1 Introduction

A definite description is a phrase that denotes an object as the unique thing satisfying a certain description, for example, ‘the present king of France’, ‘the first dog born at sea’, ‘the bed’, ‘John’s father’, and perhaps also ‘2 + 6’. Although not all definite descriptions have the form ‘the Φ’, they can all be rephrased that way, so we’ll talk in what follows as if all definite descriptions have that form.

2 Grammar and Semantics

There is no consensus among linguists and philosophers of language about the semantics of definite descriptions in natural languages. One camp holds that definite descriptions are quantifiers—so that ‘the F’ classes with ‘most Fs’ and ‘some Fs’—while the other holds that they are referring expressions, more like individual constants than like quantifiers. Here I’ll present the quantificational alternative. Later, when we’re discussing facts and the “slingshot argument,” we’ll consider how a non-quantificational account of definite descriptions might go.

2.1 Quantifiers in general

The quantifiers we use in first-order logic—‘∀’ and ‘∃’—are not the only quantifiers there are. Consider ‘∃²’, which we can read “there exist at least two….” It seems just as “quantificational” as ‘∃’, since the truth of ‘∃²xΦx’ in a model depends solely on how many things in the model satisfy Φx. ‘∃²’ can be defined in terms of ‘∃’ and ‘=’, but there are other quantifiers that cannot be defined in the language of first-order logic with identity.

One is ‘most’. Consider how you might render ‘Most cows eat grass’ in the language of first-order logic. You will quickly find that there is no way to do it. In fact, it turns out that ‘most’ can’t be captured by any unary quantifier. (A unary quantifier is one that takes one formula as argument and forms a sentence.) You might think that the quantifier ‘M’ would do the trick, where ‘MxΦx’ is true in a model just in case over half the objects in the domain satisfy Φx in that model. But this won’t help with ‘Most cows eat grass’, since ‘Mx(Cx ⊃ Gx)’ will be true in any model where cows make up fewer than half the objects in the domain, no matter how many of them eat grass, while ‘Mx(Cx ∧ Gx)’ will be true only in models where cows are the majority of objects in the domain. (Do you see why?)

The solution is to treat ‘most’ as a binary quantifier—one that takes two formulas and makes a formula. Thus, if Φ and Ψ are formulas, so is ‘[most x Φx](Ψx)’. This formula is
2.2 Definite descriptions as quantifiers

true in a model just in case most of the objects in the domain that satisfy \( \Phi x \) also satisfy \( \Psi x \).

In general, where \( Q \) is a binary quantifier, \( [Q_x \Phi]x[\Psi x] \) is true in a model just in case some relation holds between the following quantities:

- \( \#D \) = the number of objects in the domain
- \( \#\Phi \) = the number of objects in the domain that satisfy \( \Phi x \)
- \( \#\Psi \) = the number of objects in the domain that satisfy \( \Psi x \)
- \( \#(\Phi \land \Psi) \) = the number of objects in the domain that satisfy both \( \Phi \) and \( \Psi \).

In the case of ‘most’, the relation that must hold is

\[
\frac{\#(\Phi \land \Psi)}{\#\Phi} > \frac{1}{2}
\]

In the case of ‘some’, it is

\( \#(\Phi \land \Psi) \geq 1 \)

In the case of ‘every’, it is

\( \#(\Phi \land \Psi) = \#\Phi \)

What makes these expressions quantifiers is that they only care about how many things satisfy their associated subformulas, not about what these things are like.

2.2 Definite descriptions as quantifiers

It turns out that ‘the’ can be treated as a binary quantifier as well. We say that ‘[the] \( \Phi x \)[\Psi x]’ is true in a model just in case there is just one \( \Phi \) and every \( \Phi \) is \( \Psi \):

\[
\#\Phi = 1 \text{ and } \#(\Phi \land \Psi) = \#\Phi
\]

This is a purely quantificational condition. Whether it is satisfied depends only on how many \( \Phi \)s and \( \Psi \)s there are, and not at all on which things are \( \Phi \) and \( \Psi \).

So is ‘the’ in English the quantifier ‘the’? As I mentioned, this is a highly contentious question. The match is pretty close. We don’t use ‘the’ \( F \) when there is more than one (salient) \( F \), or when there aren’t any.\(^1\) So it is tempting to suppose that ‘the \( F \) is \( G \)’ just means that there is a unique (salient) \( F \) and it is \( G \). If that’s right, then ‘the’ in English is a quantifier.

