

The Generation of Noun Phrases: Adequacy and Anaphora

Kjell Johan Sæbø
University of Oslo

Abstract

This paper investigates the semantics of anaphora in the generative perspective, centering on the so-called Principle of Adequacy for the generation of referential noun phrases, which states that the NP should be sufficient to identify the intended referent. I present an algorithm for generating discourses with definite and indefinite NPs, including definite descriptions containing pronouns, from discourse representations. In this framework, I develop a formal definition of PA and show how it must be modified for the discourse to preserve the truth conditions of the representation. The requirement that the NP identify the familiar referent is too strong and must be replaced by the requirement that it distinguish it from every other distinguishable referent. And, adequacy cannot be stated in terms of an NP alone but only in terms of the tuple of NPs at top level in a sentence.

Introduction

When generating NPs in a dynamic environment, where what is generated depends on what has been generated, it is often necessary - for the discourse to preserve the truth conditions of the discourse representation, so that it is possible to revert to it - to use definite NPs. In fact, whenever a referent occurs twice in the representation, the second NP ought to express coreference with the first. Robert Dale (1988: 161) formulates three principles for the problem of generating referential noun phrases, the three «Principles of Reference»:

1. The *principle of sensitivity* states that the referring expression chosen should take account of the hearer's knowledge.
2. The *principle of adequacy* states that the referring expression chosen should be sufficient to identify the intended referent.
3. The *principle of efficiency* states that the referring expression chosen should provide no more information than is necessary for the identification of the intended referent.

One could say that Adequacy concerns truth conditions while Sensitivity concerns presuppositions. I concentrate on the Principle of Adequacy (PA). Appelt (1985), Novak (1988), and Dale (1988) represent attempts at satisfying this requirement. As yet, however, there is no treatment of definite descriptions containing pronouns, and there is no satisfactory formal definition of the principle.

A formal definition must be set in a formal framework. To this end, I have written a program that generates discourses with definite or indefinite noun phrases from discourse representations. The greater framework is Discourse Representation Theory. I presuppose an analysis of definite and indefinite noun phrases as in Kamp (1981) or Heim (1982). While there are several implementations of anaphora interpretation in this framework - eg. Johnson / Klein (1986), Johnson / Kay (1990) - treatments of anaphora generation are scarce.

My aim is twofold: To construct an algorithm which demonstrates the possibility of treating noun phrases as parts of other noun phrases, ie. in relative clauses, and to use such an algorithm in an effort to provide a satisfactory formal definition of the Principle of Adequacy. I suggest a succession of definitions of this principle on the basis of the generation algorithm, and I stress the importance of what I call 'semi-definite' descriptions in noun phrase generation.

A Survey of the System

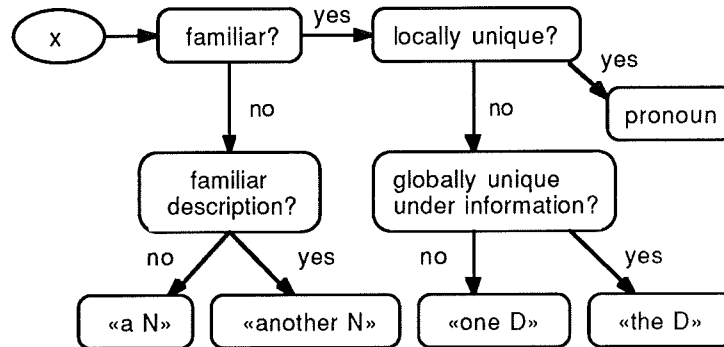
0 Overview

The algorithm is based on a simplistic semantic representation, corresponding to discourse representation structures without quantifiers. For example, the following results in the discourse «A farmer owns a donkey. He beats it.»:

[farmer(x),donkey(y),owns(x,y),beats(x,y)].

If the referent to be «covered» by the noun phrase under construction is familiar, not from the local but from the global history of the discourse, an attempt is made at generating a definite description. In comparison with a pronoun, the definite description adds information to guide the identification. The test is whether the referent is unique under the description: There is no other familiar referent with the same properties. Information may have to be added in many stages, and the preliminary description is step by step tested for uniqueness, until all accessible information on the referent is consumed. Should uniqueness fail ultimately, the definite determiner «the» yields to the semi-definite determiner «one», signalling familiarity in spite of non-uniqueness.

The overall noun phrase generation algorithm looks as follows.



