A Revised Projectivity Calculus for Inclusion and Exclusion Reasoning*

Ka-fat CHOW
The Hong Kong Polytechnic University
kfzhouy@yahoo.com

Abstract

We present a Revised Projectivity Calculus (denoted RC) that extends the scope of inclusion and exclusion inferences derivable under the Projectivity Calculus (denoted C) developed by Icard (2012). After pointing out the inadequacies of C, we introduce four opposition properties (OPs) which have been studied by Chow (2012, 2017) and are more appropriate for the study of exclusion reasoning. Together with the monotonicity properties (MPs), the OPs will form the basis of RC instead of the additive/multiplicative properties used in C. We also prove some important results of the OPs and their relation with the MPs. We then introduce a set of projectivity signatures together with the associated operations and conditions for valid inferences, and develop RC by inheriting the key features of C. We then show that under RC, we can derive some inferences that are not derivable under C. We finally discuss some properties of RC and point to possible directions of further studies.

Keywords: inclusion; exclusion; opposition properties; projectivity signatures; Natural Logic

1. Introduction

1.1 Overview

In recent years, there arises an interest among some scholars (e.g. MacCartney (2009), MacCartney and Manning (2009), Icard (2012, 2014), Icard and Moss (2014), Chow (2012, 2017) in studying natural language reasoning involving the exclusion relations (to be defined below) as an extension of the study of reasoning involving the inclusion relations (also called monotonicity inferences) studied under modern Generalized Quantifier Theory (GQT) and Natural Logic (or Natural Language Inferences). Among the various approaches, Icard's Projectivity Calculus

* This is a post-peer-review, pre-copyedit version of an article published in *Journal of Logic, Language and Information*. The final authenticated version is available online at http://dx.doi.org/10.1007/s10849-019-09292-5.

is interesting because it extends the Monotonicity Calculus first developed by van Benthem (1986) and Sánchez Valencia (1991), and deals with inclusion and exclusion reasoning under the same system. An example of this mixed type of reasoning is given below (discussed in detail in Icard (2012)):

(1) Every job that involves a giant squid is dangerous ≤
Not every job that involves a cephalopod is safe

The inference above involves the contradictory relation between *every* and *not every*, the exclusive relation between *dangerous* and *safe*, the subset relation between *squid* and *cephalopod* and the overall entailment relation (similar to the subset relation from the point of view of Boolean algebra) between the two sentences.

While the Projectivity Calculus developed by Icard (2012), which is denoted "C", can handle well exclusion reasoning of sentences that involve the classical quantifiers, (i.e. *every*, *some*, *no* and *not every*) as well as some logical operators (e.g. *if*, *not*)[1], it fails to derive exclusion inferences that involve generalized quantifiers. The problem is not that C fails to derive all exclusion inferences, which is not surprising as Icard (2012) has already shown that C is incomplete[2], but that it is based on the additive, multiplicative, anti-additive and anti-multiplicative properties, which are not appropriate notions for studying exclusion reasoning.

In this paper, we will introduce four properties studied by Chow (2012, 2017) to replace the four notions used in C. We will show that these new properties are more appropriate for studying exclusion reasoning. We will then propose a Revised Projectivity Calculus and show that this revised system can derive some inclusion and exclusion inferences that are not derivable under C.

1.2 Some Basic Notions

Before discussing the inadequacies of C, we first introduce some basic notions used in C. These include relations between elements in a Boolean algebra and properties of functions on Boolean algebras. Given a Boolean algebra **B** = (B, ¬, ∨, ∧,

---

[1] Icard (2012) claimed that the quantifier *most* is multiplicative. But it will be shown below that *more than 1/2 of* is not multiplicative. Note that many GQT scholars, such as Peters and Westerståhl (2006), assume that *most* is equivalent to *more than 1/2 of.*

[2] Icard (2014) provided an axiomatization for the inclusion and exclusion reasoning and conjectured that the resulting calculus (called C_2) is complete (but without proving it). Since C_2 does not contain as much detail as the calculus C developed in Icard (2012), the discussion of this paper is mainly based on Icard (2012).

0, 1), we can define seven relations between elements in B. These seven relations comprise a set which is denoted by R. The names, symbols and definitions of the members of R are set out in the following table. In the following definitions, x and y are elements of B.

Definition 1.1[3]:

Name	Symbol	Definition
subset	$\leq$	$x \leq y$ iff $x \wedge y = x$
superset	$\geq$	$x \geq y$ iff $x \vee y = x$
exclusive	Δ	$x \,\Delta\, y$ iff $x \wedge y = 0$
exhaustive	∇	$x \,\nabla\, y$ iff $x \vee y = 1$
equivalent	$\equiv$	$x \equiv y$ iff $x \leq y$ and $x \geq y$
contradictory	$\perp$	$x \perp y$ iff $x \,\Delta\, y$ and $x \,\nabla\, y$
general	#	$x \# y$ iff $x \in B$ and $y \in B$

Among the seven relations above, the first four relations are the basic ones, with the subset and superset relations being the inclusion relations and the exclusive and exhaustive relations being the exclusion relations. The definitions of the equivalent and contradictory relations are based on those of the four basic ones. The general relation is the most uninformative relation. It exists between any two elements of B. Moreover, from Definition 1.1, one can deduce the following proposition[4].

Proposition 1.2: For all x and y of a Boolean Algebra,

(i) $x \,\Delta\, y$ iff $x \leq \neg y$.

(ii) $x \,\nabla\, y$ iff $\neg x \leq y$.

By using the symmetry of Δ and ∇ and the contrapositivity of $\leq$ and $\geq$, one can deduce even more propositions, which will not be listed here.

In the above, we have used the standard symbols $\vee$, $\wedge$, 0, 1, $\leq$, $\geq$ for general

---

[3] The names and symbols of the relations are adapted from MacCartney (2009), MacCartney and Manning (2009) and Icard (2012). The definitions are taken from Icard (2012). The "exclusive" and "exhaustive" relations are the same as the "CC" (short for "contrary or contradictory") and "SC" (short for "subcontrary or contradictory") relations in Chow (2012, 2017). Moreover, in what follows, "iff" is short for "if and only if". Note that we have chosen to use Δ and ∇ so that the shape of Δ reminds us of the "$\wedge$" in the definition of the exclusive property whereas the shape of ∇ reminds us of the "$\vee$" in the definition of the exhaustive property.

[4] The following proposition can be easily proved by invoking the fact that $x \leq y$ iff $x \wedge \neg y = 0$ and de Morgan's Law. The above fact can be found in Keenan and Faltz (1985).

Boolean algebras. But in some cases (especially when talking about the truth conditions of generalized quantifiers), it would be more convenient to use the corresponding symbols in set theory, i.e. $\cup$, $\cap$, $\varnothing$, U^*, $\subseteq$, $\supseteq$ (note that all the subsets of a universe U^* form a Boolean algebra). Hence, the definitions of Δ and ∇ can also be stated in set notation as "X Δ Y iff X $\cap$ Y = $\varnothing$" and "X ∇ Y iff X $\cup$ Y = U^*". Both the standard notation and set notation will be used in this paper depending on the context.

Note that the universe U^* discussed above has to be understood in a relative sense depending on the context. In case the proposition under discussion involves entities of different types, the entire universe should be seen as comprising a number of sub-universes and ∇ is defined with respect to these sub-universes. For example, suppose we are considering a proposition about "clubs" and "members". Then the universe associated with this proposition should be seen as comprising two sub-universes, namely the set of all "clubs" and the set of all "members". Although {x: x is a member} $\neq U^*$ (because U^* also contains "clubs"), we should have {x: x is a male member} ∇ {x: x is a female member} under certain context, because {x: x is a male member} $\cup$ {x: x is a female member} = {x: x is a member}, where {x: x is a member} is a sub-universe in this case.

As pointed out in Icard (2012), there is a natural ordering among the members of R, which is depicted by Fig. 1.

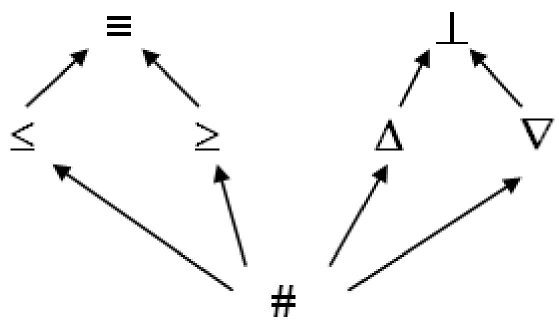

Fig. 1 Partial order of members of R

In Fig. 1, an arrow pointing from a relation ρ_1 to another relation ρ_2 means that ρ_2 is a stronger relation than ρ_1, i.e. for all x and y $\in$ B, x ρ_2 y entails x ρ_1 y. If we now use $\leq$ to represent the "stronger than" relation, then Fig. 1 can also be viewed as representing a partial order comprising the members of R. We can also define the meet ($\wedge$) between some of the members of R and so we have $\equiv$ = $\leq \wedge \geq$ and $\perp = \Delta \wedge \nabla$. Moreover, it can be easily seen that the members of R satisfy the following proposition.

Proposition 1.3: Let $\rho_1, \rho_2 \in$ R. Then for all x and y, x ρ_1 y and x ρ_2 y iff x ($\rho_1 \wedge \rho_2$) y.

We next define the JOIN operation[5] on the members of R as follows (adapted from Icard (2012).

Definition 1.4: Let $\rho_1, \rho_2 \in$ R. The JOIN of ρ_1 and ρ_2, denoted ρ_1 JOIN ρ_2, is the strongest member of R such that if x ρ_1 y and y ρ_2 z, then x (ρ_1 JOIN ρ_2) z.

The results of the JOIN operation among the members of R are given in the following table (adapted from Icard (2012)):

Table 1 The JOIN Operation

JOIN	≤	≥	≡	**Δ**	**∇**	⊥	#
≤	≤	#	≤	Δ	#	Δ	#
≥	#	≥	≥	#	∇	∇	#
≡	≤	≥	≡	Δ	∇	⊥	#
Δ	#	Δ	Δ	#	≤	≤	#
∇	∇	#	∇	≥	#	≥	#
⊥	∇	Δ	⊥	≥	≤	≡	#
#	#	#	#	#	#	#	#

Note that ≡ and # behave like 1 and 0 in the ordinary multiplication, respectively.

The JOIN operation enables us to derive inclusion inferences by combining exclusion inferences appropriately. For example, from *awake* Δ *asleep* and *asleep* ⊥ *not asleep*, we have, by Definition 1.4, *awake* (Δ JOIN ⊥) *not asleep*. From Table 1, we obtain *awake* ≤ *not asleep*.

We can also define properties of functions on Boolean algebras. The following two properties are the most basic ones in the study of Monotonicity Calculus.

[5] Note that the JOIN operation defined by Icard (2012) is different from the concept of "join" in a partial order. To highlight this difference, the symbol representing this special operation is capitalized in this paper.

Definition 1.5[6]: Let f: B → B' be a function on Boolean algebras, x and y be elements of B.

(i) f is (monotone) increasing iff $x \leq y$ entails $f(x) \leq f(y)$, or equivalently $x \geq y$ entails $f(x) \geq f(y)$.
(ii) f is (monotone) decreasing iff $x \leq y$ entails $f(x) \geq f(y)$, or equivalently $x \geq y$ entails $f(x) \leq f(y)$.
(iii) f is monotonic iff f is either increasing or decreasing; f is non-monotonic iff f is neither increasing nor decreasing.

Apart from the monotonicity properties, Icard (2012) also defined the following four properties, which can be seen as strengthened versions of the increasing or decreasing properties.

Definition 1.6[7]: Let f: B → B' be a function on Boolean algebras, x and y be elements of B.

(i) f is additive iff $f(x \vee y) \equiv f(x) \vee f(y)$, and completely additive iff in addition $f(1) \equiv 1$.
(ii) f is multiplicative iff $f(x \wedge y) \equiv f(x) \wedge f(y)$, and completely multiplicative iff in addition $f(0) \equiv 0$.
(iii) f is anti-additive iff $f(x \vee y) \equiv f(x) \wedge f(y)$, and completely anti-additive iff in addition $f(1) \equiv 0$.
(iv) f is anti-multiplicative iff $f(x \wedge y) \equiv f(x) \vee f(y)$, and completely anti-multiplicative iff in addition $f(0) \equiv 1$.

For convenience, the above properties will be collectively called "+-× properties". They serve as the cornerstone of Icard's Projectivity Calculus.

Icard (2012) also proved the following two propositions, which relate the monotonicity properties and the +-× properties.

Proposition 1.7: Let f: B → B' be a function on Boolean algebras. The following are equivalent:

(i) f is increasing;
(ii) $f(x) \vee f(y) \leq f(x \vee y)$;
(iii) $f(x \wedge y) \leq f(x) \wedge f(y)$.

---

6 Icard (2012) used the terms "monotone" and "antitone" instead of "increasing" and "decreasing".
7 What Icard (2012) studied is the "complete" version of the properties.

