On Identifying Sets

Matthew Stone
Department of Computer Science &
Center for Cognitive Science
Rutgers University
110 Frelinghuysen Road, Piscataway NJ 08854-8019
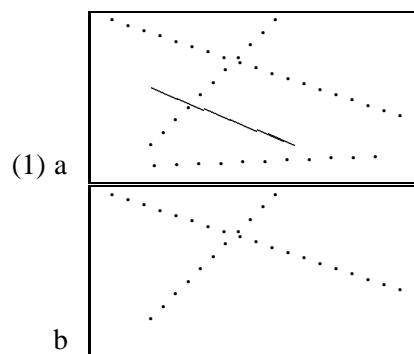mdstone@cs.rutgers.edu

Abstract

A range of research has explored the problem of generating referring expressions that uniquely identify a single entity from the shared context. But what about expressions that identify sets of entities? In this paper, I adapt recent semantic research on plural descriptions—using covers to abstract collective and distributive readings and using sets of assignments to represent dependencies among references—to describe a search problem for set-identifying expressions that largely mirrors the search problem for singular referring expressions. By structuring the search space only in terms of the words that can be added to the description, the proposal defuses potential combinatorial explosions that might otherwise arise with reference to sets.

1 Introduction

Natural language interaction lends itself to tasks like generalization, abstraction, comparison, and summarization which call for SETS of objects to be picked out using definite referring expressions.
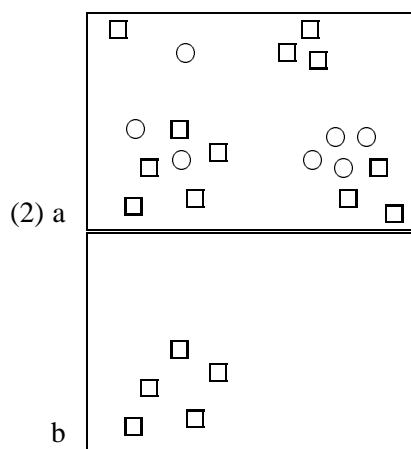
For example, consider the concrete function of describing the elements of a figure. In (1b), we find a pair of elements from (1a); imagine that some notable relationship leads to an intention to identify and comment on just THOSE elements.

(1) a

b

c   The intersecting dotted segments.

As (1c) records, these are *the intersecting dotted segments* of (1a), and can be designated as such.

Or again, we find distinguished in (2b) five elements of (2a), which might hold some independent interest. So we can and should identify these elements, and (2c), *the squares clustered at the lower left*, will do the trick.

(2) a

b

c   The squares clustered at the lower left.

Concrete problems like those of (1) and (2) cast into relief the potential difficulty of identifying sets. The world provides sets with embarrassing abundance, yet we are able to call attention to any of these sets at will, and to describe it at will both by properties its members have on their own—the members individually may be *square* or *dotted*—and by properties or relationships that the set enjoys as a collection—the set as a whole may be *intersecting*, or *clustered* in a certain region.
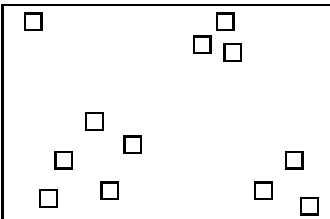
Reference to sets is more general than picturesque examples like (1) and (2) might suggest. Linguistic research suggests that covert reference to SETS OF SALIENT ALTERNATIVES plays a pervasive and fundamental role in mediating between the meanings of sentences and their interpretations in context. Consider (3), for example.
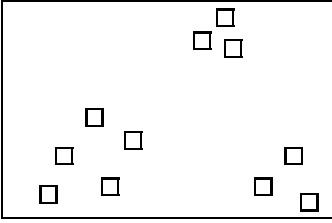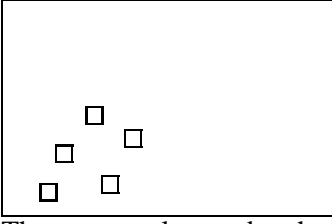
(3) a Only [Mary]$_F$ passed.
    b Well, [I]$_F$ passed.
    c Another student passed.