1 The Representation

The algorithm is based on a conceivably simple semantic representation language. It contains discourse referents, unary and binary relation symbols, and commas for conjunctions. Unary relations are always nouns, and one referent is never ascribed two different unary relations. The representation has two limiting characteristics: It is scopeless, and, it is omniscient. Scopeless means that there are no quantifiers with scope. The only «quantifiers» are implicit existential quantifiers which bind globally in the sense that variables with the same name are understood to corefer. This corresponds to a discourse representation structure (DRS) without quantifiers. But unlike a DRS, a structure of my representation does not present a «cast» of discourse referents at the top, and it is in linear list format. For example, the following corresponds to the discourse «A farmer owns a donkey. He beats it.».

[farmer(x),donkey(y),owns(x,y),beats(x,y)].

The representation is omniscient in the following sense. Two variables which do not have the same name are understood to not corefer. This implies that a second mention of a noun corresponds to the determiner «another» instead of the indefinite article «a». The reason for this choice is that the indefinite article is not so neutral after all in the generative perspective. When unfamiliar referents are to be generated in the form of noun phrases, an indefinite description is frequently problematic if the property is familiar. I return to this point in section 4.

Lexical relations like hyponymy are ignored, so if we have man(x) and farmer(y), «the man» is to refer unambiguously to x. The generation algorithm is deterministic in the sense that a representation structure gives rise to just one discourse.

2 *The Scheme*

The aim of the generation of surface discourse from representation structure is to retain the truth conditions of the structure in the discourse. This is no trivial affair. It depends on the recognition of familiar entities and their proper characterisations. When a referent is recognised as familiar, ie. already processed, we must attempt to identify it with its antecedent in the form of a definite noun phrase. A definite noun phrase is at its least a pronoun in the sense that pronouns carry the least information to guide the identification. Next comes a definite description, which may have to be more or less complex. At the bottom of this definiteness hierarchy comes what I propose to term a semi-definite description. This is the description we use when a referent is familiar but identification fails because the information on it applies to another familiar referent as well. The common locution for this is the determiner «one of the» combined with a plural N.

Let me present the superordinate loop rule of the Prolog generation program:

```
generate(F,R,G,H) :-  member(P,F), P=..(V,X,Y),  
                      n1(F,G,H,K,X,P1,N1), n2(F,G,H,K,Y,P2,N2),  
                      append(N1,VP,S), append((V),N2,VP),  
                      append(S,(new||R1),R), append((P2,P1,P),G,G1),  
                      minus(F,(P,P1,P2),F1),  
                      generate(F1,R1,G1,(P2,P1,P)).
```

The variable F stands for the logical form from which to generate the discourse which the variable R stands for. G represents the global and H the local history of the current stage of the discourse. It is necessary to keep a record of several levels of context in order to characterise different context-dependencies: Basically, a pronoun is locally «bound» while the global context is what governs the distribution of definite, semi-definite, and indefinite descriptions. The variable K in the n1 and n2 predicates represents a local context for pronoun «binding» inside the sentence - the sentence-internal history.

The first task of the generator is to locate a binary relation in the representation list. The relation P consists of a relation symbol V, the referent X - the subject -, and the referent Y - the object. The two predicates n1 and n2 build the corresponding NPs with the aid of the unary relation symbols P1 and P2, describing the two referents. Then the three relations P, P1, and P2 are subtracted from the representation list, the sentence constructed is appended to the discourse to be generated, the global history grows by the three relations, and the local history is replaced by them.

Then the generator starts over again. The chief interest lies in the construction of the noun phrases.

3 *Familiar or Unfamiliar?*

When the subject NP predicate n1 and the object NP predicate n2 are to construct NPs in their N1 and N2 variable places to describe the parameters X and Y in the relation P, they run through four options: The NP is to be

- a pronoun
- a «the» or a «one» (semi-) definite description
- an indefinite «a» description
- an «another» description.

The first two options have in common that the referent is familiar: The history (a list of relations like the representation itself) contains a unary relation with the referent. This means that the same entity has been mentioned at least once before.

The pronoun requires the referent to be in the local history. The next option is only open if the first fails and requires the referent to be in the global history. The last two options only apply if the second option fails too - the referent must be novel. The «a» option requires the description of it to be novel as well, else the «another» option succeeds.