Proposition 1.8: Let f: B → B' be a function on Boolean algebras. The following are equivalent:

(i) f is decreasing;

(ii) $f(x \vee y) \leq f(x) \wedge f(y)$;

(iii) $f(x) \vee f(y) \leq f(x \wedge y)$.

These two propositions show that the increasing property satisfies part of the definitions of the additive/multiplicative properties whereas the decreasing property satisfies part of the definitions of the anti-additive/anti-multiplicative properties, thus showing that the +-× properties are indeed strengthening of the monotonicity properties.

1.3 Inadequacies of C

While the +-× properties are important properties of the classical quantifiers, they do not cover some generalized quantifiers which satisfy important exclusion inferences. In this subsection, we will discuss several types of such quantifiers.

But before discussing these quantifiers, we first explain briefly how we represent the argument structures of quantifiers. In this paper, quantified statements are represented in the curried form $Q(x_1)(x_2)...(x_n)$ where Q represents the quantifier and x_1, x_2, ... x_n represent the 1^{st}, 2^{nd}, ... n^{th} argument of the quantifier. The currying operation has in fact transformed an n-ary function to n successive applications of unary functions. As a result, we only need to deal with unary functions in this paper, although for convenience we will often talk about "the n^{th} argument" of a quantifier. Moreover, to eliminate parentheses, we associate functional application to the left. Thus, $Q(x_1)(x_2)$ should be seen as a simplified form of $(Q(x_1))(x_2)$.

We now discuss proportional quantifiers such as *more than r of*, *less than r of* ($0 < r < 1$), etc. These quantifiers satisfy certain exclusion inferences, such as the following[8]:

(2) More than 1/2 of the members are elderly Δ
More than 1/2 of the members are teenagers

This inference is valid because the two sentences above cannot be both true. It can be

[8] In this paper, we treat proportional quantifiers *more than / less than / at least / at most r of the* as equivalent to *more than / less than / at least / at most r of.*

proved that *more than 1/2 of* does not possess any of the four +-× properties in both of its arguments. For example, to prove that it is not multiplicative in the 2nd argument, we can construct the following counterexample. Let z = {a, b, c}, x = {a, b}, y = {b, c}. Then x ∧ y = {b}, and we have ‖*more than 1/2 of*‖(z)(x) ∧ ‖*more than 1/2 of*‖(z)(y) true but ‖*more than 1/2 of*‖(z)(x ∧ y) false. Similarly, one can show that *more than 1/2 of* does not possess any of the remaining three +-× properties in both of its arguments by constructing counterexamples. In view of this, the inference in (2) cannot be derived under C.

In what follows, we will discuss some quantifiers that are non-monotonic in one or more of its arguments. Given Propositions 1.7 and 1.8, these quantifiers do not possess any of the +-× properties in respect of their non-monotonic argument. Yet it can be shown that they satisfy certain types of exclusion inferences in respect of these arguments.

The first type includes proportional quantifiers such as *exactly r of*, *between q and r of*, *more than r or less than q of*, etc. where q and r represent a fraction or percentage. It can easily be shown that these quantifiers are non-monotonic in both of their arguments. Yet these quantifiers satisfy different types of exclusion inferences depending on the values of q and r. For example, it is easily seen that the following inference is valid:

(3) Exactly 3/4 of the members are elderly Δ
Exactly 3/4 of the members are teenagers

The second type includes exceptive quantifiers such as *all ... except Smith*, *no .... except Smith*, etc[9]. It can be shown that these quantifiers are non-monotonic in both arguments. Yet these quantifiers satisfy various types of exclusion inferences. For example, the following inference involving *all ... except Smith* and the contradictory concepts of *male* and *female* is valid:

(4) All male members except Smith are noisy Δ
All female members except Smith are noisy

Note that these two sentences cannot be both true because otherwise, according to the meaning of "except", "Smith" would be both male and female.

[9] The truth conditions under GQT of these two exceptive quantifiers are (where s denotes *Smith*):

‖*all except ... Smith*‖(A)(B) is true iff A – B = {s}

‖*no except ... Smith*‖(A)(B) is true iff A ∩ B = {s}

The third type includes the identity comparative quantifiers *the same ... as ...* and *different ... than ...* studied in Beghelli (1994). These two quantifiers both have three arguments[10]. It can be shown that these two quantifiers are both non-monotonic in the 2nd and 3rd arguments. Yet they satisfy various types of exclusion inferences in respect of these two arguments. For example, the following inference involving *the same ... as ...* and the contradictory concepts of *asleep* and *awake*[11] is valid:

(5) (Given that there is some noisy member)
The same members are asleep as noisy Δ
The same members are awake as noisy

2. Projectivity Marking

2.1 Opposition Properties

In the previous section, we have shown that the +-× properties are not appropriate for studying exclusion reasoning. The main problem is that the definitions of these properties are not directly related to the exclusion relations. We thus propose to replace the +-× properties with the following properties studied in Chow (2012, 2017).

Definition 2.1: Let f: B → B' be a function on Boolean algebras, x and y be elements of B.

(i) f is homo-exclusive iff x Δ y entails f(x) Δ f(y).
(ii) f is homo-exhaustive iff x ∇ y entails f(x) ∇ f(y).
(iii) f is anti-exclusive iff x Δ y entails f(x) ∇ f(y).
(iv) f is anti-exhaustive iff x ∇ y entails f(x) Δ f(y).
(v) f is o(pposition)-sensitive iff f possesses at least one of the above properties; f is o-insensitive iff f does not possess any of the above properties.

Note that the definitions of the above properties are very similar to those in Definition 1.5. Thus, following the nomenclature of this definition, the increasing

---

[10] The truth conditions under GQT of these two identity comparative quantifiers are:

$$\|\textit{the same ... as ....}\|(A)(B_1)(B_2) \text{ is true iff } A \cap B_1 = A \cap B_2$$
$$\|\textit{different ... than ....}\|(A)(B_1)(B_2) \text{ is true iff } A \cap B_1 \neq A \cap B_2$$

[11] For the purpose of discussion in this paper, we assume that any person is either asleep or awake but not both, and so treat *asleep* and *awake* as contradictory.

property may also be called “homo-subset” or “homo-superset” property, whereas the decreasing property may also be called “anti-subset” or “anti-superset” property. To distinguish the two sets of properties, the two properties in Definition 1.5 (i)-(ii) will be collectively called “monotonicity properties” (MPs) while the four properties in Definition 2.1 (i)-(iv) will be collectively called “opposition properties” (OPs). A list of the MPs and OPs (in the form of projectivity signatures) of some important functions is given in the Appendix. The results are extracted from Chow (2012, 2017)[12].

Unlike the +-× properties which are strengthening of the MPs, OPs are in a parallel relation with the MPs. There are monotonic but o-insensitive functions (such as the (absolute) numerical quantifier *more than n*), as well as o-sensitive but non-monotonic functions (such as the proportional quantifier *exactly r of* in respect of the 2nd argument where $1/2 < r < 1$ or $0 < r < 1/2$). Moreover, there can be various possibilities of combinations among these properties. But not all combinations yield meaningful results. In this subsection, we explore some possible combinations of these properties.

We first discuss combinations among MPs. We have the following proposition.

Proposition 2.2: A function is both increasing and decreasing iff it is constant.

Proof: Let f be a function. On the one hand, if f is constant, then $f(x) \leq f(y)$ for all x and y. The definitions of the increasing and decreasing properties are thus trivially satisfied. On the other hand, if f is both increasing and decreasing, then for all x and y, $x \leq y$ entails both $f(x) \leq f(y)$ and $f(x) \geq f(y)$, i.e. $f(x) \equiv f(y)$. Since for all x in a Boolean algebra, we have $0 \leq x \leq 1$, this entails $f(0) \equiv f(x) \equiv f(1)$, i.e. f is constant. □

Thus, a quantifier that is both increasing and decreasing always gives the same truth value for any input. Since these quantifiers are trivial and uninteresting, they are excluded from consideration in this paper.

We next discuss combinations among OPs, which have been studied in Chow (2017). Here are some of the propositions adapted from Chow (2017)[13].

---

[12] In Chow (2012, 2017), the homo-exclusive, homo-exhaustive, anti-exclusive and anti-exhaustive properties are denoted “CC→CC”, “SC→SC”, “CC→SC” and “SC→CC”, respectively.

[13] Although the results in Chow (2017) are about generalized quantifiers, these results (together with their proofs) can be readily extended to general functions on Boolean algebras.

Proposition 2.3: There is no function that is both homo-exclusive and anti-exclusive. Neither is there function that is both homo-exhaustive and anti-exhaustive.

From this proposition one can easily deduce that there is no function that possesses three or four of the OPs.

Proposition 2.4: There are functions that are both homo-exclusive and homo-exhaustive, as well as functions that are both anti-exclusive and anti-exhaustive. Moreover, these two types of functions must also be increasing and decreasing, respectively. Examples of the first type of functions are the singular proper names and the identity function, denoted ID. An example of the second type of functions is the negative particle *not*.

Proposition 2.5: There are functions that are both homo-exclusive and anti-exhaustive, as well as functions that are both homo-exhaustive and anti-exclusive. Examples of the first type of functions are the quantifiers *all … except* (in respect of both arguments), *no … except* (in respect of both arguments), *the same … as …* (in respect of the 2nd and 3rd arguments). An example of the second type of functions is the quantifier *different … than …* (in respect of the 2nd and 3rd arguments).

We next discuss combinations between MPs and OPs. It has been shown in Chow (2017) that there are functions that are both increasing and homo-exclusive (such as *every* in the 2nd argument), both increasing and homo-exhaustive (such as *some* in both arguments), both decreasing and anti-exclusive (such as *not every* in the 2nd argument), and both decreasing and anti-exhaustive (such as *no* in both arguments).

Moreover, according to Proposition 2.4, there are functions that are increasing, homo-exclusive and homo-exhaustive. Note that these three properties are similar in that they are all “homo” properties (recall that the increasing property can also be called homo-subset or homo-superset property). There are also functions that are decreasing, anti-exclusive and anti-exhaustive. Note that these three properties are all “anti” properties (recall that the decreasing property can also be called anti-subset or anti-superset property).

While Proposition 2.5 shows that the "homo" and "anti" properties can be mixed within OPs, the following proposition shows that the "homo" and "anti" properties cannot be mixed across MPs and OPs.

Proposition 2.6: Let f be a function on Boolean algebras.

(i) f is both increasing and anti-exclusive iff $f(x) \equiv 1$ for all x.

(ii) f is both increasing and anti-exhaustive iff $f(x) \equiv 0$ for all x.

(iii) f is both decreasing and homo-exclusive iff $f(x) \equiv 0$ for all x.

(iv) f is both decreasing and homo-exhaustive iff $f(x) \equiv 1$ for all x.

Proof: We will only prove (i) and (ii). The proofs of (iii) and (iv) are similar.

(i) On the one hand, if $f(x) \equiv 1$ for all x, then $f(x) \leq f(y)$ and $f(x) \nabla f(y)$ for all x and y. The definitions of the increasing and anti-exclusive properties are thus trivially satisfied. On the other hand, suppose f is both increasing and anti-exclusive and let x be any element. Then since $0 \leq x$ and $0 \Delta x$, by the increasing and anti-exclusive properties of f, we have $f(0) \leq f(x)$ and $f(0) \nabla f(x)$ (i.e. $\neg f(0) \leq f(x)$). Thus, we have $f(0) \vee \neg f(0) \leq f(x)$, which is equivalent to $1 \leq f(x)$. This means that $f(x) \equiv 1$ for all x.

(ii) On the one hand, if $f(x) \equiv 0$ for all x, then $f(x) \leq f(y)$ and $f(x) \Delta f(y)$ for all x and y. The definitions of the increasing and anti-exhaustive properties are thus trivially satisfied. On the other hand, suppose f is both increasing and anti-exhaustive and let x be any element. Then since $x \leq 1$ and $x \nabla 1$, by the increasing and anti-exhaustive properties of f, we have $f(x) \leq f(1)$ and $f(x) \Delta f(1)$ (i.e. $f(x) \leq \neg f(1)$). Thus, we have $f(x) \leq f(1) \wedge \neg f(1)$, which is equivalent to $f(x) \leq 0$. This means that $f(x) \equiv 0$ for all x. □

Thus, the four types of function with mixed properties discussed in Proposition 2.6 are trivial constant functions. For this reason, they are excluded from consideration in this paper.

Before closing this subsection, we will briefly discuss the relationship between the OPs and the +-× properties. Similar to their relation with the MPs, the +-× properties can also be seen as strengthening of the OPs, as shown in the following proposition.

Proposition 2.7: Let f be a function on Boolean algebras.

(i) If f is completely additive, then it is homo-exhaustive.
(ii) If f is completely multiplicative, then it is homo-exclusive.
(iii) If f is completely anti-additive, then it is anti-exhaustive.
(iv) If f is completely anti-multiplicative, then it is anti-exclusive

Proof: We will only prove (i) and (iii). The proofs of (ii) and (iv) are similar.

(i) Let x and y be any two elements such that x ∇ y, i.e. $x \vee y \equiv 1$. Then, since f is completely additive, we have $f(x) \vee f(y) \equiv f(x \vee y) \equiv f(1) \equiv 1$, i.e. f(x) ∇ f(y). This shows that f is homo-exhaustive.