According to (Rooth, 1992), the interpretation of focusing adverbs such as *only* relates an instance α to a set of alternatives C: the adverb describes a property that makes α unique in C. Thus in (3a) Mary is unique among some set C of individuals in passing. Likewise, scalar implicature, as illustrated in (3b), depends on distinguishing one claim—my passing, say—as the strongest claim that can be supported among some salient set of claims C—students in the class passing, for (3b). Rooth formalizes the focus marking in these examples as contributing a presupposition that helps to identify these alternative sets. More generally, a range of lexical items, including the morpheme *other* from (3c), carry discourse presuppositions that relate their referent to salient alternatives from the context—like the students we accommodate in understanding (3c) (Bierner and Webber, 2000).

Overtly, all the examples in (3) involve singular noun phrases that specify isolated individuals. Nevertheless, representing and reasoning about reference to sets is required for faithful account of how such sentences are interpreted, and thus how such sentences can achieve the communicative goals of a system for natural language generation (NLG).

So how are expressions that refer to sets to be constructed? In this paper, I will argue that identifying sets of individuals is not as forbidding as it may at first appear. The extensive literature in NLG on singular references starting with (Dale and Haddock, 1991) tells us what to do. We must use the INTERPRETATION of provisional descriptions in context to assemble a combination of descriptive elements which identifies the intended target. Take (2), where we used the descriptive elements *square*, *clustered somewhere*, and *at the lower left*. Tracking the incremental interpretation of these descriptors should lead to a sequence like that in (4).

(4) a
    b The squares.

c
d The squares clustered somewhere.

e
f The squares clustered at the lower left.

This high-level story leaves us on familiar ground.

The project of this paper is to realize this high-level story in formal terms. I begin in Section 2 by framing the problem of singular noun phrase generation more precisely. The sequel extends this framework with a formal account of plural interpretation and generation. Section 3 introduces the two independently-motivated observations from formal semantics which form the basis of this account.

- The ASSIGNMENT-SET semantics for reference to plurals provides a way to evoke and describe collections with variables that range only over individuals (van den Berg, 1993; van den Berg, 1996). By using the assignment-set semantics, we can dispense with explicit collections in formalizing an interpretation such as that schematized in (4c); we represent only the individuals involved.

- The COVER semantics for predications about pluralities provides a simple scheme of implicit quantification to abstract collective and distributive predication (Gillon, 1987; Verkuyl and van der Does, 1991; Schwarzschild, 1994; Schwarzschild, 1996). The cover semantics offers an elegant, and convenient, definition of what it means for the set distinguished in (4c) to be characterized as *the squares clustered somewhere*.
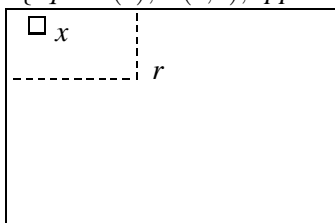
Section 4 presents the computational model of plural descriptions based on these principles. In keeping with (4), this model simply and naturally extends the models used to generate singular references. In particular, as (4) suggests, this model continues to structure the search space for generation in terms of

the words that can be added to the description and to arrive at corresponding interpretations by constraint satisfaction over individuals. In so doing, the proposal defuses the potential combinatorial explosions that might otherwise arise with reference to sets.

2 Background

At a high level, we can characterize generation procedures like that of (Dale and Haddock, 1991) or its successors as manipulating linguistic data structures that link together FORMS, MEANINGS and INTER-PRETATIONS. (5) illustrates such a data structure, as it might be entertained in identifying a uniquely identifiable element of (2a).

(5) a   F: /the square in the upper left/
    b   M: $\{square(x), in(x,r), upper\text{-}left(r)\}$

    c   I:

(5a) proposes the form *the square in the upper left*— a syntactic structure represented to some degree of abstraction. (5b) records the semantics for the description as a set of constraints—each constraint is an atomic formula with free variables that specifies the requirement that some lexical meaning contributes to the description; the variables are placeholders for the discourse entities that the description identifies. And (5c) anticipates how the hearer could process the description, by outlining the possible candidate referents for it; in (5c) we find the element of the figure which x must represent, along with the corresponding (vaguely delimited) region r in the upper left where x is located.