4 *Indefinite Descriptions: «A» and «another»*

I start the description of the particular options by describing the indefinites, then treat the definite or semi-definite descriptions and only at last turn to the pronouns. The distinction between the «a» and the «another» description deserves comment. It involves a strict condition on the indefinite article: The representation

[farmer(x),donkey(y),farmer(z),donkey(u),owns(x,y),owns(z,u)]

yields the text «A farmer owns a donkey. Another farmer owns another donkey.» In general, whenever a referent is unfamiliar and has a familiar noun description, only the «another» option succeeds. This implies an «omniscient» interpretation of the logical form: The case where a referent may or may not corefer with another never occurs (assuming that two referents with different descriptions do not corefer either). To be sure, this is an oversimplification. In reality, we can often interpret two occurrences of an indefinite in the same discourse in the sense that the two referents may but must not corefer. Thus the following is no worse than the above:

A farmer owns a donkey. Another farmer owns a donkey too.

These cases, where we are only interested in the type and not in the token, so to speak, are the contexts where «one» anaphora are used.

A farmer owns a donkey. Another farmer owns one too.

However, the alternative to the method I use is to only use the determiner «another» if a condition like $\neq(x,z)$ is stated in the representation, and this leads to texts like

? *A farmer owns a donkey. A farmer owns a horse (too).*

Note that a «one» anaphor is much more natural in object than in subject position.

? *A German sub has sunk a British ship. One has sunk a Norwegian ship too.*
A German sub has sunk a British ship. A Norwegian ship has been sunk by one too.

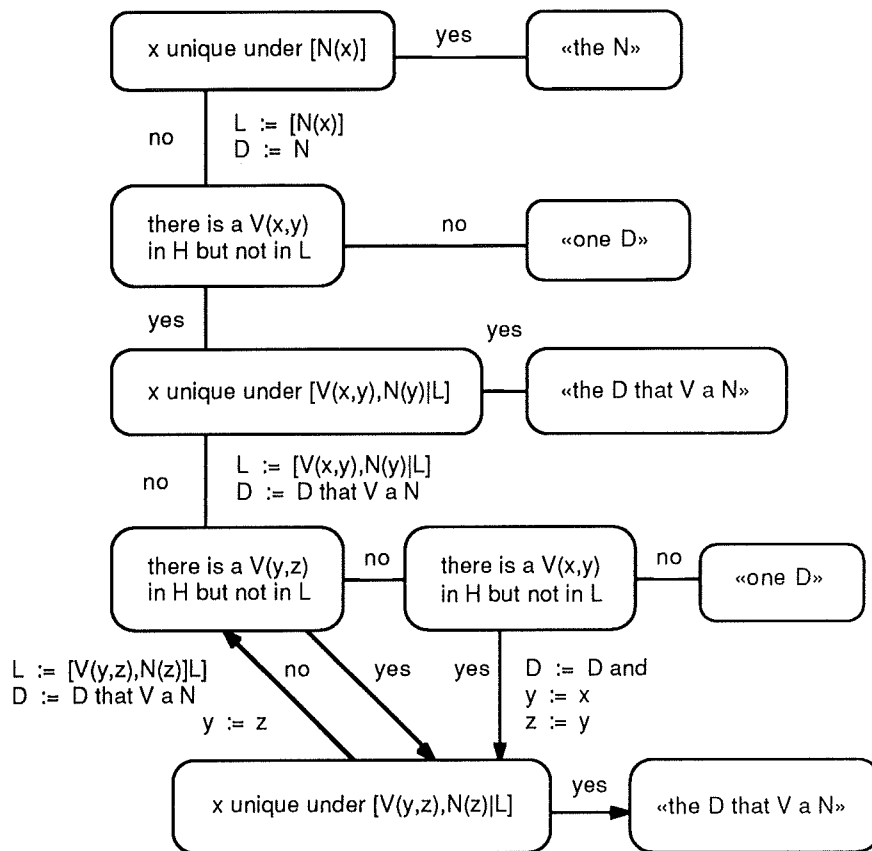
Surely, the felicity of one indefinite description in the neighbourhood of another depends on topic-focus phenomena. I shall not pursue the matter any further, only note that this question only arises when indefinite descriptions are generated and not just interpreted.

5 *Semi-Definite and Definite Descriptions*

If the parameter to be «covered» by the NP under construction is familiar, not from the local but from the global history of the discourse, the system tries to generate a definite description. A «the» description serves the purpose of identifying the parameter in cases where a pronoun would not succeed in identifying it. In comparison with a pronoun, the definite description adds information to guide the identification. The addition of a noun description may not suffice. The test is whether the familiar parameter is unique under the description: There is no other referent with the same description in the history. Information may have to be added in many stages, and the preliminary description is step by step tested for uniqueness, until all accessible information on the parameter is consumed. Should uniqueness fail ultimately, the definite determiner «the» yields to the semi-definite determiner «one».