\(^1\)The qualification “salient” is needed because we can say that the book is on the table without committing ourselves to the claim that there is only one book and one table in the universe. Plausibly, the domain of quantification is restricted to objects that are salient in the conversational context. There are other solutions to the problem that do not require flexibility in the domain, but this solution is perhaps the most straightforward.
However, the quantificational analysis also predicts that a sentence like ‘The present king of France is bald’ should come out false. And that has seemed odd to many philosophers. Surely, they say, if there is no present king of France, then ‘The present king of France is bald’ fails to make a determinate claim—and so fails to be either true or false. One who uses this sentence to make an assertion may presuppose that there is a present king of France, but it seems odd to say (as on the quantificational account) that the sentence implies this.

Historically, the quantificational account of ‘the’ is due to Russell, while the nonquantificational approach was championed by Frege and Strawson. If you’d like to explore this debate, an excellent place to start is Gary Ostertag’s anthology *Definite Descriptions: A Reader* (Cambridge: MIT Press, 1998). Stephen Neale’s book *Descriptions* (Cambridge: MIT Press, 1990) is an influential presentation and defense of the quantificational view.

### 2.3 Definite descriptions as “incomplete symbols”

Our approach has been to take ‘the F’ seriously as a meaningful unit and say how it operates semantically. Russell’s actual approach was a bit different. He did not have the theory of generalized quantifiers at his disposal. So instead of representing ‘the F’ as a quantifier, as we did above, he represented it as a kind of term, which he then showed how to eliminate in favor of (standard) quantifiers.

Russell’s definite description terms are constructed using an upside-down iota (‘ι’). ‘ι’ is a variable-binding operator, just like ‘∀’ and ‘∃’, but unlike them it forms a term, not a formula. If Φ is a formula and α is a variable, then ιαΦ is a term. For example:

- ‘the F’: ‘ιxF x’
- ‘the F that Gs b’: ‘ιx(F x ∧ Gx b)’

Terms formed using ‘ι’ can occur in formulas wherever other kinds of terms (variables and individual constants) can occur. For example:

- ‘the F is H’: ‘ΗιxF x’
- ‘the F Gs the H that Gs the K’: ‘G(ιxF x)(ιy(Hx ∧ Gx ιKy))’

(Put parentheses around iota-terms when there is threat of ambiguity.)

Russell understood the terms formed using his upside-down iota not as genuine terms, but as “incomplete symbols.” In effect, he took formulas containing iota-terms to be abbreviations for formulas not containing them. Recall that on the quantificational reading, there are two conditions for the truth of ‘[the F x][Ψ x]’.

(a) there must be exactly one Φ, and

(b) all the Φs must be Ψ.

We can express these in logical notation as follows:
2.3 Definite descriptions as “incomplete symbols”

(a°) \( \exists x (\Phi x \land \forall y (\Phi y \supset y = x)) \)

(there is at least one \( x \) such that everything that is a \( \Phi \) is identical with it)

(b°) \( \forall x (\Phi x \supset \Psi x) \)

Conjoining these, we get the equivalence we were looking for:

(R1) \((the \Phi x)(\Psi x) \equiv (\exists x (\Phi x \land \forall y (\Phi y \supset y = x)) \land \forall x (\Phi x \supset \Psi x))\)

or equivalently,

(R2) \((the \Phi x)(\Psi x) \equiv \exists x (\Phi x \land \forall y (\Phi y \supset y = x)) \land \Psi x)\).

If we use Russell’s upside-down iota operator instead of the explicit quantificational notation, we get the following:

(R3) \(\Psi_x \Phi x \equiv \exists x (\Phi x \land \forall y (\Phi y \supset y = x)) \land \Psi x)\).

However, there is a problem with (R3) as it stands. The problem stems from the fact that definite descriptions, like other quantifiers, have scopes. Thus, just as we can distinguish between

\[-[Every_x \Phi x](R) \quad \text{Not every philosopher is rich} \quad (1)\]

and

\([Every_x \Phi x](\neg R) \quad \text{Every philosopher is not-rich, i.e., no philosopher is rich} \quad (2)\]

so we can distinguish between

\[-[the_x \Phi x](R) \quad \text{It is not the case that the philosopher is rich} \quad (3)\]

and

\([the_x \Phi x](\neg R) \quad \text{The philosopher is not-rich.} \quad (4)\]

In (3), the quantifier takes “narrow scope” with respect to the negation—that is, it occurs within the negation’s scope. In (4), the quantifier takes “wide scope” with respect to the negation—that is, the negation occurs within its scope. (3) and (4) are not equivalent: if there is not a unique philosopher, then (3) is true, but (4) is false.

