

A New Predictive Theory of Presupposition Projection

Benjamin R. George
University of California, Los Angeles

1. New and Revived Issues in Projection Theory

This paper will present a new approach¹ to the projection problem for presuppositions with a number of distinctive predictions and theoretical properties. Although the new approach is interesting as an exercise in providing a purely semantic account of presupposition projection that does not make use of dynamic machinery (compare Peters 1977), its primary interest is in the methodological and empirical characteristics that distinguish it from most other theories.

1.1. Overgeneration and Predictive Theories

Many theories of presupposition projection, including most in the popular dynamic semantics tradition (exemplified by Heim 1983), encode the projection behavior of a quantifier or connective in its lexical entry. While a dynamic meaning has no separate component that describes a projection property, dynamic theories do overgenerate dynamic meanings in the sense that, from the basic set-theoretic components regularly used to state dynamic meanings, we could build for any classical truth function or quantifier many different dynamic meanings that would agree with the classical function for static, bivalent, but would produce vastly different patterns of projection behavior. This overgeneration is a form of predictive and explanatory weakness, and it fails to account for the apparent non-arbitrariness of pairings between normal truth-conditional content and projection behavior.²

The issue of overgeneration or lack of predictiveness has been known for some time (see, e.g., Heim 1990, Soames 1989), and although it is most often brought up in discussions of dynamic semantics, it is relevant to many other theories of projection as well. These concerns are not new, but attention to the development of predictive theories has increased in recent years, with a number of re-

The research leading to this paper has benefited immeasurably from the communications with Philippe Schlenker, Ed Keenan, all the participants in the Fall 2007 UCLA presupposition seminar, Emmanuel Chemla, Roni Katzir, Giorgio Magri, Danny Fox, Ed Stabler, Daniel Büring, Marcus Kracht, and others. My work on presupposition projection has been supported in part by Graduate Research Mentorship and a Graduate Summer Research Mentorship provided by the UCLA Graduate Division and by NSF Grant BCS-0617316 (under the supervision of Philippe Schlenker).

¹This approach is one of a family of approaches discussed in my M.A. thesis (George 2008b), to which the reader may should refer for further discussion of the empirical issues involved and a more rigorous formalization of the system.

²Space constraints prevent a more complete treatment of these issues, but many of the references (see especially Schlenker 2008a, 2006, Heim 1990, Soames 1989, LaCasse 2008, George 2008b) provide additional discussion.

cent pragmatic proposals (such as transparency theory (Schlenker 2006, 2008a) and new attempts to unite the study of presuppositions and scalar implicatures (Chemla 2008)) and predictive variations on dynamic semantics (such as local context theory (Schlenker 2008b) and dynamic semantics with a constrained lexicon (LaCasse 2008)).

The predictive theory presented in this paper differs from those mentioned above in that it maintains a static semantics, but analyzes projection in semantic, rather than pragmatic, terms.

1.2. Empirical Issues

Although a more constraining theory may be methodologically desirable, empirical adequacy is, obviously, also quite important, so I want to consider some natural language examples where my judgments are not in keeping with the predictions of many popular theories. Such disparity is of interest to the study of projection in general, but bad predictions are especially problematic for a predictive theory. A non-predictive projection theory can often address the problematic predictions of the projection behavior of a connective or quantifier by changing its lexical entry, but in a predictive theory such a response will be difficult or impossible: incorrect predictions will usually require us to revise the architecture of the whole theory, not just to make minor lexical adjustments.

1.3. Differences Between Quantifiers

Many theories of projection, whatever their theoretical orientation, attribute universal or near-universal presuppositions to all cases of quantification over a presuppositional predicate. This seems appropriate for certain quantifiers, since, for example, it is plausible that (3) should be a presupposition of (1) and (2):

- (1) Each of these ten students has stopped smoking.
- (2) None of these ten students has stopped smoking.
- (3) Each of these ten students has smoked in the past.