(iii) Let x and y be any two elements such that x ∇ y, i.e. $x \vee y \equiv 1$. Then, since f is completely anti-additive, we have $f(x) \wedge f(y) \equiv f(x \vee y) \equiv f(1) \equiv 0$, i.e. f(x) Δ f(y). This shows that f is anti-exhaustive. □

Propositions 1.7, 1.8 and 2.7 show that each +-× property implies an MP and an OP. For example, if a function is completely multiplicative, then it is increasing and homo-exclusive. On the other hand, there is function that possesses an MP and an OP but not any +-× property. For example, the quantifier *more than 1/2 of* is increasing and homo-exclusive in the 2nd argument, but does not possess any +-× property. We have thus shown that the +-× properties (complete version) are strengthening of the MPs and OPs. Our conclusion is that the +-× properties are not the proper notions for studying exclusion reasoning. They are so strong that they fail to include many quantifiers that satisfy important types of exclusion inferences.

2.2 Projectivity Signatures

The foregoing discussion suggests that we need 15 projectivity signatures to denote (non-trivial) functions according to the MPs and OPs they possess: two for the MPs, four for the OPs, eight for possible combinations of MPs and OPs, and one for all functions. These projectivity signatures comprise a set which is denoted by Σ. We will use a special symbolism for the members of this set, which is set out in the following table. The idea of the symbols for the four OPs (i.e. the form $(\rho_1 \rightarrow \rho_2)$) is borrowed from Chow (2012, 2017).

Definition 2.8

Signature	**Property**

↑	increasing
↓	decreasing
(Δ→Δ)	homo-exclusive
(∇→∇)	homo-exhaustive
(Δ→∇)	anti-exclusive
(∇→Δ)	anti-exhaustive
(↑ ∧ Δ→Δ)	increasing and homo-exclusive
(↑ ∧ ∇→∇)	increasing and homo-exhaustive
(↓ ∧ Δ→∇)	decreasing and anti-exclusive
(↓ ∧ ∇→Δ)	decreasing and anti-exhaustive
(Δ→Δ ∧ ∇→Δ)	homo-exclusive and anti-exhaustive
(∇→∇ ∧ Δ→∇)	homo-exhaustive and anti-exclusive
(↑ ∧ Δ→Δ ∧ ∇→∇)	increasing, homo-exclusive and homo-exhaustive
(↓ ∧ Δ→∇ ∧ ∇→Δ)	decreasing, anti-exclusive and anti-exhaustive
●	general (applicable to any function)

While some symbols above look cumbersome, they are in fact transparent and self-explanatory. For example, the symbol ∇→Δ represents the property that yields exclusive (Δ) function values from exhaustive (∇) inputs, i.e. the anti-exhaustive property. Theoretically speaking, the signatures for the MPs can also be written in the form ($\rho_1 \rightarrow \rho_2$), where ρ_1, $\rho_2 \in \{\leq, \geq\}$, i.e. (≤→≤) (or (≥→≥)) for the increasing property, and (≤→≥) (or (≥→≤)) for the decreasing property. But since the symbols are not unique (each property has two possible signatures), we decide to use ↑ and ↓ as the signatures of the MPs, as these two symbols are also commonly used in GQT.

The first six signatures shown above are collectively called "basic signatures" and are the building blocks of other signatures, which are collectively called "complex signatures" (except ●). Like R, there is also a natural ordering among the members of Σ, which is depicted by Fig. 2.

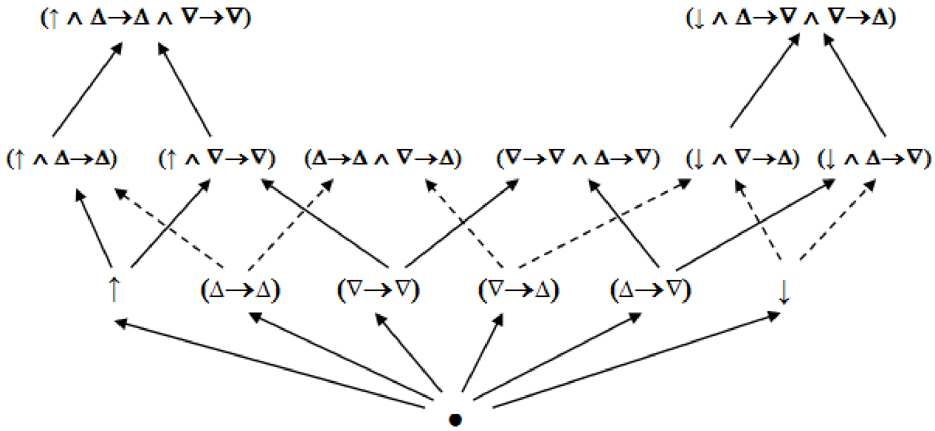

Fig. 2 Partial order of members of Σ

In Fig. 2, an arrow pointing from a signature ϕ_1 to another signature ϕ_2 means that ϕ_2 represents a stronger property than ϕ_1, i.e. for any function f, if f possesses the property represented by ϕ_2, then f also possesses the property represented by ϕ_1. Fig. 2 can also be viewed as representing a partial order comprising the members of Σ. One can now see that the symbol ∧ within some of the signatures in fact represents the meet between some of the members of Σ.

The form ($\rho_1 \rightarrow \rho_2$) opens up the possibility of defining properties other than those studied in this paper. To keep the discussion of this paper manageable, we will not explore all possibilities, but will state the following proposition which will be useful below.

Proposition 2.9: Let f be a function on Boolean algebras.

(i) f has signature (≤→Δ) or (≥→Δ) iff f(x) ≡ 0 for all x.

(ii) f has signature (≤→∇) or (≥→∇) iff f(x) ≡ 1 for all x.

(iii) f has signature (Δ→≤), (Δ→≥), (∇→≤) or (∇→≥) iff f is constant.

Proof: We will only prove the first part of (i) and the last part of (iii). The proofs of the remaining parts are similar.

(i) On the one hand, if f(x) ≡ 0 for all x, then f(x) ≤ ¬f(y) for all x and y. The definition of the property represented by (≤→Δ) is thus trivially satisfied. On the other hand, if f has signature (≤→Δ), then since x ≤ x, we have f(x) Δ f(x) for all x. This is true only if f(x) ≡ 0 for all x.

(iii) On the one hand, if f is constant, then f(x) ≥ f(y) for all x and y. The definition of the property represented by $(\nabla \rightarrow \geq)$ is thus trivially satisfied. On the other hand, if f has signature $(\nabla \rightarrow \geq)$, then since x ∇ 1 and 1 ∇ x for all x, we have f(x) ≥ f(1) and f(1) ≥ f(x), which is equivalent to f(x) ≡ f(1) for all x. Thus, f is constant. □

2.3 Projection

An advantage of the signatures introduced above is that they greatly facilitate the computations of projection and composition, which are two of the three necessary computations in the Calculus to be introduced below (the third one is the JOIN operation introduced in Subsection 1.2). We discuss projection in this subsection. Here is the definition of projection (adapted from Icard (2012)).

Definition 2.10: Let $\rho \in R$ and $\phi \in \Sigma$. The projection of ρ under ϕ, denoted $\phi[\rho]$, is the strongest member of R such that the following holds: whenever x ρ y and f is a function with signature ϕ, then we have f(x) $\phi[\rho]$ f(y).

Writing some of the projectivity signatures in the form $(\rho_1 \rightarrow \rho_2)$, where ρ_1 and ρ_2 is either Δ or ∇, makes it easy to compute the projection of any $\rho \in R$ under any of these signatures. For example, we have $(\nabla \rightarrow \Delta)[\nabla] = \Delta$ because whenever x ∇ y and f is an anti-exhaustive function (i.e. with signature $(\nabla \rightarrow \Delta)$), then f(x) Δ f(y). This is precisely the definition of the anti-exhaustive property. Moreover, Δ is the strongest relation such that the aforesaid statement holds. To see this, in R there is only one member stronger than Δ, namely ⊥, which is equal to $\Delta \wedge \nabla$. If $(\nabla \rightarrow \Delta)[\nabla]$ were to be equal to ⊥, it would be the case that whenever x ∇ y and f is an anti-exhaustive function, then f(x) ⊥ f(y), i.e. f(x) Δ f(y) and f(x) ∇ f(y). But this would mean that f is both anti-exhaustive and homo-exhaustive, which is impossible according to Proposition 2.3.

Moreover, we also have $(\nabla \rightarrow \Delta)[\Delta] = \#$. To prove this, we have to show that $(\nabla \rightarrow \Delta)[\Delta]$ is not equal to ≤, ≥, Δ and ∇ (and so also not equal to ≡ and ⊥, leaving # as the only possible result). We will prove $(\nabla \rightarrow \Delta)[\Delta]$ is not equal to ∇ and ≤. The remaining parts of the proof are similar. Let f be a function that possesses the anti-exhaustive property (i.e. with signature $(\nabla \rightarrow \Delta)$). If $(\nabla \rightarrow \Delta)[\Delta]$ were to be equal to ∇, then according to Definition 2.10, whenever x Δ y, we would have f(x) ∇ f(y). But this is precisely the definition of the anti-exclusive property (i.e. with signature $(\Delta \rightarrow \nabla)$). This means that every anti-exhaustive function is anti-exclusive. But this is incorrect because there is anti-exhaustive function that is not anti-exclusive, such as the quantifier *no*. Thus, $(\nabla\to\Delta)[\Delta] \neq \nabla$. Similarly, if $(\nabla\to\Delta)[\Delta]$ were to be equal to $\leq$, then whenever x Δ y, we would have $f(x) \leq f(y)$. But this is precisely the definition of the property with signature $(\Delta\to\leq)$. This means that every anti-exhaustive function is $(\Delta\to\leq)$. But this is incorrect because according to Proposition 2.9, functions with signature $(\Delta\to\leq)$ are constant, and there is certainly non-constant anti-exhaustive function. Thus, $(\nabla\to\Delta)[\Delta] \neq \leq$.

Using the same line of reasoning, we can compute other results of $(\rho_1\to\rho_2)[\rho_3]$, which are summarized as the following proposition.

Proposition 2.11: Let ρ_i (i = 1, 2, 3) be either Δ or ∇. Then $(\rho_1\to\rho_2)[\rho_1] = \rho_2$, and $(\rho_1\to\rho_2)[\rho_3] = \#$ if $\rho_1 \neq \rho_3$.

For the signatures $\uparrow$ and $\downarrow$, we have the following results from GQT: $\uparrow[\leq] = \leq$, $\uparrow[\geq] = \geq$, $\downarrow[\leq] = \geq$ and $\downarrow[\geq] = \leq$. One can also show that $\uparrow[\rho] = \downarrow[\rho] = \#$ if ρ is either Δ or ∇ by following the same line of reasoning as in the preceding paragraphs.

To compute $\phi[\equiv]$ or $\phi[\perp]$, we first note that $\equiv$ and $\perp$ can be written as $\rho_1 \wedge \rho_2$. We can then compute $\phi[\rho_1]$ and $\phi[\rho_2]$ separately and then combine the results. For example, to compute $\downarrow[\equiv]$, we reason as follows. Suppose $x \equiv y$. Then we have both $x \leq y$ and $x \geq y$. Let f be a function with signature $\downarrow$. Then by Definition 2.10, we have both $f(x) \downarrow[\leq] f(y)$ and $f(x) \downarrow[\geq] f(y)$, i.e. $f(x) \geq f(y)$ and $f(x) \leq f(y)$. But by Proposition 1.3, this is equivalent to $f(x) (\geq \wedge \leq) f(y)$, i.e. $f(x) \equiv f(y)$. Moreover, it is clear that $\equiv$ is the strongest relation such that the aforesaid statement holds. Thus, $\downarrow[\equiv] = \equiv$.

To compute $(\phi_1 \wedge \phi_2)[\rho]$ where ϕ_1 and ϕ_2 are basic signatures, we can compute $\phi_1[\rho]$ and $\phi_2[\rho]$ separately and then combine the results. For example, to compute $(\Delta\to\Delta \wedge \nabla\to\Delta)[\Delta]$, we reason as follows. Suppose x Δ y. Let f be a function with signature $(\Delta\to\Delta \wedge \nabla\to\Delta)$. Then f has the properties represented by $(\Delta\to\Delta)$ and $(\nabla\to\Delta)$. From the former and Definition 2.10, we have $f(x)\ (\Delta\to\Delta)[\Delta]\ f(y)$, i.e. $f(x)\ \Delta\ f(y)$. From the latter and Definition 2.10, we have $f(x)\ (\nabla\to\Delta)[\Delta]\ f(y)$, i.e. $f(x)\ \#\ f(y)$. By Proposition 1.3, we have $f(x)\ (\Delta \wedge \#)\ f(y)$, i.e. $f(x)\ \Delta\ f(y)$. Moreover, it is clear that Δ is the strongest relation such that the aforesaid statement holds. Thus, $(\Delta\to\Delta \wedge \nabla\to\Delta)[\Delta] = \Delta$.