How can we represent these two distinct quantificational readings of “The philosopher is not rich” using the iota-term notation? As it stands, we can’t. The formula

\(\neg R_{x}{\Phi x} \quad (5)\)

is ambiguous between a narrow-scope and a wide-scope reading. (R3), as it is currently stated, says that it is equivalent to both

\(\neg \exists x (P x \land \forall y (P y \supset y = x)) \land R x) \quad (6)\)

January 28, 2016
(taking $\Psi x$ to be ‘$Rx$’) and

$$\exists x (P x \land \forall y (P y \supset y = x) \land \neg Rx)$$

(taking $\Psi x$ to be ‘$\neg Rx$’). But (5) can’t be equivalent to both (6) and (7), because they aren’t equivalent to each other! Our rule (R3) is not sound.

What we need to solve this problem is a way of indicating the scope of definite descriptions written using iota-terms. Russell and Whitehead do this in *Principia Mathematica* by putting a copy of the iota term in square brackets in front of the description’s scope. Since we’ll be using square brackets for another purpose, I suggest we use angle-brackets instead. So, the narrow-scope reading of (5) would be written

$$\neg \langle \iota x P x \rangle R: x P x$$

and the wide-scope reading would be written

$$\langle \iota x P x \rangle \neg R: x P x.$$  

(Note that the bracketed iota-term serves no function other than to indicate scope.) Using this notation, we can write a (sound) version of our equivalence rule:

$$\text{(R4)} \quad \langle \iota x \Phi x \rangle \Psi: x \Phi x \equiv \exists x (\Phi x \land \forall y (\Phi y \supset y = x) \land \Psi x).$$

Following Russell and Whitehead, we will adopt the convention that if the scope-indicator is omitted, the iota-term will be assumed to have the narrowest possible scope. Thus,

$$\neg R: x P x$$

is to be read as

$$\neg \langle \iota x P x \rangle R: x P x,$$

which according to (R4) is equivalent to

$$\neg \exists x (P x \land \forall y (P y \supset y = x) \land Rx).$$

3 Proofs

Since for any formula containing the quantifier ‘the’ or Russell’s ‘$\iota$’ operator we can always find an equivalent formula that uses only the standard quantifiers, it is easy to extend our proof system to accommodate definite descriptions.

**Russellian Equivalences:** Whenever you have the right-hand side of an instance of (R1), (R2), or (R4), you may replace it with the left-hand side, and vice versa, with justification “RE.” This is a substitution rule, so it may be used even on subformulas.

---

Examples:

1. \[ \text{[the}_x(Fx \land Gx)](Hx) \]
   hyp.

2. \[ \exists x((Fx \land Gx) \land \forall y((Fy \land Gy) \supset y = x) \land Hx) \]
   RE 1 (\( \Phi x = Fx \land Gx, \Psi x = Hx \))

1. \[ \exists x(Gx \land \forall y(y \supset x) \land (Fx \supset Hx)) \]
   hyp.

2. \[ \langle \text{i}xGx \rangle (Fx \supset Hx) \]
   RE 1 (\( \Phi x = Gx, \Psi x = Fx \supset Hx \))

Important note: Although terms formed using ‘\( \text{i} \)’ are grammatically terms, they do not function semantically as terms (on Russell’s account). Thus

- In specifying a model, you do not specify an interpretation for these terms.
- You cannot use these terms to get substitution instances when doing UI, EG, EI, or UG.\(^3\)
- In particular, you cannot instantiate ‘\( \forall x(x = x) \)’ with ‘\( \text{i}xFx \)’ to get ‘\( \forall x Fx = \text{i}xFx \)’. You’d better not be able to, because ‘\( \forall x Fx = \text{i}xFx \)’ implies ‘\( \exists x Fx \)’. So you’d be able to prove the existence of an \( F \) for any \( F \)!

(Contrary to what you may be thinking, this is not a good thing.)

Exercises:

1. How would you express the following sentences in logical notation? Do it first using the generalized quantifier ‘the’, and then using the Russellian ‘\( \text{i} \)’ operator.
   (a) ‘The man who killed Kennedy is a murderer.’ (b) ‘The shortest spy is the tallest pilot.’ (c) ‘Not every woman likes her father.’

2. Give a deduction of ‘\( \exists x Fx \)’ from ‘\( \langle \text{i}xFx \rangle (Fx \supset Hx) \)’.

3. Show that Substitution of Identicals holds when one term has the form ‘\( \text{i}xFx \)’, not just when both terms are individual constants. That is, give a deduction that shows the validity of the following:

   \[ a = \text{i}xFx, Ga, / G\text{i}xFx \]

\(^3\)This may seem too restrictive. After all, if we have ‘the farthest star is a gas giant’, can’t we conclude ‘something is a gas giant’? Yes—but you can get this conclusion even with the restrictive rules we have. (Convince yourself of this.)