However, although theories of projection often treat all quantifiers as the same or nearly the same in their projection behavior, many quantifiers with a less universal flavor than those in (1) and (2) don't seem to be associated with strong universal inferences. In particular, it is not at all clear that we should attribute a universal presupposition to (4), and (5) seems quite true and appropriate in a world where exactly three students are former smokers, and the rest have never smoked:

- (4) At least one of these ten students has stopped smoking.
- (5) Exactly three of these ten students have stopped smoking.

If my claims above are right,³ and (4) and (5) neither entail nor presuppose (3), we will need a theory that attributes different projection behavior to different quantifiers. For a predictive theory, we will need some account of how projection and truth-conditional content are associated, but the need to at least superficially account of the disparity between quantifiers is independent of the issue of predictiveness, and this need is unmet by many popular theories of projection.

1.4. Restrictors

The projection of presuppositions from restrictors is poorly understood, but many theories make the resulting presuppositions implausibly strong. Consider the following example:

(6) Every logician who dislikes her employer drinks tea.

Although some theories would predict that (6) should presuppose that every logician is employed, my intuition is that (6) can be quite true in a world where there are unemployed logicians. Indeed, if the only employed entity in the world is a logician, and there are thousands of unemployed logicians, but the one employed logician hates her employer and drinks tea, (6) seems quite true.

2. Description of the Theory

Although the analysis presented here is not dynamic, it is trivalent, with a third truth value - written # - encoding presupposition failure. The presuppositions of a sentence are just the proposition picking out the cases where its truth value is not #. The theory describes a procedure for combining trivalent functions, and projection predictions are identified by seeing how this procedure allows or doesn't allow presupposition failure in one constituent to cause presupposition failure for the whole sentence. That presuppositions place requirements on common ground is presumed to be the result of a conversational rule that renders it uncooperative to utter a sentence to which any participant might assign the truth value #.⁴

2.1. Function Deployment

The basic extensions of linguistic expressions will, as usual, be functions, although here we'll want to work with functions of multiple arguments in their un-Curried

³Although I am reasonably confident of my judgments, especially for the *exactly* cases, it cannot be overemphasized that there is a considerable disagreement among both linguists and naïve speakers about which inferences are warranted with such examples; recent experimental work (Chemla 2007) calls attention to the high degree of uncertainty on this point.

⁴This is, of course, not a complete account of the discourse act of presupposing. Given the some presupposition triggers, and a system of function combination that explains how triggers attach semantic presuppositions to sentences, we can choose any of a number of discourse theories to relate semantic presuppositions of sentences to various conversational acts.

forms. Functions will be lexically defined only on a domain of bivalent, non-presuppositional arguments, so ordinary function application will not be able to combine a function with a presuppositional argument. Instead, the formalism presented here is based on an alternative notion of function-argument combination - function deployment - to handle presuppositional arguments.

Function deployment will be built from function application using the notion of a repair: every argument x is associated with a family of repairs that depends on x , the extension of the function of which x is an argument, and the arguments that precede x in linear order. The repairs for any x preserve the bivalent part of x , differing from x only in that they substitute some bivalent content for its presuppositional part. For this reason, if x is entirely bivalent, the only repair of x will be x itself, but if x is presuppositional in any way then x will not be a repair of itself. Putting aside for the moment how we compute repairs, deployment is defined as follows:

Definition 1 (Function Deployment) *For an n -ary function f and a list x_1, \dots, x_n of arguments for f , if there is b such that for any y_1, \dots, y_n and z_1, \dots, z_n that are acceptable lists of repairs for x_1, \dots, x_n , $f(y_1, \dots, y_n) = f(z_1, \dots, z_n) = b$, then f deployed on x_1, \dots, x_n , written $f[x_1, \dots, x_n]$, is equal to b . Otherwise, $f[x_1, \dots, x_n] = \#$.*

The intuition underlying this is that the deployment of f on a list of arguments is the value that would result from applying f to any repairs of its arguments, and if there is not a unique consensus value (because different repairs produce different outcomes), then we have presupposition failure.