Any data structure linking form, meaning and interpretation combines two kinds of information. Form and meaning are related by the generator's model of linguistic resources. In the concrete case, this model is a grammar; LEXICAL semantics determines the separate constraints that can go into a description and COMPOSITIONAL semantics determines how these constraints can share variables and so describe common objects. Meaning and interpretation, meanwhile, are related by a model of the context in which the form is to be uttered. In interpreting referring expressions, we appeal to a CONTEXT SET enumerating the salient individuals at some point in the discourse and a COMMON GROUND listing the

instances of constraints that can be presumed to be mutually known at that point. To determine the interpretation from the meaning, we must instantiate the free variables to individuals in the context set and match the instantiated constraints against the elements of the common ground. In practice, constraint satisfaction heuristics (Mackworth, 1987) are required to accomplish the process of instantiation and matching with any hope of efficiency.

With an understanding of what data structures such as (5) represent and how to carry out reasoning over them, solving descriptive problems becomes a matter of search. In practice, this search is typically managed quite simply: for example, (Dale and Haddock, 1991) select transitions among states according to a greedy heuristic, while (Dale and Reiter, 1995) select alternatives by exploring different kinds of constraints in a fixed order. In any case, the search starts with a structure defining an empty description, which means nothing and could refer to anything. Structures are then extended and considered in turn until the interpretation satisfies the system's goals (for example because it allows only a specified value, the intended referent, for a particular variable). The process of extension simply consists of deriving a more elaborate form with a richer meaning using the generator's linguistic resources—it is useful to think of obtaining this by carrying out a step of derivation in a lexicalized grammar (Stone and Doran, 1997)—and then consulting the model of the context to obtain an updated interpretation.

To extend these data structures to sets, we cannot introduce set variables and maintain the alternative candidate set values those variables might ultimately refer to—for one thing, there are just too many sets to represent an interpretation this way.

3 An intuition and some semantics

Here is a suggestion: REINTERPRET data structures like (5) as compatible with descriptions of collections as well as singletons. This should have some intuitive appeal. After all, we always thought that a form like (5a) abstracted out details of syntax and morphology; there's no difficulty then in seeing it as short for a family of singular and plural expressions like */the square(s) in the upper left/*. Similarly, the interpretation is already defined in terms of a set of instances that satisfy the description; why not use this as THE set that the description refers to?

The problem is the meaning. We have to allow

for both DISTRIBUTIVE predicates, which characterize collections based on properties of the individuals involved, and COLLECTIVE predicates, which describe collections that jointly participate in some relation. If we have collective predicates, how can we get away without explicit set variables which could take on any set as a possible value?

Van den Berg's treatment of dependent plurals in dynamic semantics provides the first half of the answer (van den Berg, 1993; van den Berg, 1996). Van den Berg's starting observation is that discourse can set up and maintain dependencies between the individuals in one set and the individuals in another.

(6) a Every man loves a woman.
    b They prove this by giving them flowers.

In (6) for example, the first sentence introduces a set of men and a set of women, where each man in the one set is related to a woman in the other set (by love); the second sentence builds on that relationship, indicating another connection (of giving) between each man and the corresponding woman.

For van den Berg, data like (6) show that discourses describe sets of CASES generally. Each case involves a sequence of entities that stand in various relationships to one another, sometimes directly as individuals and sometimes indirectly through their membership in larger, related groups. Some sentences in discourse aggregate cases together, to express relationships that hold collectively among groups. Other sentences, like (6), zoom in on individual cases, and describe distributive properties which hold of isolated individuals. In zooming in on cases, rather than individuals, these sentences maintain and extend the dependencies and other relationships that define a case.

The second half of the answer derives from the observation, made in (Gillon, 1987; Verkuyl and van der Does, 1991; Schwarzschild, 1994; Schwarzschild, 1996), that the collective and distributive readings of plurals represent only the extremes in a larger space of readings. Take (7):

(7)   Rogers, Hammerstein and Hart wrote
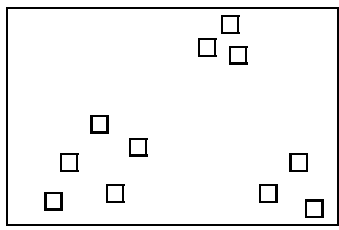   musicals.

This sentence is true, but only in virtue of the joint action of Rogers and Hammerstein in writing some musicals and the joint action of Rogers and Hart in writing other musicals. As a matter of fact, the three never wrote a musical individually or as a single

team, so both the collective and distributive readings are false.

