbitrary quantification, and proved a soundness result for the inference rule that we have introduced; finally, we have shown how our approach can be applied to resolution.

Acknowledgements

This work is partially funded by the Information Society Technologies programme of the European Commission under the IST-2001-32530 SOCS Project [SOC], and by the MIUR COFIN 2003 project *La Gestione e la negoziazione automatica dei diritti sulle opere dell'ingegno digitali: aspetti giuridici e informatici*.

References

[And86] Peter B. Andrews. *An Introduction to Mathematical Logic and Type Theory: to Truth through Proof.* Academic Press, Orlando, Florida, 1986.
[Mil92] Dale Miller. Unification under a mixed prefix. *Dale Miller*, 14:321–358, 1992.
[SOC] Societies Of ComputeeS (SOCS): a computational logic model for the description, analysis and verification of global and open societies of heterogeneous computees. http://lia.deis.unibo.it/Research/SOCS/.