The results of function deployment will depend on what the repairs of the arguments are in the context at hand. The general definition of repair sets appears in Section 2.2, but things are simple in the truth-functional case: the only repair of 0 is 0, the only repair of 1 is 1, and both 0 and 1 are repairs of $\#$. In the case of truth values, we can understand $\#$ as representing uncertainty between 0 and 1.

Knowing these repairs, we can compute the deployment effects and projection facts for the various truth functions. The resulting trivalent system is just that of the strong Kleene trivalent logic (Kleene 1952), with symmetrical projection behaviors that agree with standard dynamic accounts for presuppositions in the latter argument of a truth function, as can be seen in the linguistic examples below.

2.1.1. Deployment of Negation

To see how this works for negation, consider the following pair of sentences:

- (7) Smith knows that he is incompetent.
- (8) Smith doesn't know that he is incompetent.

The truth value of (7) is 1 if Smith is incompetent and knows it, 0 if Smith is incompetent but doesn't know it, and $\#$ if Smith is competent. To compute the presuppositions of (8), one deploys the usual (bivalent) negation function on the

truth value of (7). If (7) has truth value 0, then 0 is the only repair for this truth value, so the truth value of (8) is just $\neg 0$, which is to say 1. Likewise, if (7) is true, the only repair we must consider is 1, so its negation (8) is false. Finally, if (7) suffers presupposition failure and takes the truth value #, then we have two repairs - 0 and 1, so we need to evaluate both. We find that $(\neg 0) \neq (\neg 1)$, so the function does not produce the same value for all possible repairs of the argument, so the value of this \neg deployed at # is #. That is, (8) suffers presupposition failure iff (7) does, so it is predicted that both presuppose that Smith is incompetent.

2.1.2. Deployment of Disjunction

For disjunction, consider the following four sentences:

- (9) Jones has stopped smoking.
- (10) Jones has never smoked.
- (11) Jones has never smoked, or Jones has stopped smoking.
- (12) Jones has stopped smoking, or Jones has never smoked.

To understand the truth, falsehood, and failure conditions of (11), we compute the deployment of the truth function $\llbracket \text{or} \rrbracket$ over the truth values of (9) and (10). First, then, what does it take to make (11) true? If Jones has never smoked, then (10) takes truth value 1, and (9) takes truth value #. The two repairs of # are 0 and 1, but $\llbracket \text{or} \rrbracket(1, 0) = 1$ and $\llbracket \text{or} \rrbracket(1, 1) = 1$, so both repairs lead to the same value, so $\llbracket \text{or} \rrbracket[1, \#] = 1$. So if Jones has never smoked, (11) is true, as it is if Jones is a former smoker (in which case we have $\llbracket \text{or} \rrbracket[0, 1] = 1$, since if Jones is a former smoker (9) is true and (10) is false). If neither of these is the case, then (10) must be false, so Jones smoked at one point, so the presuppositions of (9) are met, but Jones is not an ex-smoker and has smoked, and so must still smoke, so (9) is false as well. Since both disjuncts are false, (11) is in this case false. This exhausts all the cases: (11) is true if Jones has never smoked or Jones is an ex-smoker, and false otherwise (i.e., when Jones persists in smoking), and so has no presuppositions, even though (9) is presuppositional. Further, since the system as presented so far is symmetric, (12) has the same truth conditions and presuppositions as (11), and so is again presuppositionless. This is plausible, but somewhat contentious.

The fact that the disjunction examples above do not have any predicted presuppositions is a result of the choice of disjuncts. The system does allow a sentence formed by disjunction to suffer presupposition failure, as in the following example:

- (13) Jones called Smith, or Smith knows Jones is incompetent.