We will follow Schwarzschild's proposal most closely. Schwarzschild argues that we establish that a linguistic predicate applies to a plural argument by recovering a salient cover of that argument from the context. A cover here means a set of pluralities whose union or sum is the overall plural argument. Given the cover, the overall plural predication holds just in case the basic property denoted by the predicate is true (collectively) of each of the sets (or CELLS) in the cover. For example, the sets consisting of Rogers and Hammerstein and of Rogers and Hart form the salient cover of Roger, Hammerstein and Hart in (7); the example is true because each of the cells in this cover directly enjoys the property of having written a musical.

Schwarzschild's covering proposal and van den Berg's assignment-set proposal are perfectly compatible. Following van den Berg, we interpret discourses in terms of sets of cases, where these cases spell out dependencies among related individuals. But now, following Schwarzschild, we zoom in on those cases flexibly, by covering them. Sometimes we consider all the cases together and describe relationships among aggregated groups; sometimes we consider cases separately and describe individuals distributively; and sometimes, as in (7), we take an intermediate step and cluster the cases into some salient subgroups.

Now let us return to (4c), repeated as (8a), and consider informally what this proposal amounts to:

(8) a
    b The squares clustered somewhere.

The assignment-set cover semantics fits the description to the figure this way. As in (5c), the figure schematizes a set of cases; here each case involves two entities, a square and the location of the cluster to which the square belongs. The description applies because we can look at the individual cases to see that we have *squares*, and we can group the cases together by region into a cover so that in each cell the squares are indeed *clustered at* the location.

At this point, some formalism is required to pro-

ceed with the development. We'll use assignment variables like $\mathbf{g}$ to range over cases; g_x is the value of $\mathbf{g}$ for variable x.[1] Interpretations are defined in terms of sets of cases, naturally; we'll use Γ to range over a set of cases and write $\Gamma(x)$ for $\{g_x : \mathbf{g} \in \Gamma\}$. Most constraints will involve several variables; we can abstract this in terms of a sequence of variables $\mathbf{x}$ and the tuple of collections that those variables take on across a set of cases, $\Gamma(\mathbf{x})$. (We can define this explicitly as $\Gamma(\mathbf{x}) = \mathbf{G}$ where $G_{x_i} = \Gamma(x_i)$.)

Now, consider an atomic constraint $F(\mathbf{x})$. In general, $F(\mathbf{x})$ will have multiple known instances, and each instance will relate collections of individuals to one another. Thus the common ground will associate $F(\mathbf{x})$ with a set of tuples of sets, which we write as $[\![F(\mathbf{x})]\!]$. An interpretation Γ will fit one of those instances directly iff $\Gamma(\mathbf{x}) \in [\![F(\mathbf{x})]\!]$. In this case we say $F(\mathbf{x})$ DESCRIBES Γ.

For example, consider the constraint $clustered(x,r)$. Let us say a set X is clustered around R if R is a singleton spatial location $\{r\}$ and X is a group of sufficient cardinality and density located together at r. Then we might find three tuples of $[\![clustered(x,r)]\!]$ in the explicit depiction of (8). If we define Γ_1 as in (9) then $clustered(x,r)$ describes Γ_1.

(9) $\{\langle x,r \rangle : x$ a square in the lower left region $r\}$

Of course, we are principally interested in the ability to zoom in to particular cases, using covers. We represent a cover using a reflexive binary relation that links each assignment to any assignment in its cell. Given such a relation C, the constraint $\copyright_C\, p$—read "covered by C, p"—says that p is true on each of the cells of the cover specified by C. We will only consider the case where p is an atomic constraint $F(\mathbf{x})$.

(10) $\copyright_C\, F(\mathbf{x})$ DESCRIBES Γ iff for all $\mathbf{g} \in \Gamma$, $F(\mathbf{x})$ describes $\{\mathbf{h} \in \Gamma : C(\mathbf{g},\mathbf{h})\}$.

Continuing from (9), define Γ_2 and Γ_3 in (11a) and (11b) respectively.

(11)a $\{\langle x,r \rangle : x$ a square in the center top $r\}$
    b $\{\langle x,r \rangle : x$ a square in the lower right $r\}$

---

[1] I adopt the notation throughout that $\mathbf{v}$ is a tuple and v_i is component i of $\mathbf{v}$, where components may be indexed equivalently by variables or numbers. Lower case Roman letters are for ordinary individuals and tuples thereof; upper case Roman letters are for sets of individuals and tuples thereof; upper case Greek letters are for sets of tuples.