In general, we have $\phi[\rho_1 \wedge \ldots \wedge \rho_n] = \phi[\rho_1] \wedge \ldots \wedge \phi[\rho_n]$ and $(\phi_1 \wedge \ldots \wedge \phi_n)[\rho] = \phi_1[\rho] \wedge \ldots \wedge \phi_n[\rho]$, where ρ, $\rho_1 \ldots \rho_n$ are basic relations and ϕ, $\phi_1 \ldots \phi_n$ are basic signatures.

Finally, it is easy to see that $\phi[\#] = \#$ for any $\phi \in \Sigma$ and $\bullet[\rho] = \#$ for any $\rho \in R$. Moreover, one can show (by exhaustive checking say) that the results of all possible projections are members of R and the results can be found by using the computation method introduced above.

2.4 Composition

We turn to composition in this subsection. Here is the definition (adapted from Icard (2012)).

Definition 2.12: Let $\phi_1, \phi_2 \in \Sigma$. The composition of ϕ_2 and ϕ_1, denoted $\phi_2 \circ \phi_1$, is the strongest member of Σ such that the following holds: if f_2 is a function with signature ϕ_2 and f_1 is a function with signature ϕ_1, then $f_2 \circ f_1$ is a function with signature $\phi_2 \circ \phi_1$.

Writing some of the projectivity signatures in the form $(\rho_1 \rightarrow \rho_2)$, where ρ_1 and ρ_2 is either Δ or ∇, makes it easy to compute the composition of these signatures. For example, we have $(\Delta\rightarrow\nabla) \circ (\nabla\rightarrow\Delta) = (\nabla\rightarrow\nabla)$. To prove this, we first note that if x ∇ y and f_1 is a function with signature $(\nabla\rightarrow\Delta)$, then we have $f_1(x)\ \Delta\ f_1(y)$. Viewing $f_1(x)$ and $f_1(y)$ as inputs of the function f_2 with signature $(\Delta\rightarrow\nabla)$, we have $f_2 \circ f_1(x)\ \nabla\ f_2 \circ f_1(y)$. We next have to show that $(\Delta\rightarrow\nabla) \circ (\nabla\rightarrow\Delta)$ is not equal to $(\nabla\rightarrow\nabla \wedge \Delta\rightarrow\nabla)$ and $(\uparrow \wedge \Delta\rightarrow\Delta \wedge \nabla\rightarrow\nabla)$, which are the only two members in Σ stronger than $(\nabla\rightarrow\nabla)$. Now suppose x Δ y and f_1 is a function with signature $(\nabla\rightarrow\Delta)$, then according to the results of the previous subsection, we have $f_1(x)\ \#\ f_1(y)$. Next viewing $f_1(x)$ and $f_1(y)$ as inputs of the function f_2 with signature $(\Delta\rightarrow\nabla)$, we have $f_2 \circ f_1(x)\ \#\ f_2 \circ f_1(y)$. This shows that $(\Delta\rightarrow\nabla) \circ (\nabla\rightarrow\Delta)$ cannot be equal to $(\Delta\rightarrow\nabla)$. Neither can it be equal to $(\nabla\rightarrow\nabla \wedge \Delta\rightarrow\nabla)$. The proof that $(\Delta\rightarrow\nabla) \circ (\nabla\rightarrow\Delta) \neq (\uparrow \wedge \Delta\rightarrow\Delta \wedge \nabla\rightarrow\nabla)$ is similar.

Moreover, we also have $(\Delta\rightarrow\nabla) \circ (\Delta\rightarrow\nabla) = \bullet$. To prove this, we have to show that $(\Delta\rightarrow\nabla) \circ (\Delta\rightarrow\nabla)$ is not equal to $\uparrow$, $\downarrow$, $(\Delta\rightarrow\Delta)$, $(\nabla\rightarrow\nabla)$, $(\Delta\rightarrow\nabla)$ and $(\nabla\rightarrow\Delta)$ (and so also not equal to the complex signatures, leaving $\bullet$ as the only possible result). We will prove $(\Delta\rightarrow\nabla) \circ (\Delta\rightarrow\nabla)$ is not equal to $\uparrow$. The remaining parts of the proof are similar. Suppose $x \leq y$ or $x \geq y$ and f_1 is a function with signature $(\Delta\rightarrow\nabla)$, then according to the results of the previous subsection, in either case we have $f_1(x)\ \#\ f_1(y)$. Next viewing $f_1(x)$ and $f_1(y)$ as inputs of the function f_2 with signature $(\Delta\rightarrow\nabla)$, we have $f_2 \circ f_1(x)\ \#\ f_2 \circ f_1(y)$. This shows that $(\Delta\rightarrow\nabla) \circ (\Delta\rightarrow\nabla)$ cannot be equal to $\uparrow$.

Using the same line of reasoning, we can compute other results of $(\rho_3 \rightarrow \rho_4) \circ (\rho_1 \rightarrow \rho_2)$, which are summarized as the following proposition.

Proposition 2.13: Let ρ_i (i = 1, 2, 3, 4) be either Δ or ∇. Then $(\rho_2 \rightarrow \rho_3) \circ (\rho_1 \rightarrow \rho_2) = (\rho_1 \rightarrow \rho_3)$, and $(\rho_3 \rightarrow \rho_4) \circ (\rho_1 \rightarrow \rho_2) = \bullet$ if $\rho_2 \neq \rho_3$.

For the signatures ↑ and ↓, we have the following results from GQT: $\uparrow \circ \uparrow = \uparrow$, $\uparrow \circ \downarrow = \downarrow$, $\downarrow \circ \uparrow = \downarrow$ and $\downarrow \circ \downarrow = \uparrow$. One can also show that $\uparrow \circ (\rho_1 \rightarrow \rho_2) = \downarrow \circ (\rho_1 \rightarrow \rho_2) = (\rho_1 \rightarrow \rho_2) \circ \uparrow = (\rho_1 \rightarrow \rho_2) \circ \downarrow = \bullet$ if ρ_1, ρ_2 is either Δ or ∇ by following the same line of reasoning as above.

To compute $(\phi_2 \wedge \phi_3) \circ \phi_1$ where ϕ_1, ϕ_2, ϕ_3 are basic signatures, we can compute $\phi_2 \circ \phi_1$ and $\phi_3 \circ \phi_1$ separately and then combine the results. For example, to compute $(\nabla \rightarrow \nabla \wedge \Delta \rightarrow \nabla) \circ (\Delta \rightarrow \Delta)$, we reason as follows. Suppose $x \; \Delta \; y$ and let f_1 be a function with signature $(\Delta \rightarrow \Delta)$. Then we have $f_1(x) \; \Delta \; f_1(y)$. Next, view $f_1(x)$ and $f_1(y)$ as inputs of the function f_2 with signature $(\nabla \rightarrow \nabla \wedge \Delta \rightarrow \nabla)$, Then f_2 has the properties represented by $(\nabla \rightarrow \nabla)$ and $(\Delta \rightarrow \nabla)$. From the former, we have $f_2 \circ f_1(x) \; \# \; f_2 \circ f_1(y)$. From the latter, we have $f_2 \circ f_1(x) \; \nabla \; f_2 \circ f_1(y)$. By Proposition 1.3, we have $f_2 \circ f_1(x) \; (\# \wedge \nabla) \; f_2 \circ f_1(y)$, i.e. $f_2 \circ f_1(x) \; \nabla \; f_2 \circ f_1(y)$. Moreover, it is clear that ∇ is the strongest relation between $f_2 \circ f_1(x)$ and $f_2 \circ f_1(y)$ by assuming the Δ relation between x and y, and all other relations assumed between x and y will only yield the # relation between $f_2 \circ f_1(x)$ and $f_2 \circ f_1(y)$. Thus, $(\nabla \rightarrow \nabla \wedge \Delta \rightarrow \nabla) \circ (\Delta \rightarrow \Delta) = (\Delta \rightarrow \nabla)$.

In general, we have $(\phi_1 \wedge \ldots \wedge \phi_n) \circ (\psi_1 \wedge \ldots \wedge \psi_m) = (\phi_1 \circ \psi_1) \wedge \ldots \wedge (\phi_1 \circ \psi_m) \wedge \ldots \wedge (\phi_n \circ \psi_1) \wedge \ldots \wedge (\phi_n \circ \psi_m)$ where $\phi_1 \ldots \phi_n$, $\psi_1 \ldots \psi_m$ are basic signatures.

Finally, it is easy to see that $\bullet \circ \phi = \phi \circ \bullet = \bullet$ for any $\phi \in \Sigma$. Moreover, one can show (by exhaustive checking say) that the results of all possible compositions are members of Σ and the results can be found by using the computation method introduced above.

2.5 Sufficient Conditions for Valid Inferences[14]

In the foregoing, to avoid complicating the discussion, we have put aside the issue that a particular MP/OP of a quantifier may be associated with certain condition. Note that in Definitions 1.5 and 2.1, $x \; \rho_1 \; y$ can be seen as (inherent) conditions of the

[14] The previous version of this paper did not discuss (sufficient) conditions associated with inclusion / exclusion reasoning. I am grateful to an anonymous reviewer for pointing out the need for clarifying these conditions. Without the reviewer's advice, this paper would not have included this subsection.

inferences f(x) ρ_2 f(y) (where ρ_1, ρ_2 are one of $\leq$, $\geq$, Δ and ∇). In the simplest case, this condition is a sufficient condition, i.e. whenever the condition holds, the inference pattern in question is valid. In case the condition does not hold, there is no guarantee for the validity of the inference pattern, i.e. there exist models in which the condition does not hold and the inference pattern is invalid. For example, B ≤ B' is a sufficient condition for the inference pattern ‖*every*‖(A)(B) ≤ ‖*every*‖(A)(B'). In case B is not a subset of B', there is no guarantee for the validity of the aforesaid inference pattern, i.e. there exist models in which B is not a subset of B' and it is not the case that ‖*every*‖(A)(B) ≤ ‖*every*‖(A)(B').

But for certain quantifiers, the inherent conditions are not sufficient to guarantee the validity of some inferences involving MPs/OPs, and we need some additional conditions. What we discuss in this subsection are these additional conditions[15]. Chow (2012, 2017) has identified the additional conditions associated with the OPs of a number of quantifiers. The additional condition and the inherent condition together serve as sufficient conditions for the inference pattern in question, i.e. whenever the additional condition and the inherent condition hold, the inference pattern is valid. In case either of these conditions does not hold, there is no guarantee for the validity of the inference pattern, i.e. there exist models in which either of these conditions does not hold and the inference pattern is invalid.

For illustration, *all … except Smith* is $\nabla\rightarrow\Delta$ in the 1st argument if $B \cup \{s\} \neq U^*$ (where B, s and U^* denote the 2nd argument of the quantifier, the individual "Smith" and the universe, respectively). In case $B \cup \{s\} = U^*$, there is no guarantee for the validity of the aforesaid inference pattern, i.e. there exist models in which A ∇ A', $B \cup \{s\} = U^*$ and it is not the case that ‖*all … except Smith*‖(A)(B) Δ ‖*all … except Smith*‖(A')(B). For example, in a universe composed of "members" in which "Smith" is a male member and all members except "Smith" are asleep, i.e. ‖*asleep*‖ $\cup \{s\} = U^*$, even though *member* ∇ *male member*, the following inference is invalid:

(6) All members except Smith are asleep Δ
All male members except Smith are asleep

Instead of providing the proofs of the sufficiency of all additional conditions associated with the quantifiers discussed in this paper, which will be lengthy and blur

---

[15] These conditions are not parts of the projectivity signatures but are additional information specified in the lexicon (the Appendix to this paper can be seen as a part of this lexicon). Therefore, the discussion in this subsection should be seen as a justification of the conditions specified in the Appendix rather than part of the calculus to be introduced in the next section.

the focus of this paper, here we will only provide the sufficiency proofs of two particular cases as an illustration. The proofs of the other cases follow the same line of reasoning. Note that three types of quantifiers given at the Appendix are associated with additional conditions. They are the classical quantifiers, exceptive quantifiers and identity comparative quantifiers. The additional conditions associated with the classical quantifiers are easy to derive and prove, and so we will only discuss the other two types of quantifiers.

First, we prove that the exceptive quantifier *all … except Smith* is $\nabla \rightarrow \Delta$ in the 1st argument if $B \cup \{s\} \neq U^*$. Suppose that the condition holds and $A \nabla A'$. By way of contradiction, assume that ‖*all … except Smith*‖(A)(B) and ‖*all … except Smith*‖(A')(B) are both true. Then $A - B = \{s\}$ and $A' - B = \{s\}$, i.e. $A - \{s\} \subseteq B$ and $A' - \{s\} \subseteq B$. We thus have $(A - \{s\}) \cup (A' - \{s\}) \subseteq B$, i.e. $(A \cup A') - \{s\} \subseteq B$. Since $A \nabla A'$, this implies $U^* - \{s\} \subseteq B$. But this contradicts the condition that $B \cup \{s\} \neq U^*$.