The definiteness algorithm is the core of the system, composed from complicated components. While on the whole, sentences are constructed as terse as possible, and in particular, indefinite descriptions contain no relative clauses, noun phrases that signal familiarity and serve to identify parameters as closely as possible may have to contain relative clauses both coordinated and subordinated. In addition, they may have to contain pronouns, pronouns «bound» both outside and inside themselves, and they must provide suitable contexts for pronouns in the next NP.

The overall algorithm can be depicted in rough outline in the following way. Consider the parameter x found to be familiar not from the local but from the global history.



The overall (semi-)definite description algorithm.

Uniqueness under a list of formulae is a complicated notion that takes into account sloppy and strict identity. The subalgorithm is described in the next section.

Note that definite descriptions cannot contain other definite descriptions. However, they can contain pronouns, a fact which complicates the picture. First, pronouns can replace «a N» in the description, and second, if they are bound externally, they act as constants in the uniqueness algorithm described in the next section.

6 Pronouns in Descriptions I

When a description contains a pronoun, there are two distinct cases:

- (1) The pronoun refers inside the description, ie. it acts as a bound variable;
- (2) the pronoun refers outside the description, ie. it acts as a free variable.

(1) and (2) have different effects on the definiteness of the description. Specifically, in the uniqueness algorithm, free-variable pronouns cause the initial pair $\langle y_1, y_2 \rangle$ to be not anonymous but instantiated to one and the same referent in both places (see below). This section treats case (1), and the next section treats case (2).

In natural language, indefinite descriptions contained in definite descriptions are read «sloppily» in the following sense. For an x_1 to be unique under a description «N that V a N*», corresponding to the list $[N(x_1), V(x_1, y), N^*(y)]$, the condition is that there is no x_2 such that the history contains $N(x_2)$, $V(x_2, z)$, and $N^*(z)$ for a z ; it is not that there is no x_2 such that the history contains $N(x_2)$, $V(x_2, y)$ and $N^*(y)$. An alternative formulation of this is that the indefinite cannot outscope the definite. Moreover, if a definite description contains a pronoun «bound by» a referent of that description, that pronoun is interpreted sloppily too: For an x_1 to be unique under a description «N that V an N* that V* it», corresponding to the list $[N(x_1), V(x_1, y), N^*(y), V^*(y, x_1)]$, the condition is that there is no x_2 such that the history contains $N(x_2)$, $V(x_2, z)$, $N^*(z)$, and $V^*(z, x_2)$; not that there is no x_2 such that the history contains $N(x_2)$, $V(x_2, z)$, $N^*(z)$, and $V^*(z, x_1)$. The system captures this conduct.

In sum, uniqueness of a parameter x under a list L means that there is no parameter y different from x such that there is a list L^* that comes from L by replacing x by y and every occurrence of every parameter uniformly and such that every member of L^* is a member of the history. To accommodate this notion of uniqueness in terms of uniform substitution in a list, the definition of uniqueness involves four entities - the parameter, the list, the history, and a pair of parameters - and four recursive rules. The main rule makes use of an auxiliary relation we may notate \approx .

The parameter x is unique under the list L with respect to the history H iff there is no $y \neq x$ such that $x \approx y$ with respect to L , H , and the empty parameter pair $\langle _, _ \rangle$. Two parameters x_1 and x_2 stand in the approximate equality relation \approx wrt. L , H , and an arbitrary pair $\langle y_1, y_2 \rangle$ iff for every formula P a member of L such that x_1 occurs in P there is a formula Q a member of H such that either

- (i) $x_1 = x_2$
- (ii) $P = R(x_1)$ and $Q = R(x_2)$
- (iii) $P = R(x_1, y_1)$, $!$, and $Q = R(x_2, y_2)$ and $y_1 \approx y_2$ wrt. $L \setminus P$, H , and $\langle x_1, x_2 \rangle$
- (iv) $P = R(x_1, z_1)$ and $Q = R(x_2, z_2)$ and $z_1 \approx z_2$ wrt. $L \setminus P$, H , and $\langle y_1, y_2 \rangle$
- (v) $P = R(y_1, x_1)$, $!$, and $Q = R(y_2, x_2)$ and $y_1 \approx y_2$ wrt. $L \setminus P$, H , and $\langle x_1, x_2 \rangle$
- (vi) $P = R(z_1, x_1)$ and $Q = R(z_2, x_2)$ and $z_1 \approx z_2$ wrt. $L \setminus P$, H , and $\langle y_1, y_2 \rangle$.