Then with C defined as in (12), $\copyright_C\, clustered(x,r)$ describes $\Gamma_1 \cup \Gamma_2 \cup \Gamma_3$.

(12) $C = (\Gamma_1 \times \Gamma_1) \cup (\Gamma_2 \times \Gamma_2) \cup (\Gamma_3 \times \Gamma_3)$.

Observe the close connection between this formal judgment and the informal discussion of (8) presented earlier. We have a set of cases involving a square and the location of the cluster to which the square belongs; we cover the cases together by region and find that the resulting groups define a specified cluster at a specified location.

Schwarzschild's proposal is that the salient cover C is supplied from context. In the case of definite reference to tuples Γ, we can regard the tuples in any predicate as defining the appropriate salient cover for plural predication; any tuples that help to identify Γ must be prominent parts of the shared context. Meanings of referring expressions should therefore appeal to a condition $\copyright\, p$ which describes Γ iff there is a C for which $\copyright_C\, p$ describes Γ.

Clearly, if $\copyright\, p$ describes Γ and $\copyright\, p$ describes Γ' then $\copyright\, p$ describes $\Gamma \cup \Gamma'$. This in turn entails that any condition $\copyright\, p$ describes a maximal set of cases from the current context; the same goes for any conjunction of conditions of this form. We can treat this set of cases as the interpretation of a description. In particular, consider a description $\mathbf{L}$ that consists of a list of constraints $\copyright\, L_i(\mathbf{x})$ formulated in terms of a tuple of variables $\mathbf{x}$ and atomic conditions on those variables $L_i(\mathbf{x})$. Assume a context set D defining a domain of salient individuals, so that candidate cases to interpret $\mathbf{L}$ are given by $\Upsilon := \{\mathbf{a} : a_i \in D\}$. The development thus far leads us to define the INTERPRETATION of $\mathbf{L}$—$I(\mathbf{L})$—as:
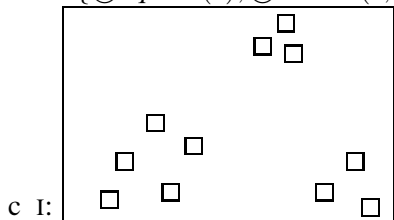
(13) $I(\mathbf{L}) := \max_{\Gamma \subseteq \Upsilon} \forall i : \copyright\, L_i(\mathbf{x})$ describes Γ

Drawing on our running discussion, we can apply this definition to the description $\mathbf{L} = \{\copyright square(x), \copyright clustered(x,r)\}$ and the context schematized by (2a). Of course, we find $I(\mathbf{L}) = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3$. The fully distributive cover shows that the *square* condition is satisfied; the cover of (12) shows that the *clustered* condition is satisfied. Meanwhile, no further cases can be considered without adding either a circle or the unclustered square.

The reader will already have recognized $I(\mathbf{L}) = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3$ as the set of cases that goes with (8a). Thus, we have reconciled the informal picture of (4) with the concrete data structures of form, meaning

and interpretation that NLG demands. For (8) we can now read (14).

(14)a F: /the squares(s) clustered __ /
b M: $\{\copyright\, square(x), \copyright\, cluster(x,r)\}$

c I:

4 Computing referring expressions

At this point, we have an understanding of what kinds of representations we can use to describe the derivation of plural referring expressions. But we still must devise appropriate reasoning methods for these representations. The problem is the subject of this section.

4.1 Collective Constraints

The first step is to formulate a constraint-satisfaction heuristic that accounts for cover-constraints on collections. In general, constraint-satisfaction heuristics provide a technique for approximating the interpretation of a description. The key notion is that of a CONSTRAINT NETWORK for a description $\mathbf{L}$, which determines a tuple $\mathbf{C}$ of CONSTRAINED VALUES. This tuple specifies a generous set of possible values C_i for each variable x_i in $\mathbf{x}$; it is obtained by conservatively eliminating values that are determined to be inconsistent with $\mathbf{L}$ according to heuristic tests. For example, the usual arc-consistency heuristic for a constraint over individuals $K(\mathbf{x})$ is to eliminate a value v for variable z unless some $\mathbf{g} \in [\![K(\mathbf{x})]\!]$ has $g_z = v$ and $g_k \in C_k$ for all k.