We next prove that the identity comparative quantifier *the same … as …* is $\Delta \rightarrow \Delta$ in the second argument if $A \cap B_2 \neq \varnothing$ where A, B_1 and B_2 represent the 1st, 2nd and 3rd arguments of *the same … as …*, respectively. Suppose that the condition holds and $B_1 \Delta B_1'$. By way of contradiction, assume that ‖*the same … as …*‖(A)(B_1)(B_2) and ‖*the same … as …*‖(A)(B_1')(B_2) are both true. Then $A \cap B_1 = A \cap B_2$ and $A \cap B_1' = A \cap B_2$. This implies $A \cap B_1 = A \cap B_1'$. Since $B_1 \Delta B_1'$, we have $(A \cap B_1) \cap (A \cap B_1') = \varnothing$. But since $A \cap B_1 = A \cap B_1'$, we must have $A \cap B_1 = A \cap B_1' = \varnothing$. This implies $A \cap B_2 = \varnothing$. But this contradicts the condition that $A \cap B_2 \neq \varnothing$.

In the above, we have discussed additional conditions associated with inclusion/exclusion inferences between sentences. For example, the quantifier *no* is $\nabla \rightarrow \Delta$ in the 1st argument given the condition that its second argument does not denote the empty set. One should then be able to derive the following:

(7) (Given that some member died)
No male members died Δ No female members died

In the above, we are using ‖*male members*‖ ∇ ‖*female members*‖ in a (sub-)universe composed of "members" and ‖*died*‖ ≠ ∅.

But inclusion/exclusion inferences also exist between predicates. In the above example, if we replace "died" by "admits" and suitably change the word order, then "admits no male members" and "admits no female members" are predicates (instead of sentences) which can be represented in set form by $\{x_1\text{: } no(\|\textit{male members}\|)(\{x_2\text{: } (x_1, x_2) \in \|\textit{admit}\|\})\}$ and $\{x_1\text{: } no(\|\textit{female members}\|)(\{x_2\text{: } (x_1, x_2) \in \|\textit{admit}\|\})\}$, respectively. Note that in this paper we follow the practice of some GQT scholars, such as Keenan and Westerståhl (2011), in treating generalized quantifiers corresponding to full noun phrases as "arity reducers", i.e. functions that, when applied to an n-ary predicate, will yield an (n – 1)-predicate. Thus, when $no(\|\textit{male members}\|)$ is applied to the unary predicate $\|\textit{died}\|$, it yields a proposition (a proposition can be seen as a 0-ary predicate). When applied to the binary predicate $\|\textit{admit}\|$, it yields the unary predicate $\{x_1\text{: } no(\|\textit{male members}\|)(\{x_2\text{: } (x_1, x_2) \in \|\textit{admit}\|\})\}$[16].

By analogy with the above example, the two predicates $\{x_1\text{: } no(\|\textit{male members}\|)(\{x_2\text{: } (x_1, x_2) \in \|\textit{admit}\|\})\}$ and $\{x_1\text{: } no(\|\textit{female members}\|)(\{x_2\text{: } (x_1, x_2) \in \|\textit{admit}\|\})\}$ should also satisfy the Δ relation given certain condition. Since $\{x_2\text{: } (x_1, x_2) \in \|\textit{admit}\|\}$ plays a similar role to $\|\textit{died}\|$ in the above example, this condition should be something like $\{x_2\text{: } (x_1, x_2) \in \|\textit{admit}\|\} \neq \varnothing$. But here x_1 is an unbound variable which needs to be bound by a quantifier. What quantifier is this? It turns out that $\forall x_1.\{x_2\text{: } (x_1, x_2) \in \|\textit{admit}\|\} \neq \varnothing$ would be a suitable condition. This is because under this condition, for every particular x_1, we have $\{x_2\text{: } (x_1, x_2) \in \|\textit{admit}\|\} \neq \varnothing$. Then, we must have $no(\|\textit{male members}\|)(\{x_2\text{: } (x_1, x_2) \in \|\textit{admit}\|\})\ \Delta\ no(\|\textit{female members}\|)(\{x_2\text{: } (x_1, x_2) \in \|\textit{admit}\|\})$ (note that when x_1 is a particular constant, $\{x_2\text{: } (x_1, x_2) \in \|\textit{admit}\|\}$ is just a unary predicate like $\|\textit{died}\|$ in the above example). This means that for every x_1, x_1 cannot be both a member of $\{x_1\text{: } no(\|\textit{male members}\|)(\{x_2\text{: } (x_1, x_2) \in \|\textit{admit}\|\})\}$ and a member of $\{x_1\text{: } no(\|\textit{female members}\|)(\{x_2\text{: } (x_1, x_2) \in \|\textit{admit}\|\})\}$. Thus, the two predicates are represented by disjoint sets and so satisfy the Δ relation. Summarizing the above discussion, we have the following inference (for convenience, in what follows we assume a universe with "clubs" and "members" as sub-universes):

(8) (Given that every club admits some member)
admits no male members Δ admits no female members

The above example shows a Δ relation between two predicates. We next discuss a ∇ relation between two predicates. Consider the predicates "admits some male member" and "admits some female member" which can be represented by $\{x_1\text{: } some(\|\textit{male member}\|)(\{x_2\text{: } (x_1, x_2) \in \|\textit{admit}\|\})\}$ and $\{x_1\text{: } some(\|\textit{female member}\|)(\{x_2\text{: } (x_1, x_2) \in \|\textit{admit}\|\})\}$, respectively. Similar to the above example, it can be shown that

[16] Here we have adapted definitions (86) and (88) in Keenan and Westerståhl (2011): let Q be an arity reducer and B an n-ary predicate, then $Q(B) = \{(x_1, \ldots x_{n-1})\text{: } Q(\{x_n\text{: } (x_1, \ldots x_n) \in B\})\}$.

these two predicates satisfy the ∇ relation given the condition $\forall x_1.\{x_2: (x_1, x_2) \in \|admit\|\} \neq \varnothing$. This is because under this condition, for every particular x_1, we have $\{x_2: (x_1, x_2) \in \|admit\|\} \neq \varnothing$. Then, we must have *some*($\|$*male member*$\|$)($\{x_2: (x_1, x_2) \in \|admit\|\}$) ∇ *some*($\|$*female member*$\|$)($\{x_2: (x_1, x_2) \in \|admit\|\}$), because *some* is ∇→∇ in the 1st argument given the condition that its second argument does not denote the empty set. This means that for every x_1, x_1 must be either a member of $\{x_1$: *some*($\|$*male member*$\|$)($\{x_2: (x_1, x_2) \in \|admit\|\}$)$\}$ or a member of $\{x_1$: *some*($\|$*female member*$\|$)($\{x_2: (x_1, x_2) \in \|admit\|\}$)$\}$. Thus, the two predicates are represented by sets whose union covers the sub-universe of "clubs" and so satisfy the ∇ relation. Summarizing the above discussion, we have the following inference:

(9) (Given that every club admits some member)
admits some male member ∇ admits some female member

The above discussion can be generalized to the case when the arity reducer is applied to an n-ary predicate, where $n \geq 1$. For example, let B be an n-ary predicate. Then, *no*($\|$*male members*$\|$) applied to B yields the $(n-1)$-ary predicate $\{(x_1, \ldots x_{n-1})$: *no*($\|$*male members*$\|$)($\{x_n: (x_1, \ldots x_n) \in B\}$)$\}$, and by generalizing the above cases, it can easily be seen that we must have $\{(x_1, \ldots x_{n-1})$: *no*($\|$*male members*$\|$)($\{x_n: (x_1, \ldots x_n) \in B\}$)$\}$ Δ $\{(x_1, \ldots x_{n-1})$: *no*($\|$*female members*$\|$)($\{x_n: (x_1, \ldots x_n) \in B\}$)$\}$ given the condition $\forall x_1 \ldots x_{n-1}.\{x_n: (x_1, \ldots x_n) \in B\} \neq \varnothing$. The conditions associated with other quantifiers can be generalized in a similar fashion and these are summarized in the Appendix.

Before leaving this subsection, we have to point out that the conditions discussed in this paper are different from the presuppositions associated with some quantifiers. Presuppositions are assumptions taken for granted by the speaker. A sentence with presupposition failure will have no truth value instead of being false. For example, since the proper name "Smith" presupposes that "Smith exists", $\{s\} \neq \varnothing$ should be treated as a presupposition associated with the Montagovian individual[17] *Smith* as well as the exceptive quantifiers *all ... except Smith* and *no ... except Smith*. Similarly, since the truth condition of the proportional quantifier *at least r of*, i.e. $\|$*at least r of*$\|$(A)(B) is true iff $|A \cap B| / |A| \geq r$, involves division by $|A|$, this quantifier has no truth value if $A = \varnothing$, and so $A \neq \varnothing$ should be treated as a presupposition associated with this quantifier. Since presuppositions are not related to the inference patterns of quantifiers, they will not be further discussed in this paper and will not be included in

[17] In GQT, a proper name can be seen as a generalized quantifier. Such quantifiers are called "Montagovian individuals" in Peters & Westerståhl (2006).

the Appendix.

3. A Revised Projectivity Calculus

3.1 Basic Framework

Having defined the projectivity signatures and the associated operations, we now propose a Revised Projectivity Calculus that is denoted “RC”. Since RC will inherit the key features of C with necessary amendments to cater for the new properties and signatures introduced in this paper, we will not repeat all the details contained in Icard (2012), but will focus on those features that are different from C as well as those features that are necessary for the propositions and worked examples to be discussed below.

The set of types T under RC is generated from a basic type *n* representing nouns and a type v_i representing verbs with *i* arguments. We also adopt the convention of using v_0 to represent the type of truth values. Moreover, to cater for words with polymorphic types such as *not*, we will use the symbol τ to represent the general type.

We assume that T contains all relevant functional types whose symbols are marked with projectivity signatures, such as $n \xrightarrow{(\Delta\to\Delta \wedge \nabla\to\Delta)} (v_i \xrightarrow{(\Delta\to\Delta \wedge \nabla\to\Delta)} v_{i-1})$, which is the type of the quantifier *all … except Smith*. This symbol contains the information that *all … except Smith* has signature (Δ→Δ ∧ ∇→Δ) both in its 1st argument, which has type *n*, and its 2nd argument, which has type v_i, and its output has type v_{i-1}. This shows that *all … except Smith*, after being applied to an argument of type *n* such as *boys*, is an arity reducer because *all … except Smith*(*boys*) turns an argument of type v_i to an output of type v_{i-1}.

The language L of RC comprises basic terms such as constants and variables as well as terms that are formed by function application. In what follows, a term *t* with its type τ will be denoted $t : \tau$. We also assume that L has the usual semantics with the usual definitions of notions such as models, domains, interpretation function, etc. Specifically, each type τ corresponds to a domain that is denoted $D(\tau)$, while the interpretation function is denoted as $\| \ \|$ such that if *t* is a constant term, then $\|t\| \in D(\tau)$; and if *t* has the form $s(u)$, then $\|s(u)\| = \|s\|(\|u\|)$.

3.2 Ground Terms and Contexts

RC inherits from C the notions of ground terms and contexts. The former are terms that contain no variables, while the latter are terms that contain exactly one variable. A context can be seen as a function. In fact, each context can be associated with a function as defined below (adapted from Icard (2012)).

Definition 3.1: Let $t : \tau_2$ be a context with a variable $x : \tau_1$. Then we define a function F[t] from D(τ_1) to D(τ_2) inductively as follows:

(i) If $t = x$, then F[x] = ID, the identity function;

(ii) If $t = s(u)$ and x is a sub-term of s, then $F[s(u)] = \lambda z.F[s](z)(\|u\|)$ for $z \in D(\tau_1)$;

(iii) If $t = s(u)$ and x is a sub-term of u, then $F[s(u)] = \|s\| \circ F[u]$.

For illustration, suppose we replace "male" in "All male members except Smith are noisy" by a variable x. We then obtain the following context[18]:

(10) *all … except Smith*(x(*members*))(*are noisy*)

We next compute as follows:

(11) F[*all … except Smith*(x(*members*))(*are noisy*)]

= λz.F[*all … except Smith*(x(*members*))](z)(‖*are noisy*‖)

= λz.‖*all … except Smith*‖ ∘ F[x(*members*)](z)(‖*are noisy*‖)

= λz.‖*all … except Smith*‖ ∘ λw.F[x](w)(‖*members*‖) (z)(‖*are noisy*‖)

= λz.‖*all … except Smith*‖ ∘ $\lambda w.w$(‖*members*‖) (z)(‖*are noisy*‖)

= λz.‖*all … except Smith*‖ ∘ z(‖*members*‖)(‖*are noisy*‖)

= $\lambda z.z$(‖*members*‖) – ‖*are noisy*‖ = {s}

The above result gives a correct representation of "All x members except Smith are noisy".

The purpose of defining F(t) is not so much to provide a representation of contexts as to help prove the soundness of projectivity marking. To this end, we first need the following definitions (adapted from Icard (2012)).

Definition 3.2: Let t be a term. The topmost projectivity of t, denoted Top[t], is defined as follows.

---

[18] To simplify matters, we treat "are noisy" as a unit.