(The «cut» - ! - in option (iii) in this quasi-Prolog notation has the effect that if the option succeeds thus far and subsequently fails, the other options are not eligible.)

Consider a history

[farmer(x),farmer(y),donkey(z),donkey(u),owns(x,z),owns(y,u),kicks(z,x),
kicks(u,x)]

and a list

[farmer(x),owns(x,z),donkey(z),kicks(z,x)],

corresponding to the description «... farmer that owns a donkey that kicks him». «The» is warranted, in spite of there being another farmer owning a donkey kicking «him» in the strict sense. The algorithm works as follows. x is compared to y , and $x \approx y$ only if for the formula $\text{owns}(x,z)$ there is a formula $\text{owns}(y,u)$ in the history and $z \approx u$ wrt. [farmer(x),donkey(z),kicks(z,x)], the history, and $\langle x,y \rangle$, which is the case only if for the formula $\text{kicks}(z,x)$ there is a formula $\text{kicks}(u,y)$ in the history (option (iii) again), which is not the case. Option (iii) fails, and option (iv) is not eligible, because of the cut.

It is important that this (non-)uniqueness algorithm in terms of the parameter, the alternative parameter, the property list, the history, and the anonymous parameter pair works in arbitrary depth - a pronoun and its antecedent may be arbitrarily deeply embedded in the description. Consider a more complex example, the history

[farmer(x),farmer(y),horse(z),horse(u),donkey(v),donkey(w),
owns(x,z),owns(y,u),loves(z,v),loves(u,w),kicks(v,x),kicks(w,x)]

and a list [farmer(x),owns(x,z),horse(z),loves(z,v),donkey(v),kicks(v,x)], corresponding to the description «farmer that owns a horse that loves a donkey that kicks him». Again, «the» is warranted, because for x to be $\approx y$, z must be $\approx u$ wrt. $\langle x,y \rangle$, and for this to be the case, for $\text{loves}(z,v)$ there must be a formula $\text{loves}(u,w)$ in the history (option (iv)) and v must be $\approx w$ wrt. $\langle x,y \rangle$, which cannot be the case for the same reason as above (option (iii) again).

The next section describes pronouns that refer strictly, ie. to antecedents outside the description.

7 *Pronouns in Descriptions II*

According to the last section, a list of formulae [farmer(x),owns(x,z),donkey(z),kicks(z,x)] wrt. x corresponds to a description «farmer that owns a donkey that kicks him» (and «the» is chosen as the determiner in case uniqueness holds, or else «one» is selected). The pronoun «him» is chosen for the third mention of x because the noun for the first mention provides an antecedent - in terms of logical form, the unary formulae in the list underlying the description add up to a local history K for the three noun-phrase-in-description rules:

n3(⌊,H,K,F,X,P,NP,(X,X)) :- **obpron(X,P,H,K,N), obpron(X,P,H,F,N), NP=(N).**
n3(⌊,H,K,⌊,X,P,NP,(⌊,⌊)) :- **obpron(X,P,H,K,N), NP=(N).**
n3(G,⌊,⌊,⌊,X,P,NP,(⌊,⌊)) :- **m(P,G), P=..(N,X), NP=(α,N).**

Superficially, there are two options: A pronoun or an indefinite description. However, there are two distinct pronoun cases - the case where the pronoun is «bound» inside the description, as in «the farmer that owns a donkey that kicks him», or in «the farmer that owns a donkey that kicks a farmer that beats it», etc., and the case where it is «bound» outside the description, as in «the farmer that feeds it», be it in the last sentence or in the subject NP description. The second rule takes care of the former case and the first rule takes care of the latter case.

Let me explain the various variables.

H is the recent history consisting of the formulae used for the last sentence. K is the particular recent history consisting of the formulae used for the description so far, - together with the formulae used for the subject. F is in the subject case the same as H but in the object case the particular recent history consisting of the formulae used for the subject. Totally three obpron goals are required because pronouns have two sides to them: First, their distribution itself, and second, their effect on uniqueness of the description containing them: If they are bound description-internally, they act as bound variables, subject to uniform substitution; if they are bound description-externally, they act as constants, subject to no substitution.