We will adapt this to the case of cover constraints with the following test of consistency. An individual value v for a variable x_i maintains its membership in C_i in the presence of a collective constraint $\copyright\, L_j(\mathbf{x})$ whenever v belongs to a SUBSET G_i of C_i which participates directly in the relation denoted by $L_j(\mathbf{x})$ with sets of possible values for the other variables. This criterion is spelled out formally in the definition in (15).

(15)   Value v for variable x_i is
        COVER-CONSISTENT (C-CONSISTENT)
        with constraint $\copyright\, L_j(\mathbf{x})$ under constrained
        values $\mathbf{C}$ if there is an $\mathbf{G} \in [\![L_j(\mathbf{x})]\!]$ with
        $v \in G_i$ and $G_k \subseteq C_k$ for all k.

All values of x_i in C_i that are not c-consistent with constraint $\copyright\, L_j(\mathbf{x})$ may be deleted from C_i, as they will not satisfy the constraint. Doing so makes C_i ARC c-consistent with respect to $\copyright\, L_j(\mathbf{x})$, and provides the basic step in a network-based arc-consistency constraint-satisfier. As with ordinary constraint satisfaction, we arrive at a final tuple of values for $\mathbf{x}$ by starting with an initial tuple $\mathbf{C}_0$ of values—often an assignment $\mathbf{D_X}$ giving each variable D—and a queue of arcs linking each C_i with any $\copyright\, L_j(\mathbf{x})$ that constrains it. Until the queue is empty, we select an arc and enforce the arc c-consistency by pruning C_i; if C_i changes we requeue all arcs that might no longer be arc c-consistent after the deletion. I will refer to the final tuple of values as $P(\mathbf{L}; \mathbf{C}_0)$, for the PLURAL constraint network on description $\mathbf{L}$ and domains $\mathbf{C}_0$.

The properties of this algorithm are in line with ordinary constraint satisfaction. The output will not provide all and only solutions to the constraints without further assumptions about the constraints. However, we can show, as usual, that the network converges on consistent values for variables in the ordinary linguistic case where the constraint graph is a tree—a semantic property, that there are no disjoint sets of constraints that overlap on the same two variables, that follows under plausible assumptions about the derivation of semantics from a tree-structured syntax. We can show further that these values, together with the tuples in $[\![L_i(\mathbf{x})]\!]$ that cover these values, determine precisely the collection of assignments $I(\mathbf{L})$.

4.2 Search for Referring Expressions

The second step is to formalize the task of constructing a description as a state-space search task. Suppressing details of form for exposition, each state is a tuple Σ as set out in (16).

(16) $\Sigma = \langle \mathbf{L}, \mathbf{r}, \mathbf{R}, \mathbf{x}, P(\mathbf{L}; \mathbf{R}, \mathbf{D_X}), P(\mathbf{L}; \mathbf{D_r}, \mathbf{D_X}) \rangle$

The state represents:

(17)a a description $\mathbf{L}$;
b a tuple $\mathbf{r}$ of distinguished free variables in the description for which we must identify specific intended values;
c a tuple $\mathbf{R}$ of sets describing the value R_i which we intend for the corresponding variable r_i;
d the remaining free variables of the description $\mathbf{x}$;

e a constraint network $P(\mathbf{L}; \mathbf{R}, \mathbf{D_X})$ describing the values for all the free variables in the description, on the assumption that the distinguished variables take on the values we intend; and

f a constraint network $P(\mathbf{L}; \mathbf{D_r}, \mathbf{D_X})$ describing the values for all the free variables in the description, on the assumption that the distinguished variables may, like other variables, take on any values from the context set.

The distinction between the variables whose intended reference is fixed and those for which it is derived as a byproduct of the search process is due to Horacek (Horacek, 1995; Horacek, 1996); the distinction derives increased importance when relating one collection to another as the choice of collections need not give rise to explicit branching in search.

The initial state involves an empty description and so has the form given in (18).