This sentence is assigned the truth value # if Jones didn't call Smith (so that the first disjunct is false) but Jones is not incompetent (so the second disjunct suffers failure). In this case, the deployment that determines the truth value of (13) is $\llbracket \text{or} \rrbracket[0, \#]$. To evaluate this, one must consider the two repairs of #: 0 and 1, so the possibilities are $\llbracket \text{or} \rrbracket(0, 0) = 0$ and $\llbracket \text{or} \rrbracket(0, 1) = 1$. Since $0 \neq 1$, the two repairs yield

different outcomes, so $\llbracket \text{or} \rrbracket[0, \#] = \#$. This is the only case in which failure results, for this sentence: if Jones did call Smith and Jones is not incompetent, the truth value is $\llbracket \text{or} \rrbracket[1, \#]$, which is 1, as noted above, and in all other cases both disjuncts have their presuppositions met, so there's no need to worry about multiple repairs. (13) suffers failure iff Jones didn't call and Jones is not incompetent, or, equivalently, it presupposes that if Jones didn't call then Jones is incompetent. This is the standard prediction, and we again predict that reversing the order of the disjuncts will not change the truth-conditions or presuppositions.

2.1.3. Deployment of Conjunction

For conjunction, consider the following:

(14) France is a monarchy and the king of France is bald.

(14) is true if both conjuncts are true: in this case, since 1 is the only repair for 1 we have $\llbracket \text{and} \rrbracket[1, 1] = \llbracket \text{and} \rrbracket(1, 1) = 1$. Next, the sentence is false if either conjunct is false: if France is not a monarchy then since $\llbracket \text{and} \rrbracket(0, 1) = 0$ and $\llbracket \text{and} \rrbracket(0, 1) = 0$, $\llbracket \text{and} \rrbracket[0, \#] = 0$ as well, and if France has a non-bald king then likewise this is enough to guarantee falsehood. The only remaining case is the case where one conjunct is true and the other has truth value # - that is, when France is a monarchy without a king, (because, for example, it has an empress instead), in which case we have $\llbracket \text{and} \rrbracket[1, \#]$. The two repairs to consider are $\llbracket \text{and} \rrbracket(1, 1) = 1$ and $\llbracket \text{and} \rrbracket(1, 0) = 0$. Since the repairs disagree, $\llbracket \text{and} \rrbracket[1, \#] = \#$. Thus, (14) suffers failure iff France is a monarchy but there is no king of France; equivalently, it presupposes that the king if France is a monarchy then it has a king. This is the standard claim for this sentence. Here, again, a simple deployment approach predicts symmetry, which is a bit hard to swallow.

2.2. Computing Repair Sets

For the case of truth values, the repairs for # were 0 and 1. To use deployment with quantifiers, we need to define repairs for higher-type arguments like predicates.

2.2.1. Two Possible Repairs

For every function g that maps sequences of arguments to truth values, two repairs will be of potential interest - the first, written $g^{0/\#}$, treats all cases of presupposition failure as if they were cases of falsehood, and the second $g^{1/\#}$, treats all cases of presupposition failure as if they were cases of truth. So, for example, (16) is true of those entities of which (15) is true, and false of all others, while (17) is false of all entities of which (15) is false, and true of all others - the two collapse the presuppositional part of (15) in opposite directions, while leaving the bivalent content from (15) unchanged.

- (15) $g = \llbracket \text{has stopped smoking} \rrbracket$
 (16) $g^{0/\#} \approx \llbracket \text{once smoked, but has stopped} \rrbracket$
 (17) $g^{1/\#} \approx \llbracket \text{has never smoked or has stopped smoking} \rrbracket$

If we treat truth values as zero-ary functions, they're a special case of this: $0^{0/\#} = 0^{1/\#} = 0$, and $1^{0/\#} = 1^{1/\#} = 1$, but $\#^{0/\#} = 0$ while $\#^{1/\#} = 1$: the two repairs are only different for $\#$.

One simple approach would be to generalize things straightforwardly by saying that we always consider both repairs. For many quantifiers commonly studied, the results of this would be tolerable, if a bit odd.⁵ Unfortunately, the presuppositions predicted by this approach are too strong in cases where quantifiers appear to treat presupposition failure as indistinguishable from falsehood, as with restrictors and *exactly* quantification:

- (18) Every student who knows that he is incompetent becomes depressed.
 (19) Exactly three of these students have stopped smoking.