The distribution of pronouns in object position is governed by the following algorithm. A pronoun may be used to «cover» the parameter x wrt. the two histories H and K if x is the only parameter to occur in K or if x is the only parameter in H+K (under a noun of the specific gender). Consider now the NP «the farmer that owns it». The pronoun «it» is «bound» outside the NP, either in the last sentence or, in the event of an object NP, in the subject NP. This means that the first n3 rule above succeeds via the second obpron goal, and it has consequences for the definiteness of the large NP: The last argument position of n3 is then a pair of one and the same variable, and this is the one and only case where the initial pair of parameters for the uniqueness algorithm is not empty (anonymous). It effects a strict reading of pronouns referring out of the NP. Thus if we consider a global history

[farmer(x),farmer(y),donkey(z),donkey(u),owns(x,z),owns(y,u),...],

a list [farmer(x),owns(x,z),donkey(z)], possibly corresponding to the description «farmer that owns it», and the parameter pair (z,z), we see that «the» is again warranted: For x to be ≈y with reference to the uniqueness algorithm in the last section, for the formula owns(x,z) for R(x1,y1) in option (iii) there must be a formula owns(y,z) for R(x2,y2) in the history, <y1,y2> being set to <z,z>. So x is unique under the list wrt. the excerpt of the history and the parameter pair.

8 Donkey Anaphora

Donkey sentences have been studied in depth, but mostly, the determiner of the NP containing the antecedent has been «every», «most», etc., not «(one of) the». The semantics of the sentence «the farmer that owns a donkey beats it» is unproblematic as long as the farmer that owns at least one donkey owns just one. As with current analyses of donkey sentences with other determiners, summed up by Heim (1990), however, difficult questions arise when a farmer owns two or more donkeys. It is clear that if the farmer that owns a donkey does in fact own at least two, the donkey pronoun cannot be interpreted as «every donkey he owns». The interpretation in the system is that he beats at least one of his donkeys - the «weak reading» advocated by Chierchia (1991). The program produces eg.:

The farmer that owns a donkey beats it.

The man that loves a woman that owns a dog kicks it.

The man that loves a woman beats the donkey that kicks her.

The system functions as follows. The context for the pronoun is the list of formulae used for generating the subject description. If a farmer owns two donkeys x and y , this list will contain either $\text{donkey}(x)$ or $\text{donkey}(y)$, depending on their order in the history. Thus the system will produce a pronoun *or* a (semi-)definite description. It is unsatisfactory to have one and the same state of affairs in essentially the same context described in two different ways depending on the generational history. But if we accept the «weak reading» of the pronoun, there is no harm in it. The problem arises if we require «it» to mean essentially the same, not as «one donkey he owns» but as «the donkey he owns». If the subject noun phrase is held responsible, a solution based on the Kamp (and early Heim) analysis of «every» cases suggests itself: The determiner affects every referent in the description, so that uniqueness requires there to be a unique tuple of referents. For an x to be described as «the farmer that owns a donkey», there must be a unique pair (x,y) such that $[\text{farmer}(x), \text{donkey}(y), \text{owns}(x,y)]$. However, the alternative to describing the farmer owning two donkeys as «the farmer that owns a donkey» - the semi-definite description «one farmer that owns a donkey» - is clearly misleading: It presupposes that there is another farmer that owns a donkey. In other words, the «the» determiner may like «every» be symmetric in the sense that pairs or tuples are what count, but the «one» determiner is like «most» asymmetric.

Definite descriptions can occur in «donkey pronoun» position too (Cooper 1991):

The farmer that owns a donkey and that owns a horse beats the donkey.

This is a case where the definite description is sensitive to the local history built up by the subject noun phrase, and it is accommodated in the system.

Towards a Definition of Adequacy

An adequate definition of adequacy must not require complete identification but only that the referent be distinguished from all from which it is distinguishable. Dale's original formulation (1988: 161) is

«the NP is sufficient to identify the intended referent».

Which is too strong; a referent may share its history with others, and still the noun phrase must count as adequate if only it makes it possible to restrict it to that class. This requirement is the motivation behind what I term semi-definite descriptions, indicating familiarity by means of the determiner «one (of the)». To account for cases where it is impossible to identify the referent because it is indistinguishable from some other referent, we reformulate the informal definition as

«the NP is sufficient to distinguish the intended referent from every other referent from which it is distinguishable».

Formally, the Principle of Adequacy must be defined in terms of the NP itself, the «intended referent», and the history. On the basis of the generation algorithm, it is possible to define a partial function mapping a noun phrase, a discourse referent, and a pair of histories onto a list of formulae. The function will be undefined if a pronoun has no proper antecedent in the local history or if a description cannot be matched with corresponding relations in the global history.