(i) If t is of basic type, then Top[t] = ($\uparrow \wedge \Delta \rightarrow \Delta \wedge \nabla \rightarrow \nabla$).
(ii) If t is of functional type $\tau_1 \xrightarrow{\phi} \tau_2$ where $\phi \in \Sigma$, then Top[t] = ϕ.

Definition 3.3: Let t be a context with a variable x. The projectivity of t, denoted Pro[t], is defined inductively as follows:

(i) If $t = x$, then Pro[x] = ($\uparrow \wedge \Delta \rightarrow \Delta \wedge \nabla \rightarrow \nabla$);
(ii) If $t = s(u)$ and x is a sub-term of s, then Pro[$s(u)$] = Pro[s];
(iii) If $t = s(u)$ and x is a sub-term of u, then Pro[$s(u)$] = Top[s] $\circ$ Pro[u].

For illustration, the projectivity of the context in (10) can be computed according to the above definitions as follows:

(12) Pro[*all … except Smith*(x(*members*))(*are noisy*)]
= Pro[*all … except Smith*(x(*members*))]
= Top[*all … except Smith*] $\circ$ Pro[x(*members*)]
= Top[*all … except Smith*] $\circ$ Pro[x]
= ($\Delta \rightarrow \Delta \wedge \nabla \rightarrow \Delta$) $\circ$ ($\uparrow \wedge \Delta \rightarrow \Delta \wedge \nabla \rightarrow \nabla$)
= ($\Delta \rightarrow \Delta \wedge \nabla \rightarrow \Delta$)

The following proposition links up F[t] and Pro[t] and establishes the soundness of projectivity marking (the basic idea of the proof is from Icard (2012) with essential modifications).

Proposition 3.4: Given a context t with a variable x, if Pro[t] = ϕ, then F[t] is a function with signature ϕ.

Proof: We prove by induction on the structural complexity of t. If $t = x$, then since F[x] = ID, it has signature ($\uparrow \wedge \Delta \rightarrow \Delta \wedge \nabla \rightarrow \nabla$) according to the Appendix, which is equal to Pro[x].

Next suppose $t = s(u)$. We hypothesize that the proposition is true for s and u. Now there are two cases to consider. In the first case, x is a sub-term of s, then Pro[$s(u)$] = Pro[s]. We have to show that F[$s(u)$] is a function with signature Pro[s] for each possible Pro[s]. Here we will only prove the case where Pro[s] = ($\nabla \rightarrow \Delta$), as the proofs of other basic cases are similar. By the induction hypothesis, F[s] is a function with signature Pro[s], i.e. ($\nabla \rightarrow \Delta$). This means if $x \ \nabla \ y$, then F[s](x) Δ F[s](y), or equivalently F[s](x) $\wedge$ F[s](y) = 0. Based on this fact, we compute F[$s(u)$](x) $\wedge$ F[$s(u)$](y) = (λz.F[s](z)($\|u\|$))(x) $\wedge$ (λz.F[s](z)($\|u\|$))(y) = F[s](x)($\|u\|$) $\wedge$ F[s](y)($\|u\|$) =

$(F[s](x) \wedge F[s](y))(\|u\|) = 0(\|u\|) = 0$, i.e. $F[s(u)](x)\ \Delta\ F[s(u)](y)$. We have thus shown that $F[s(u)]$ has signature $(\nabla\rightarrow\Delta)$, which is equal to $Pro[s(u)]$. If $Pro[s]$ is equal to a complex signature, the only complication is that we have to consider separate cases but the basic reasoning is the same.

In the second case, x is a sub-term of u, then $Pro[s(u)] = Top[s] \circ Pro[u]$. Suppose u has type τ_1. Then s must have type $\tau_1 \xrightarrow{\phi_1} \tau_2$ for some $\phi_1 \in \Sigma$, or in other words, $\|s\|$ has signature ϕ_1. By Definition 3.2, $Top[s] = \phi_1$. If $Pro[u] = \phi_2$, then by the induction hypothesis, $F[u]$ is a function with signature ϕ_2. By Definition 2.12, we thus conclude that $F[s(u)] = \|s\| \circ F[u]$ is a function with signature $\phi_1 \circ \phi_2$, which is equal to $Top[s] \circ Pro[u]$, i.e. $Pro[s(u)]$. □

3.3 Inference Rules

We finally come to the inference rules of RC. Instead of repeating all the rules given in Icard (2012), here we will only focus on two rules that will appear in the worked examples below.

The first is the JOIN Rule[19].

$$\textbf{JOIN:}\ \frac{t_1\ \rho_1\ t_2 \qquad\qquad t_2\ \rho_2\ t_3}{t_1\ (\rho_1\ \text{JOIN}\ \rho_2)\ t_3}$$

The soundness of this rule is guaranteed by Definition 1.4 and Table 1.

The second is the Substitution Rule. Before stating this rule, we need to define one more notation (adapted from Icard (2012) and Moss (2012)).

Definition 3.5: Let t_1 be a ground term, s_1 be a sub-term of t_1, and s_2 and x be respectively a ground term and a variable of the same type as s_1. Then $t_1(s_2 \leftarrow s_1)$ represents the ground term obtained from t_1 by substituting s_2 for s_1, whereas $t_1(x \leftarrow s_1)$ represents the context obtained from t_1 by substituting x for s_1.

According to Definition 3.1, a context t_2 with a variable x is associated with a function $F[t_2]$, which can be applied to any term of the same type as x. Now Moss (2012) has

[19] Icard (2012) called this the "Composition" Rule. To avoid confusing this rule with the composition of signatures introduced in subsection 2.4 above, we prefer to call this the JOIN Rule.

proved the following proposition[20].

Proposition 3.6: Let t_2 be a context with a variable x, and s be a ground term of the same type as x. Then $F[t_2](\|s\|) = \|t_2(s \leftarrow x)\|$.

By Definition 3.5, we can obtain a context from a ground term by introducing a variable. By Proposition 3.6, we can get back a ground term by applying the context to a ground term of the same type as the variable. Thus, if t_1 is a ground term and x is a variable, we have the following relations: $F[t_1(x \leftarrow s_1)](\|s_1\|) = \|t_1(x \leftarrow s_1)(s_1 \leftarrow x)\| = \|t_1\|$ and $F[t_1(x \leftarrow s_1)](\|s_2\|) = \|t_1(x \leftarrow s_1)(s_2 \leftarrow x)\| = \|t_1(s_2 \leftarrow s_1)\|$. These two relations will be useful in the proof of the following proposition.

The following proposition links up the notions of context projectivity and the projection operation of signatures (adapted from Icard (2012). Note that Icard (2012) did not provide the proof).

Proposition 3.7: Let t be a ground term, s_1 be a sub-term of t, and s_2 and x be respectively a ground term and a variable of the same type as s_1. If $\text{Pro}[t(x \leftarrow s_1)] = \phi$ and $\|s_1\| \ \rho \ \|s_2\|$, then $\|t\| \ \phi[\rho] \ \|t(s_2 \leftarrow s_1)\|$.

Proof: By Proposition 3.4, if $\text{Pro}[t(x \leftarrow s_1)] = \phi$, then $F[t(x \leftarrow s_1)]$ is a function with signature ϕ. Now if $\|s_1\| \ \rho \ \|s_2\|$, then by Definition 2.10, we have $F[t(x \leftarrow s_1)](\|s_1\|) \ \phi[\rho] \ F[t(x \leftarrow s_1)](\|s_2\|)$. But it has been shown in the previous paragraph that $F[t(x \leftarrow s_1)](\|s_1\|) = \|t\|$ and $F[t(x \leftarrow s_1)](\|s_2\|) = \|t(s_2 \leftarrow s_1)\|$. We thus have $\|t\| \ \phi[\rho] \ \|t(s_2 \leftarrow s_1)\|$. □

We thus have the following rule.

Substitution: $$\frac{s_1 \ \rho \ s_2}{t \ \phi[\rho] \ t(s_2 \leftarrow s_1)} \quad \text{Pro}[t(x \leftarrow s_1)] = \phi$$

The soundness of this rule is guaranteed by Proposition 3.7.

3.4 Worked Examples

In this subsection, we illustrate how RC works by deriving several inclusion and exclusion inferences. First we provide a list of the ground terms and their types that will be used in the examples. To simplify matters, some phrases below are treated as

[20] Note that in Moss (2012), $t_2(x \leftarrow s)$ means the same thing as $t_2(s \leftarrow x)$ in this paper.

whole units.

Table 2 Some Ground Terms and their Types

Ground Term	**Type**
elderly, *teenagers*, *clubs*, *members*	n
are noisy	v_1
admit	v_2
male, *female*	$n \xrightarrow{(\uparrow \wedge \Delta \to \Delta \wedge \nabla \to \nabla)} n$
not	$\tau \xrightarrow{(\downarrow \wedge \Delta \to \nabla \wedge \nabla \to \Delta)} \tau$
every / all	$n \xrightarrow{(\downarrow \wedge \nabla \to \Delta)} (v_i \xrightarrow{(\uparrow \wedge \Delta \to \Delta)} v_{i-1})$
some	$n \xrightarrow{(\uparrow \wedge \nabla \to \nabla)} (v_i \xrightarrow{(\uparrow \wedge \nabla \to \nabla)} v_{i-1})$
no	$n \xrightarrow{(\downarrow \wedge \nabla \to \Delta)} (v_i \xrightarrow{(\downarrow \wedge \nabla \to \Delta)} v_{i-1})$
more than 1/2 of	$n \longrightarrow (v_i \xrightarrow{(\uparrow \wedge \Delta \to \Delta)} v_{i-1})$
less than 1/2 of	$n \longrightarrow (v_i \xrightarrow{(\downarrow \wedge \nabla \to \Delta)} v_{i-1})$
at most 1/2 of	$n \longrightarrow (v_i \xrightarrow{(\downarrow \wedge \Delta \to \nabla)} v_{i-1})$
all … except Smith	$n \xrightarrow{(\Delta \to \Delta \wedge \nabla \to \Delta)} (v_i \xrightarrow{(\Delta \to \Delta \wedge \nabla \to \Delta)} v_{i-1})$

In what follows, we will also determine the additional conditions for valid inferences (if any). But since these conditions are not parts of the projectivity signatures, their determination will be a separate process.

We also have the following set of premises as the basis of inferences:

(i) *elderly* Δ *teenagers*
(ii) *male* $\perp$ *female*
(iii) *no* $\perp$ *some*
(iv) *more than 1/2 of* $\perp$ *at most 1/2 of*
(v) *more than 1/2 of* Δ *less than 1/2 of*

From this we can already make some simple inferences involving phrases, such as:

$$\frac{elderly\ \Delta\ teenagers}{not(elderly)\ \nabla\ not(teenagers)}$$

To derive more complicated inferences, we need to make use of the Substitution Rule. For example, suppose we wish to derive the inference given in (4), reproduced below:

(4) All male members except Smith are noisy Δ
All female members except Smith are noisy

We first let t be the ground term representing "All male members except Smith are noisy". This ground term can be represented in the following form which can be derived from the terms given in Table 2:

(13) *all … except Smith*(*male*(*members*))(*are noisy*)

Then $t(x \leftarrow male)$ is equal to (10) above and $t(female \leftarrow male)$ is equal to the ground term representing "All female members except Smith are noisy". In (12), we have already computed $\text{Pro}[t(x \leftarrow male)] = (\Delta \rightarrow \Delta \wedge \nabla \rightarrow \Delta)$. Since $(\Delta \rightarrow \Delta \wedge \nabla \rightarrow \Delta)[\perp] = \Delta$, we can invoke the Substitution Rule to obtain:

$$\frac{male \perp female}{t\ \Delta\ t(female \leftarrow male)} \quad \text{Pro}[t(x \leftarrow male)] = (\Delta \rightarrow \Delta \wedge \nabla \rightarrow \Delta)$$

The conclusion $t\ \Delta\ t(female \leftarrow male)$ above is precisely the inference in (4). According to the Appendix, the 1st argument of *all … except Smith* is $\Delta \rightarrow \Delta$ without any condition. Since Δ can be seen as a component of the composite relation $\perp$ (remember that $\perp = \Delta \wedge \nabla$), this inference is valid without any additional condition.

We next derive the following inference that is more complicated and instructive:

(14) (Given that every club admits some member)
More than 1/2 of the clubs admit no male members $\leq$
More than 1/2 of the clubs admit some female members

To derive this inference, we have to proceed in two stages at different levels of the sentence. But before doing this, we first let t_1 be the ground term representing "More than 1/2 of the clubs admit no male members", which can be represented in the following form:

(15) *more than 1/2 of*(*clubs*)(*no*(*male*(*members*))(*admit*))

The above expression can be derived according to the types of terms given in Table 2. Note that in this expression we treat *more than 1/2 of* as of type $n \longrightarrow (v_1 \xrightarrow{(\uparrow\wedge\Delta\to\Delta)} v_0)$ and *no* as of type $n \xrightarrow{(\downarrow\wedge\nabla\to\Delta)} (v_2 \xrightarrow{(\downarrow\wedge\nabla\to\Delta)} v_1)$.