(18) $\Sigma = \langle \oslash, \mathbf{r}, \mathbf{R}, \langle \rangle, P(\oslash; \mathbf{R}), P(\oslash; \mathbf{D_r}) \rangle$

A state such as (16) represents a final state that successfully resolves the generation task when each variable x from $\mathbf{r}$ and $\mathbf{x}$ is associated with the same set C_x in both $P(\mathbf{L}; \mathbf{R}, \mathbf{D_X})$ and $P(\mathbf{L}; \mathbf{D_r}, \mathbf{D_X})$. This simply means that the hearer's interpretation of the referring expression matches the speaker's intended interpretation.
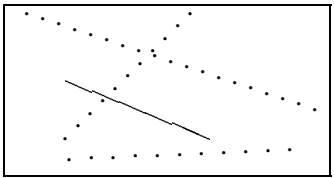
At any state Σ, the grammar defines a set of constraints of the form $\copyright L(\mathbf{rx}; \mathbf{y})$ that could potentially be added to the description to obtain $\mathbf{L}'$—L is some domain relation, $\mathbf{r}$ and $\mathbf{x}$ name the old variables from $\mathbf{L}$ while $\mathbf{y}$ names fresh variables. Of course, we want to restrict our attention to constraints that are compatible with our intended interpretation. To achieve this restriction, we begin by computing the new constraint network $C' = P(\mathbf{L}'; \mathbf{R}, \mathbf{D_{xy}})$. We check, whenever $\mathbf{R}$ assigns a value to x, that $R_x \subseteq C'_x$. If this test admits the new constraint, the new state obtained from state Σ is computed as in (19).

(19) $\langle \mathbf{L}', \mathbf{r}, \mathbf{R}, \mathbf{xy}, P(\mathbf{L}'; \mathbf{R}, \mathbf{D_{xy}}), P(\mathbf{L}'; \mathbf{D_r}, \mathbf{D_{xy}}) \rangle$

4.3   An Example

I return to (1) to provide an illustration of the final scheme; the goal is to identify the segments in (20f), R, from among those in (20a). I use figures and references to figures, in place of constraint networks; the description uses the variable r. The states proceed, perhaps, thus:
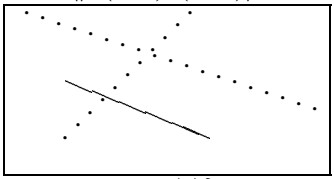
(20)a

b $\langle \oslash, r, R, \langle \rangle, (20a), (20f) \rangle$
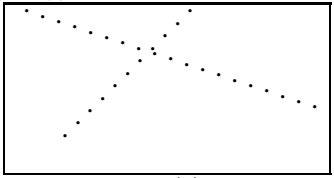c $\langle \{\copyright segment(r)\}, r, R, \langle \rangle, (20a), (20f) \rangle$

d

e $\langle \{\copyright segment(r), \copyright intersecting(r)\},$
   $r, R, \langle \rangle, (20d), (20f) \rangle$

f

g $\langle \{\copyright segment(r), \copyright intersecting(r),$
   $\copyright dotted(r)\}, r, R, \langle \rangle, (20f), (20f) \rangle$

5   Closing thoughts

Descriptions of sets obviously have much in common with expressions that describe a single entity from the shared context. In particular, adopting the standard view of NLG as goal-directed activity (Appelt, 1985; Dale, 1992; Moore, 1994; Moore and Paris, 1993), singular and plural descriptions agree both in the kinds of intentions that they can achieve and the stages of generation at which they can be formulated. We cannot expect a single process to be responsible for set descriptions across all intentions or stages of NLG.

For example, as with a singular description, a description of a set may appeal to properties that play a role in the argument the speaker is trying to make, and may therefore address goals above and beyond simple identification of discourse entities. (See (Donellan, 1966; Kronfeld, 1986) on the distinction.) (Green et al., 1998a; Green et al., 1998b) show how such descriptions may be represented and formulated in NLG at a high-level process of content or rhetorical planning. At the same time, plurals and singulars are alike in offering resources for reference—such as pronouns, *one*-anaphora or aggregated expressions—that bypass explicit description altogether. The use of these resources may be quite closely dependent on the surface form being generated and so could reflect a relatively late decision in the generation process (Dale and Haddock,

1991; Reiter, 1994; Dalianis, 1996).