An NP is adequate wrt. the referent x and the two histories H and G iff it corresponds to a list L and L is such that

there is no referent $y \neq x$ such that

there is a list L^* that comes from L by replacing x by y
and every occurrence of every other referent uniformly
such that every member of L^* is a member of H

and such that

there is no list H^* that comes from H by replacing x by y
and every occurrence of every other referent uniformly
such that every member of H^* is a member of H .

In ordinary words: There is no other referent from which the description does not distinguish the referent and from which it is distinguishable; for any other referent: If x is indistinguishable from it on the description, then x is indistinguishable from it on the complete history too.

Note that if we cut the definition off after the first clause, it expresses the notion of identification (uniqueness, definiteness) implicit in Dale's original formulation.

Pronouns that act as free variables are rendered as underscored referents, referents marked as irreplaceable in L. This is akin to a method used to a similar purpose by Dalrymple et al. (1991). Thus top level pronouns are trivially adequate, since they actually identify the intended referent even if it is indistinguishable from another. Conversely, an indefinite description will correspond to an L headed by a fresh referent and thus - on a refined definition - be trivially inadequate.

Asymmetric and Symmetric Adequacy

There is an interesting distinction between symmetrical and asymmetrical adequacy, where symmetrical adequacy relies on negative information. There are cases where the information on the referent in question is properly included in the information on another referent. Suppose the history has two farmers x and y and two donkeys z and u such that x owns z and y owns u and in addition, y beats u , and suppose the formula to be generated is $\text{feeds}(x,z)$ - if x or z cannot be covered by a pronoun and if nothing more is known about x or z than mentioned, x must be covered by «one farmer that owns a donkey», which is intuitively inadequate since it describes x or y . The general situation is this. There are two referents x and y such that what is known about x is a proper subset of what is known about y , so that x is not distinguishable from x but y is distinguishable from x : There is a list L a sublist of the history such that $\text{not } y \approx x \text{ wrt. } L$ but there is no list L^* a sublist of the history such that $\text{not } x \approx y \text{ wrt. } L^*$, with reference to the uniqueness algorithm in section 6. In a situation like this, some speakers say (1) while others say (2).

- (1) *One farmer that owns a donkey - but not the one that beats it - feeds it.*
- (2) *The farmer that owns a donkey but that does not beat it feeds it.*

(2) corresponds to a closed world assumption, while (1) reflects an open world assumption. To necessitate such an option, we must make the definition symmetric, requiring not only that x be indistinguishable from y on the complete history but that y be indistinguishable from x as well. This means strengthening the definition.

Adequacy does not Apply to NPs but to NP Tuples

The availability of pronouns is likely to enhance overall adequacy. There are two ways for a pronoun to support adequacy: As a noun phrase in its own right, ie. at top level, replacing a semi-definite description, or at a lower level, as a free-variable or a bound-variable pronoun, replacing an indefinite description in a semi-definite description and rendering the description adequate, maybe definite. However, it is interesting to note that certain cases where intuitively, a pronoun is necessary for adequacy, come out as adequate without the pronoun on the definition as it stands.

Consider these two structures:

[farmer(x),farmer(y),donkey(z),donkey(u),owns(x,z),owns(y,u),kicks(u,y)].
[farmer(x),farmer(y),donkey(z),donkey(u),owns(x,z),owns(y,u),kicks(z,x)].

With pronouns, the upper structure corresponds to the discourse «A farmer owns a donkey. Another farmer owns another donkey. It kicks him.», and the lower structure corresponds to the discourse «A farmer owns a donkey. Another farmer owns another donkey. One donkey kicks the farmer that owns it.». Without pronouns, the last sentence would in both cases have to be «One donkey (that a farmer owns) kicks one farmer (that owns a donkey)».

But on the definition of adequacy as it stands, this last sentence is not inadequate in either noun phrase. As long as the adequacy criterion is limited to one NP at a time, those pronouns cannot be shown to be adequacy-necessary. Which means that the definition of adequacy is inadequate; it is not strong enough. It misses cases where pronouns create or reflect a dependency between the two referents in a formula.

- (1) *one donkey kicks one farmer that owns a donkey*
- (2) *one donkey kicks the farmer that owns it*
- (3) *it kicks him*

Intuitively, only (2) or (3) is adequate, but according to the definition as it stands, (1) is adequate too. The point is, of course, that it does not matter which donkey or which farmer you select, but it does matter which farmer you select once you have selected a donkey. By itself, either referent (the subject or the object referent) is indeed indistinguishable, but the pair of referents is distinguishable. This shows that it is impossible to define adequacy with reference to the NP in isolation. Adequacy can only be satisfactorily defined in terms of pairs (or triples etc) of NPs and pairs of parameters, corresponding to pairs of lists. The specifics of such a modification remain to be worked out.