In the first stage, we focus on the phrase "admit no male members" which, according to (15), may be represented by the ground term t_2 = *no*(*male*(*members*))(*admit*), from which we can obtain the context $t_2(x\leftarrow male)$ = *no*(*x*(*members*))(*admit*). By Definition 3.3, the projectivity of this context is $\text{Pro}[t_2(x\leftarrow male)] = \text{Top}[no] \circ \text{Pro}[x] = (\downarrow \wedge \nabla\to\Delta) \circ (\uparrow \wedge \Delta\to\Delta \wedge \nabla\to\nabla) = (\downarrow \wedge \nabla\to\Delta)$. Since $(\downarrow \wedge \nabla\to\Delta)[\perp] = \Delta$, we can invoke the Substitution Rule to obtain:

$$\frac{male \perp female}{t_2 \;\Delta\; t_2(female\leftarrow male)} \quad \text{Pro}[t_2(x\leftarrow male)] = (\downarrow \wedge \nabla\to\Delta)$$

Writing $t_3 = t_2(female\leftarrow male)$ = *no*(*female*(*members*))(*admit*), we next obtain the context $t_3(x\leftarrow no)$ = *x*(*female*(*members*))(*admit*). By Definition 3.3, the projectivity of this context is $\text{Pro}[t_3(x\leftarrow no)] = \text{Pro}[x] = (\uparrow \wedge \Delta\to\Delta \wedge \nabla\to\nabla)$. Since $(\uparrow \wedge \Delta\to\Delta \wedge \nabla\to\nabla)[\perp] = \perp$, we can invoke the Substitution Rule to obtain:

$$\frac{no \perp some}{t_3 \perp t_3(some\leftarrow no)} \quad \text{Pro}[t_3(x\leftarrow no)] = (\uparrow \wedge \Delta\to\Delta \wedge \nabla\to\nabla)$$

Writing $t_4 = t_3(some\leftarrow no)$ = *some*(*female*(*members*))(*admit*), since Δ JOIN $\perp = \leq$ according to Table 1, we can obtain the following result by invoking the JOIN Rule:

$$\frac{t_2 \;\Delta\; t_3 \qquad\qquad t_3 \perp t_4}{t_2 \leq t_4}$$

We next derive the condition for the validity of the above inference. As discussed in Subsection 2.5 for (8), we know that the two predicates t_2 = *no*(*male*(*members*))(*admit*) and t_3 = *no*(*female*(*members*))(*admit*) satisfy the Δ relation given the condition that every club admits some member (assuming a universe composed of "clubs"). Now $t_3 \perp t_4$ is not associated with any additional condition, and so $t_2 \leq t_4$ inherits the condition associated with $t_2\ \Delta\ t_3$. Thus, we obtain the following inference at the phrase level:

(16) (Given that every club admits some member)
admit no male members ≤ admit some female members

In the second stage, we work on t_1, from which we can obtain the context $t_1(x \leftarrow t_2)$ = *more than 1/2 of*(*clubs*)(*x*). By Definition 3.3, the projectivity of this context is $\text{Pro}[t_1(x \leftarrow t_2)] = \text{Top}[\textit{more than 1/2 of}(\textit{clubs})] \circ \text{Pro}[x] = (\uparrow \wedge \Delta \rightarrow \Delta) \circ (\uparrow \wedge \Delta \rightarrow \Delta \wedge \nabla \rightarrow \nabla) = (\uparrow \wedge \Delta \rightarrow \Delta)$. Since $(\uparrow \wedge \Delta \rightarrow \Delta)[\leq] = \leq$, we can invoke the Substitution Rule to obtain:

$$\frac{t_2 \leq t_4}{t_1 \leq t_1(t_4 \leftarrow t_2)} \quad \text{Pro}[t_1(x \leftarrow t_2)] = (\uparrow \wedge \Delta \rightarrow \Delta)$$

Since $t_1(t_4 \leftarrow t_2)$ = *more than 1/2 of*(*clubs*)(*some*(*female*(*members*))(*admit*)), we obtain *more than 1/2 of*(*clubs*)(*no*(*male*(*members*))(*admit*)) ≤ *more than 1/2 of*(*clubs*)(*some*(*female*(*members*))(*admit*)). According to the Appendix, *more than 1/2 of* is not associated with any additional condition. Thus, the above inference inherits the condition that every club admits some member obtained in the first stage, and so we finally obtain our desired inference in (14).

Note that if the inference rules are used in a different order, one may not be able to yield the same desired result. For example, to derive (14) discussed above, it is necessary to use the JOIN Rule at the first stage before the final use of the Substitution Rule at the second stage. In brief, if we only use the Substiution Rule without using the JOIN Rule in the first stage, we will derive the following inference between two phrases (which is the same as (8)):

(17) (Given that every club admits some member)
admit no male members Δ admit no female members

If we then use the Substitution Rule to be followed by the JOIN Rule (using the relation *more than 1/2 of* ⊥ *at most 1/2 of*) in the second stage, we will finally derive the following inference at the sentence level (which is weaker than the one in (14)):

(18) (Given that every club admits some member)

More than 1/2 of the clubs admit no male members ≤
At most 1/2 of the clubs admit no female members

This shows that when using RC to derive inferences, one needs to make smart choices of the use of rules.

3.5 Some Properties of RC

Being a revised version of C, can RC derive all inferences involving the classical quantifiers and the two basic functions ID and *not* that are derivable under C? Here I only consider inferences involving the classical quantifiers because as shown above, among the quantifiers that are usually studied under GQT, only the classical quantifiers possess the +-× properties. Note that the definitions of the +-× properties are related to inferences involving the join and meet operations in a Boolean algebra. For instance, as *some* is additive in the 2^{nd} argument, we have the following valid inference:

(19) Somebody is singing or dancing ≡
Somebody is singing or somebody is dancing

However, C does not deal with such kinds of inferences. In other words, C does not give full play to the potential inferences that the classical quantifiers may have by virtue of their +-× properties, but instead only deals with inferences involving the seven relations given in Definition 1.1, which are exactly the inferences that MPs and OPs are defined to deal with. Thus, although the projectivity signatures under C have different meanings than those under RC, the two types of signatures in fact play the same role under their respective systems.

By using Propositions 1.7, 1.8 and 2.7, we can even establish a correspondence between the signatures under C and some of the signatures under RC in the sense that corresponding signatures share the same inference properties under their respective systems, as given in Table 3.

Table 3 Correspondence between Signatures under C and RC

Signature under C	Signature under RC
+ (increasing)	↑
– (decreasing)	↓
⨹ (additive)[a]	(↑ ∧ ∇→∇)
⊞ (multiplicative)	(↑ ∧ Δ→Δ)
⨺ (anti-additive)	(↓ ∧ ∇→Δ)
⊟ (anti-multiplicative)	(↓ ∧ Δ→∇)
⊕ (additive and multiplicative)	(↑ ∧ Δ→Δ ∧ ∇→∇)
⊖ (anti-additive and anti-multiplicative)	(↓ ∧ Δ→∇ ∧ ∇→Δ)
• (general)	•

[a] For lack of the symbols used in Icard (2012), here I use a triangle with a plus (i.e. ⨹) and a triangle with a minus (i.e. ⨺) to represent the additive and anti-additive properties, respectively. Note that in Icard (2012), a rhombus is used instead of a triangle.

Lemmas 2.4 and 2.7 in Icard (2012) give the table of projection of a relation under a signature and the table of composition of two signatures under C, respectively. These two tables in general give correct results of the two operations of projection and composition except that the table given in Lemma 2.7 contains the following three incorrect results: ⨺ ∘ ⨺ = –, ⊟ ∘ ⊞ = – and ⊖ ∘ ⊟ = ⨹. The correct results should be ⨺ ∘ ⨺ = +, ⊟ ∘ ⊞ = ⊟ and ⊖ ∘ ⊟ = ⊞. Here I will prove ⊟ ∘ ⊞ = ⊟. The proofs of the other results (as well as the correct results in Icard (2012)) are similar. Let f_1 and f_2 be functions with signatures ⊞ and ⊟, respectively. We need to show that ⊟ represents the strongest +-× property that $f_2 \circ f_1$ has. Let x and y be elements. We compute $f_2 \circ f_1(x \wedge y) \equiv f_2(f_1(x \wedge y)) \equiv f_2(f_1(x) \wedge f_1(y))$ (because f_1 is multiplicative) $\equiv f_2(f_1(x)) \vee f_2(f_1(y))$ (because f_2 is anti-multiplicative) $\equiv f_2 \circ f_1(x) \vee f_2 \circ f_1(y)$. This shows that $f_2 \circ f_1$ is anti-multiplicative, i.e. it has signature ⊟. To show that anti-multiplicativity is the strongest such property, we have to show that $f_2 \circ f_1$ is not also anti-additive. To do this, we let f_1 = *everybody* and f_2 = *not everybody* in a universe composed of "persons". Note that *everybody* and *not everybody* are functions with signatures ⊞ and ⊟, respectively. We then show that *not everybody* ∘ *everybody* is not anti-additive by showing that "Not everybody loves or hates everybody"[21] is not equivalent to "Not everybody loves everybody and not everybody hates everybody". To this end, we construct the following counterexample. Let U' = {x, y}, loves = {(x, x), (y, y)}, hates = {(x, y), (y, x)}. In this model, "Not everybody loves

[21] In GQT, "Not everybody loves or hates everybody" can be represented as *not everybody*(*everybody*)(*loves* ∨ *hates*).

everybody and not everybody hates everybody" is true but "Not everybody loves or hates everybody" is false (because it is true that everybody loves or hates everybody).

After correcting the mistakes in the aforesaid two tables, one can then see that the above pairs of corresponding signatures share the same results of projection and composition by checking each entry in the tables. For example, according to Lemma 2.4 in Icard (2012)[22], under C we have $\boxplus[\Delta] = \Delta$. By Table 3 above, the corresponding result under RC should be $(\uparrow \wedge \Delta\to\Delta)[\Delta] = \Delta$, which is correct. Moreover, according to Lemma 2.7 in Icard (2012), under C we have $\boxminus \circ \boxminus = +$. By Table 3, the corresponding result under RC should be $(\downarrow \wedge \Delta\to\nabla) \circ (\downarrow \wedge \Delta\to\nabla) = \uparrow$, which is also correct.

Moreover, the signatures of the classical quantifiers and the two basic functions ID and *not* under C and RC are also consistent with the above correspondence. For example, the signature of *every / all* in respect of its 2nd argument is $\boxplus$ under C and $(\uparrow \wedge \Delta\to\Delta)$ under RC, and this is consistent with the correspondence between $\boxplus$ and $(\uparrow \wedge \Delta\to\Delta)$ given in Table 3.

Under both C and RC, the derivation of an inference is based on the following elements: (i) the signatures of the relevant quantifiers and functions, (ii) the projection of the relevant relations under the relevant signatures, (iii) the composition of the relevant signatures, (iv) the JOIN operation among the relevant relations, (v) the projectivity of the relevant ground terms and contexts, and (vi) the relevant inference rules. For elements (i) – (iii), there is a correspondence between the result derived under C and that derived under RC as discussed above. For elements (iv) – (vi), the relevant results, definitions and rules under C and RC are exactly the same.

We can thus conclude that for every inference involving the classical quantifiers and the two basic functions that is derivable under C, there is a corresponding version under RC, or in other words, all inferences involving the classical quantifiers and the two basic functions that are derivable under C are also derivable under RC. For example, if we let $t = all(members)(elderly)$ and assume *elderly* Δ *teenagers*, then we have $\mathrm{Pro}[t(x\leftarrow elderly)] = \boxplus$ under C. By the Substitution Rule, we then have

$$\frac{elderly \;\Delta\; teenagers}{t \;\Delta\; t(teenagers\leftarrow elderly)} \quad \mathrm{Pro}[t(x\leftarrow elderly)] = \boxplus$$

[22] To facilitate comparison, in what follows we modify the notations for the seven relations and projection used in Icard (2012).

because we have $\boxplus[\Delta] = \Delta$ under C. The above inference can be written in ordinary English as (note that C does not deal with conditions (if any) associated with inferences):

(20) All members are elderly Δ All members are teenagers

Now the above derivation has a corresponding version under RC. By Definition 3.3, we have Pro[*t*(*x*←*elderly*)] = (↑ ∧ Δ→Δ) under RC. By the Substitution Rule, we then have

$$\frac{\textit{elderly}\ \Delta\ \textit{teenagers}}{t\ \Delta\ t(\textit{teenagers} \leftarrow \textit{elderly})}\ \text{Pro}[t(x \leftarrow \textit{elderly})] = (\uparrow \wedge \Delta \rightarrow \Delta)$$

because we have (↑ ∧ Δ→Δ)[Δ] = Δ under RC. Thus, the inference in (20) that is derivable under C is also derivable under RC (which can also handle the condition associated with that inference, i.e. there is some member). Moreover, since there are some signatures under RC which have no corresponding signatures under C (for example, there is no signature under C corresponding to (Δ→Δ) under RC), we know that some inferences not derivable under C are derivable under RC.