If we must consider both repairs of the restrictor in (18), then the sentence is only true if both of the repair substitutions below are true:

- (20) $p = \llbracket \text{student who knows that he is incompetent} \rrbracket$
 (21) Every student who is incompetent and knows it becomes depressed.
 (Substitution of $p^{0/\#}$.)
 (22) Every student who is not incompetent or knows that he is incompetent becomes depressed.
 (Substitution of $p^{1/\#}$.)

However, the prediction that (22) must be true in order for (18) to be true is counterintuitive. Instead, (18) appears to have almost exactly the same truth conditions as (21): considering only one repair improves our predictions here.

We encounter the same issue with the *exactly* quantification in (19), where the sentence seems to have the same truth conditions as (24), but can apparently be quite true while (25) is false:

- (23) $p = \llbracket \text{has stopped smoking} \rrbracket$
 (24) Exactly three of these students once smoked, but have stopped.
 (Substitution of $p^{0/\#}$.)
 (25) Exactly three of these students never smoked or have stopped smoking.
 (Substitution of $p^{1/\#}$.)

From these kinds of examples, it seems tempting to only consider the $p^{0/\#}$ repair, but of course if applied generally this would make presupposition failure indistinguishable from falsehood. To identify the cases in which both repairs must be considered, it will be necessary to develop the notion of a relevant set.

⁵I have discussed the implications of this approach elsewhere (George 2008a,b); this simple approach is especially reminiscent of supervaluation (van Fraassen 1969) and strong Kleene (Kleene 1952) systems.

2.2.2. *Relevant Sets*

If we accept the empirical claims outlined in sections 1.2 and 2.2.1, and we want to pursue a theory involving $p^{0/\#}$ and $p^{1/\#}$ as repairs, we'll need to devise a system where sometimes both of these repairs are in admissible, and sometimes only $p^{0/\#}$ is. These cases will be distinguished using incremental relevant sets.

A relevant set is defined with respect to a function f that takes other functions as arguments.⁶ A relevant set is a set X such that, in evaluating f with respect to its next argument, one only needs to be concerned with the behavior of that argument with respect to the elements of X . In the case where f takes predicates as arguments, X can be thought of as a set containing all entities relevant to the determination of how a particular predicate will interact with f . The relevant set at hand will change, incrementally, as we progress through the arguments of the function, or, equivalently, the relevant set for a particular argument will depend on the prior but not the subsequent arguments.⁷ The major use of the notion of a relevant set is (very roughly) that it picks out for a quantifier's first argument position (the restrictor) the set consisting of the whole universe, while for the second argument position (the nuclear scope) it picks out the set described by the restrictor. Although the formalization here somewhat obscures this, it turns out (cf. George 2008b) that the relevant set can be recovered from the behavior of the function in light of the prior arguments without explicit knowledge of the content of the prior arguments - computing relevant sets allows us, in many cases, to recover the extension of the restrictor from a generalized quantifier.

Relevant sets are defined in terms of the idea of functions *agreeing on* a set:

Definition 2 (Agreement of Functions on a Set) *Functions b and b' agree on a set of arguments⁸ X iff, for all $x \in X$, $b(x) = b'(x)$.*

For example, the b and b' below agree on the set X of all students:

(26) $b = \llbracket \text{has stopped smoking} \rrbracket$

(27) $b' = \llbracket \text{has stopped smoking and is a student} \rrbracket$

A relevant set is a set X such that agreement on that X guarantees equivalent behavior in the argument position in question, thus:

Definition 3 (Relevant Sets) *A set X is a relevant set for a function f and a (partial, perhaps empty) list of arguments a_1, \dots, a_{m-1} iff, for every a pair of arguments b_m and b'_m such that b_m and b'_m agree on X , and every (possibly empty)*

⁶of course, with a suitable notion of zero-ary functions, this is not really a substantial limitation.

⁷In the definitions of relevant sets and the subsequent formal devices, I will be glossing over some issues that would make the proper formulation of these definitions a bit more cumbersome. I have developed a somewhat more rigorous and complete treatment elsewhere (George 2008b).