Definites versus Indefinites in Descriptions

Interestingly, there seems to be no adequacy difference between a definite, a semi-definite, and an indefinite description embedded in a (semi-)definite description. As long as definite descriptions are interpreted with respect to the global history, there seems to be no adequacy benefit in embedding definite descriptions. It is just in case it refers locally that an embedded definite description may be necessary for adequacy.

The farmer that owns a donkey and a horse and that beats the donkey...

Even more interestingly, indefinite descriptions inside (semi-)definite descriptions do not, as usual, signal novelty; it is equally felicitous to use «a» as «the» or «one» in such contexts, where on account of the uppermost determiner every piece of information is presupposed as familiar anyhow. The novelty condition for indefinite descriptions (Heim 1982: 369) is invalid under (semi-)definite descriptions. (It is another matter that «the» is strongly preferred to signal functionality with relations like «owns», cf «the donkey that kicks ?a / the farmer that owns it».)

Furthermore, definite descriptions in definite descriptions are not governed by the normal rule either. The uniqueness condition for definite descriptions is weakened under definite descriptions. A definite description inside another definite description does not require uniqueness on itself but only on the uppermost description.

This point is made by Dale and Haddock (1991: 166): Suppose we have a history

[farmer(x),farmer(y),donkey(z),donkey(u),beats(x,z),...].

Dale and Haddock point out that it is felicitous to refer to x in a context like this as «the farmer that beats the donkey», although there is another donkey in the history. This poses a problem for compositional interpretation, on which the embedded NP would refer uniquely in isolation. The uniqueness of the referent of an embedded definite description must be relativised to the list underlying the head description, in this case, [donkey(z),beats(x,z),farmer(x)].

Conclusions

Let me sum up the main points of the second part of this paper:

- (1) A definition of adequacy must not require complete identification but only that the referent be distinguished from all from which it is distinguishable.
- (2) The distinction between symmetrical and asymmetrical adequacy deserves attention.
- (3) A definition of adequacy must require that the tuple of referents at top level in a sentence be distinguished from the tuples from which it is distinguishable; adequacy cannot be defined in terms of one NP alone.
- (4) The distinction between a definite and an indefinite description is irrelevant to adequacy and partly also to felicity when the description is embedded in a definite description.

References*

- Appelt, D.: 1985, Planning English Referring Expressions, *Artificial Intelligence* **26**, 1-33
- Chierchia, G.: 1991, Anaphora and Dynamic Logic, in Stokhof et al. (eds.), *Quantification and Anaphora I* (= DYANA deliverable R2.2A).
- Cooper, R.: 1991, Donkeys and Situations, in Stokhof et al. (eds.), *Quantification and Anaphora I* (= DYANA deliverable R2.2A).
- Dale, R.: 1988, *Generating Referring Expressions in a Domain of Objects and Processes*, University of Edinburgh Dissertation.
- Dale, R. and N. Haddock: 1991, Generating Referring Expressions Involving Relations, in *Proceedings of the 5th Conference of the European Chapter of the ACL*, Berlin, 161-166.
- Dalrymple, M., F. Pereira, and S. Shieber: 1991, Ellipsis and Higher-Order Unification, *Linguistics and Philosophy* **14**, 399-452.
- Heim, I.: 1982, *The Semantics of Definite and Indefinite Noun Phrases*, University of Massachusetts at Amherst Dissertation.
- Heim, I.: 1990, E-Type Pronouns and Donkey Anaphora, *Linguistics and Philosophy* **13**, 137-177.
- Johnson, M. and M. Kay: 1990, Semantic Abstraction and Anaphora, in *COLING 1990*, Helsinki, 17-26.
- Johnson, M. and E. Klein: 1986, Discourse, Anaphora, and Parsing, in *COLING 1986*, Bonn, 669-675.
- Kamp, H.: 1981, A Theory of Truth and Semantic Representation, in Groenendijk et al. (eds.), *Formal Methods in the Study of Language*, Amsterdam, 277-321.
- Novak, H.: 1988, Generating Referring Phrases in a Dynamic Environment, in Zock and Sabah (eds.), *Advances in Natural Language Generation*, Volume 2, London, 76-85.
- Reiter, E.: 1990, *Generating Appropriate Natural Language Object Descriptions*, Harvard University Dissertation.

* I am indebted to Helle Frisak Sem and Kjetil Strand for insightful comments and valuable suggestions.