These complexities notwithstanding, we can expect many descriptions of sets, like descriptions of individuals, to be formulated from scratch to achieve purely referential goals during the SENTENCE PLANNING phase of NLG, between content planning and surface realization (Rambow and Korelsky, 1992; Reiter, 1994). I have shown that using covers to abstract collective and distributive readings—and using sets of assignments to represent plural references—yields a search space for this problem which largely mirrors that for singulars, and which avoids computation and search over sets of collections. Although sets proliferate explosively, it is no surprise that the search space for plurals set up by (19) is, like that for singulars, ultimately defined by the sequences of elements that make up descriptions. NLG involves search to use words effectively—choices of words should be the only decisions a referring expression generation system has to make.

Acknowledgments

References

Douglas Appelt. 1985. *Planning English Sentences*. Cambridge University Press, Cambridge England.

Gann Bierner and Bonnie Webber. 2000. Inference through alternative-set semantics. *Journal of Language and Computation*.

Robert Dale and Nicholas Haddock. 1991. Content determination in the generation of referring expressions. *Computational Intelligence*, 7(4):252–265.

Robert Dale and Ehud Reiter. 1995. Computational interpretations of the Gricean maxims in the generation of referring expressions. *Cognitive Science*, 18:233–263.

Robert Dale. 1992. *Generating Referring Expressions: Constructing Descriptions in a Domain of Objects and Processes*. MIT Press, Cambridge MA.

Hercules Dalianis. 1996. *Concise Natural Language Generation from Formal Specifications*. Ph.D. thesis, Royal Institute of Technology, Stockholm.

K. Donellan. 1966. Reference and definite description. *Philosophical Review*, 75:281–304.

Brendan Gillon. 1987. The readings of plural noun phrases in english. *Linguistics and Philosophy*, 10(2):199–299.

Nancy Green, Giuseppe Carenini, Stephan Kerpedjiev, Steven Roth, and Johanna Moore. 1998a. A media-independent content language for integrated text and graphics generation. In *CVIR '98 – Workshop on Content Visualization and Intermedia Representations*.

Nancy Green, Giuseppe Carenini, and Johanna Moore. 1998b. A principled representation of attributive descriptions for generating integrated text and information graphics presentations. In *Proceedings of International Natural Language Generation Workshop*, pages 18–27.

Helmut Horacek. 1995. More on generating referring expressions. In *Proceedings of the Fifth European Workshop on Natural Language Generation*, pages 43–58, Leiden.

Helmut Horacek. 1996. A new algorithm for generating referring expressions. In *ECAI 8*, pages 577–581.

Amichai Kronfeld. 1986. Donellan's distinction and a computational model of reference. In *Proceedings of ACL*, pages 186–191.

Alan Mackworth. 1987. Constraint Satisfaction. In S.C. Shapiro, editor, *Encyclopedia of Artificial Intelligence*, pages 205–211. John Wiley and Sons.

Johanna D. Moore and Cécile L. Paris. 1993. Planning text for advisory dialogues: capturing intentional and rhetorical information. *Computational Linguistics*, 19(4):651–695.

Johanna Moore. 1994. *Participating in Explanatory Dialogues*. MIT Press, Cambridge MA.

Owen Rambow and Tanya Korelsky. 1992. Applied text generation. In *ANLP*, pages 40–47.

Ehud Reiter. 1994. Has a consensus NL generation architecture appeared, and is it psycholinguistically plausible? In *Seventh International Workshop on Natural Language Generation*, pages 163–170, June.

Mats Rooth. 1992. A theory of focus interpretation. *Natural Language Semantics*, 1(1):75–116.

Roger Schwarzschild. 1994. Plurals, presuppositions, and the sources of distributivity. *Natural Language Semantics*, 2:201–248.

Roger Schwarzschild. 1996. *Pluralities*. Kluwer, Dordrecht.

Matthew Stone and Christine Doran. 1997. Sentence planning as description using tree-adjoining grammar. In *Proceedings of ACL*, pages 198–205.

M. H. van den Berg. 1993. Full dynamic plural logic. In K. Bimbó and A. Máté, editors, *Proceedings of the Fourth Symposium on Logic and Language*, Budapest.

M. H. van den Berg. 1996. Generalized dynamic quantifiers. In J. van der Does and J. van Eijk, editors, *Quantifiers, Logic and Language*. CSLI.

Henk Verkuyl and Jaap van der Does. 1991. The semantics of plural noun phrases. Preprint, ITLI, Amsterdam.