Like C, RC is incomplete. For example, it is not possible to derive the following inference that involves the contradictory relation between *no* and *some* as well as the exclusive relation between *more than 1/2 of* and *less than 1/2 of* under RC (unless the number of "clubs" in question is odd, in which case *more than 1/2 of* and *less than 1/2 of* are contradictory to each other):

(21) More than 1/2 of the clubs admit no male members ≡
Less than 1/2 of the clubs admit some male members

But the above inference is valid (regardless of the oddness/evenness of the number of "clubs") and can be derived using the Natural Logic discussed in Keenan (2003, 2008), which is based on the concepts of complement (also called outer negation) and post-complement (also called inner negation). Under RC, we can only derive the following weaker inference:

(22) More than 1/2 of the clubs admit no male members ≤
At most 1/2 of the clubs admit some male members

The above facts show that there are still unsolved questions. Can we obtain a complete system by incorporating some inference rules from Keenan's Natural Logic as well as other inference rules into RC? What other techniques do we need for deriving valid inferences under RC? Moreover, in the worked examples above, the determination of conditions for valid inferences is a separate process from the operation (projection and composition) of projectivity signatures and use of inference rules. Can we incorporate these conditions into the signatures and determine these conditions in a parallel fashion with the operation of signatures and use of inference rules? These issues have to be left for future work.

4. Conclusion

This paper has mainly borrowed ideas from Icard (2012) and Chow (2012, 2017), each of which has its own strengths and weaknesses. Icard (2012) has developed a formal reasoning system that can handle inclusion and exclusion reasoning at the same time. But the system only covers a small range of quantifiers because it is based on the +-× properties, which are too strong and have excluded many quantifiers that satisfy important exclusion inferences. Neither does his system handle the conditions for valid inferences. Chow's theory covers a much wider range of quantifiers because it is based on the notions of MPs and OPs, which are more appropriate for the study of inclusion and exclusion reasoning. His theory does include the conditions for valid inferences between sentences, but has not considered the conditions for valid inferences between predicates. Moreover, under his theory, inclusion reasoning and exclusion reasoning are two separate realms.

By combining the merits and avoiding the demerits of Icard (2012) and Chow (2012, 2017), we have developed a formal system, namely RC, which can derive all inferences involving the classical quantifiers and the two basic functions that are derivable under Icard's system as well as some inferences not derivable under that system. Some of these inferences are associated with conditions which are summarized in the Appendix. We have also discussed an example which shows that the order of using the inference rules may affect the derivation results.

Moreover, we have also devised a transparent symbolism for the 15 possible projectivity signatures by borrowing ideas from Chow (2012, 2017). This symbolism greatly facilitates the computations of projection and composition, making it unnecessary to provide computation tables for these operations as was done in Icard (2012). Such computation tables will be formidable under RC as there are now 15 signatures. For example, a full table for composition will be a 15 × 15 table.

While RC is incomplete, this paper has contributed to deepening our understanding of some important properties of generalized quantifiers and inferences associated with these properties, especially the exclusion inferences. We believe that RC will have a role to play in the development of Natural Logic (or Natural Language Inferences).

Appendix: Signatures and Conditions of some Functions

This Appendix provides a list of some important functions (including logical operators / generalized quantifiers) and their projectivity signatures and sufficient conditions for valid inferences. For each function, the signature listed in the middle column below represents the strongest MP and/or OP it possesses. For quantifiers with more than one argument, the signature in respect of each argument is specified. Otherwise, the signature in respect of that argument is understood to be ●. For a function with complex signature, different components of the signature may be associated with different conditions. For clarity, the components of a complex signature are listed on separate rows with their corresponding conditions given in the right column (where "–" represents no condition). In what follows, A and B represent the 1st (unary) argument and the 2nd (n-ary) argument of a determiner, while A, B_1 and B_2 represent the 1st (unary) argument and the 2nd and 3rd (n-ary) arguments of an identity comparative quantifier. For example, the information given below shows that *every / all* has complex signature "(↓ ∧ ∇→Δ)" in respect of the 1st argument. Moreover, *every / all* is decreasing (i.e. ↓) with no additional condition and anti-exhaustive (i.e. ∇→Δ) given the additional condition that $\forall x_1 \ldots x_{n-1}.\{x_n: (x_1, \ldots x_n) \in B\} \neq U'$.

Function		Signature	Condition
ID		↑	–
		Δ→Δ	–
		∇→∇	–
not		↓	–
		Δ→∇	–
		∇→Δ	–
Smith, i.e. singular proper name		↑	–
		Δ→Δ	–
		∇→∇	–
every / all	1st	↓	–
		∇→Δ	$\forall x_1 \ldots x_{n-1}.\{x_n: (x_1, \ldots x_n) \in B\} \neq U'$
	2nd	↑	–
		Δ→Δ	$A \neq \varnothing$
some	1st	↑	–
		∇→∇	$\forall x_1 \ldots x_{n-1}.\{x_n: (x_1, \ldots x_n) \in B\} \neq \varnothing$
	2nd	↑	–
		∇→∇	$A \neq \varnothing$

no	1st	$\downarrow$	–
		$\nabla \rightarrow \Delta$	$\forall x_1 \ldots x_{n-1}.\{x_n: (x_1, \ldots x_n) \in B\} \neq \varnothing$
	2nd	$\downarrow$	–
		$\nabla \rightarrow \Delta$	$A \neq \varnothing$
not every / not all	1st	$\uparrow$	–
		$\nabla \rightarrow \nabla$	$\forall x_1 \ldots x_{n-1}.\{x_n: (x_1, \ldots x_n) \in B\} \neq U'$
	2nd	$\downarrow$	–
		$\Delta \rightarrow \nabla$	$A \neq \varnothing$
more than n, *at least n*	both: $\uparrow$		–
fewer than n, *at most n*	both: $\downarrow$		–
most	2nd	$\uparrow$	–
		$\Delta \rightarrow \Delta$	–
more than r of ($1/2 \leq r < 1$)	2nd	$\uparrow$	–
		$\Delta \rightarrow \Delta$	–
more than r of ($0 < r < 1/2$)	2nd	$\uparrow$	–
		$\nabla \rightarrow \nabla$	–
at least r of ($1/2 < r < 1$)	2nd	$\uparrow$	–
		$\Delta \rightarrow \Delta$	–
at least r of ($0 < r \leq 1/2$)	2nd	$\uparrow$	–
		$\nabla \rightarrow \nabla$	–
less than r of ($1/2 < r < 1$)	2nd	$\downarrow$	–
		$\Delta \rightarrow \nabla$	–
less than r of ($0 < r \leq 1/2$)	2nd	$\downarrow$	–
		$\nabla \rightarrow \Delta$	–
at most r of ($1/2 \leq r < 1$)	2nd	$\downarrow$	–
		$\Delta \rightarrow \nabla$	–
at most r of ($0 < r < 1/2$)	2nd	$\downarrow$	–
		$\nabla \rightarrow \Delta$	–
exactly r of ($1/2 < r < 1$)	2nd: $\Delta \rightarrow \Delta$		–
exactly r of ($0 < r < 1/2$)	2nd: $\nabla \rightarrow \Delta$		–
between q and r of ($1/2 < q < r < 1$)	2nd: $\Delta \rightarrow \Delta$		–
between q and r of ($0 < q < r < 1/2$)	2nd: $\nabla \rightarrow \Delta$		–
more than r or less than q of ($1/2 < q < r < 1$)	2nd: $\Delta \rightarrow \nabla$		–

more than r or less than q of (0 < q < r < 1/2)	2nd: $\nabla \to \nabla$		–
all ... except Smith	1st	$\Delta \to \Delta$	–
		$\nabla \to \Delta$	$\forall x_1 \ldots x_{n-1}.\{x_n: (x_1, \ldots x_n) \in B\} \cup \{s\} \neq U'$
	2nd	$\Delta \to \Delta$	$A - \{s\} \neq \varnothing$
		$\nabla \to \Delta$	–
no ... except Smith	1st	$\Delta \to \Delta$	–
		$\nabla \to \Delta$	$\forall x_1 \ldots x_{n-1}.\{x_n: (x_1, \ldots x_n) \in B\} - \{s\} \neq \varnothing$
	2nd	$\Delta \to \Delta$	–
		$\nabla \to \Delta$	$A - \{s\} \neq \varnothing$
the same ... as ...	1st	$\downarrow$	–
		$\nabla \to \Delta$	$\forall x_1 \ldots x_{n-1}.\{x_n: (x_1, \ldots x_n) \in B_1\} \neq \{x_n: (x_1, \ldots x_n) \in B_2\}$
	2nd	$\Delta \to \Delta$	$\forall x_1 \ldots x_{n-1}.A \cap \{x_n: (x_1, \ldots x_n) \in B_2\} \neq \varnothing$
		$\nabla \to \Delta$	$\forall x_1 \ldots x_{n-1}.A - \{x_n: (x_1, \ldots x_n) \in B_2\} \neq \varnothing$
	3rd	$\Delta \to \Delta$	$\forall x_1 \ldots x_{n-1}.A \cap \{x_n: (x_1, \ldots x_n) \in B_1\} \neq \varnothing$
		$\nabla \to \Delta$	$\forall x_1 \ldots x_{n-1}.A - \{x_n: (x_1, \ldots x_n) \in B_1\} \neq \varnothing$
different ... than ...	1st	$\uparrow$	–
		$\nabla \to \nabla$	$\forall x_1 \ldots x_{n-1}.\{x_n: (x_1, \ldots x_n) \in B_1\} \neq \{x_n: (x_1, \ldots x_n) \in B_2\}$
	2nd	$\nabla \to \nabla$	$\forall x_1 \ldots x_{n-1}.A - \{x_n: (x_1, \ldots x_n) \in B_2\} \neq \varnothing$
		$\Delta \to \nabla$	$\forall x_1 \ldots x_{n-1}.A \cap \{x_n: (x_1, \ldots x_n) \in B_2\} \neq \varnothing$
	3rd	$\nabla \to \nabla$	$\forall x_1 \ldots x_{n-1}.A - \{x_n: (x_1, \ldots x_n) \in B_1\} \neq \varnothing$
		$\Delta \to \nabla$	$\forall x_1 \ldots x_{n-1}.A \cap \{x_n: (x_1, \ldots x_n) \in B_1\} \neq \varnothing$

References

Beghelli, F. (1994). Structured Quantifiers. In Kanazawa, M. & Piñón, C. (Eds.), *Dynamics, Polarity and Quantification* (pp. 119-143). Stanford: CSLI Publications.

van Benthem, J. (1986). *Essays in Logical Semantics*, Dordrecht: Reidel.

Chow, K.F. (2012). Generalizing Monotonicity Inferences to Opposition Inferences. In Aloni, M. et al (Eds.), *Proceedings of the 18th Amsterdam Colloquium* (pp. 281-290). Berlin: Springer.

Chow, K.F. (2017), Opposition Inferences and Generalized Quantifiers. In Beziau, J.-Y. and Georgiorgakis, S. (Eds.), *New Dimensions of the Square of Opposition* (pp. 155-199), München: Philosophia Verlag GmbH.

Icard, T.F. (2012). Inclusion and Exclusion in Natural Language. *Studia Logica*, 100(4), 705-725.

Icard, T.F. (2014). Higher-Order Syllogistics. In: Morrill G. et al (Eds.), *Formal Grammar* (pp. 1-14). Berlin: Springer.

Icard, T. F. & Moss, L. S. (2014). Recent Progress on Monotonicity. *Linguistic Issues in Language Technology*, 9(7), 167-194.

Keenan, E.L. (2003). Excursions in Natural Logic. In Casadio, C. et al (Eds.), *Language and Grammar: Studies in Mathematical Linguistics and Natural Language* (pp. 31-52). Stanford: CSLI.

Keenan, E.L (2008). Further Excursions in Natural Logic: The Mid-Point Theorems. In Hamm, F. & Kepser, S. (Eds.), *Logics for Linguistic Structures* (pp. 87-104). Berlin: Mouton de Gruyter.

Keenan, E.L. & Faltz, L.M. (1985). *Boolean Semantics for Natural Language*. Dordrecht: Reidel.

Keenan, E.L. & Westerståhl, D. (2011). Generalized Quantifiers in Linguistics and Logic. In van Benthem, J. & ter Meulen, A. (Eds.), *Handbook of Logic and Language (Second edition)* (pp. 859-910). Amsterdam: Elsevier Science.

MacCartney, B. (2009). *Natural Language Inference*, Ph.D. Dissertation. Stanford University.

MacCartney, B., & Manning, C.D. (2009). An extended model of natural logic. In *Proceedings of the Eighth International Conference on Computational Semantics* (pp. 140-156).

Moss, L. S. (2012). The Soundness of Internalized Polarity Marking. *Studia Logica*, 100(4), 683 – 704.

Peters, S. & Westerståhl, D. (2006). *Quantifiers in Language and Logic*. Oxford: Clarendon Press.

Sánchez Valencia, V. (1991). *Studies on Natural Logic and Categorial Grammar*, Ph.D. Dissertation, Universiteit van Amsterdam.