⁸We should perhaps say sequences of arguments, to account for the possibility that b and b' might not be unary.

list of non-presuppositional arguments c_{m+1}, \dots, c_n , $f(a_1, \dots, a_{m-1}, b_m, c_{m+1}, \dots, c_n) = f(a_1, \dots, a_{m-1}, b'_m, c_{m+1}, \dots, c_n)$.⁹

To pick a linguistically motivated example, consider the two-argument function $\llbracket \text{each} \rrbracket$, with $\llbracket \text{student} \rrbracket$ for its first argument. Any set X such that all students are in X is a relevant set for $\llbracket \text{each} \rrbracket$ and $\llbracket \text{student} \rrbracket$, since for any such X and any predicates p and p' that agree on X , p and p' have the same value for any student, so each student has property p iff each student has property p' , so $\llbracket \text{each} \rrbracket(\llbracket \text{student} \rrbracket, p) = \llbracket \text{each} \rrbracket(\llbracket \text{student} \rrbracket, p')$ (since the function takes only two arguments, the list of later arguments is empty). That any set containing all students would do the job is a consequence of the conservativity of natural language determiners (Keenan and Stavi 1986), and that no set not containing all students will do the job follows from the observation that if we had a set X that left out even one student, a function p mapping all students to 1 and a function p' mapping just that one excluded student to 0 would agree on X , but would produce different outcomes for this quantification.

For purposes of this paper, I will speak of *the* relevant set of a function and a list of arguments when I mean the *smallest* relevant set. In general, of course, not every function has a least relevant set, but the exceptions will not be an issue when we restrict attention to finite domains and linguistically typical quantifiers.¹⁰ Observe in particular that, for most quantifiers, the relevant set of the quantifier with the empty argument list is the universe of all entities, and the relevant set of the quantifier with an argument list containing a well-behaved restrictor is the restrictor. The relevant set picks out ‘all the stuff we’re concerned with’.¹¹

2.2.3. Identifying the Repairs

Given the relevant set, we can identify the available repairs. For a function g , the function $g^{0/\#}$ is always considered as an available repair, but the function $g^{1/\#}$ is only considered a repair $g^{0/\#}$ is ‘too boring’ in that it maps nothing in the relevant set to 1:

Definition 4 (Available Repairs) *For any function g being evaluated as an argument of a function f and preceded by arguments a_1, \dots, a_n , where X is the relevant set of f and a_1, \dots, a_n , if $g(x) = 1$ for some $x \in X$ then $g^{0/\#}$ is the only available repair for g , but if there is no $x \in X$ such that $g(x) = 1$, then the available repairs for g are both $g^{0/\#}$ and $g^{1/\#}$.*

⁹This definition is technically problematic when a preceding argument had presuppositional content, but the generalization to this case (used in George 2008b) doesn’t involve any interesting conceptual changes.

¹⁰The way to avoid such issues is to replace all references to ‘the relevant set’ in what follows with references to ‘every relevant set’.

¹¹When truth values are understood as zero-ary functions and relevant sets are sets of argument sequences, the relevant set of a truth function with a partial argument list is either the empty set or the set containing only the empty sequence.

This means that under function deployment quantifiers will sometimes p only consider $p^{0/\#}$, in which case presupposition failure and falsehood will be equivalent. However, in certain cases $p^{1/\#}$ will come into play, and, perhaps, introduce enough uncertainty to cause presupposition failure. Note that, if p is non-presuppositional, then $p^{0/\#} = p^{1/\#} = p$, so we really only need to worry about what repair set we're choosing when presuppositional predicates are involved. If p maps everything in the relevant set to 0 (and so doesn't map anything to #), then both $p^{0/\#}$ and $p^{1/\#}$ are considered, but they're also both the same. Now that we know how to find the repairs for a broader array of functions, our definition of deployment stays the same: if all available repairs produce the same result, we go with that result, but if the available repairs yield different results we have failure.

This approach produces mostly appropriate truth conditions, but gives us rather weak presuppositions. To see this, consider the following example:

(28) Each student has stopped smoking.

To evaluate (28), we deploy $\llbracket \text{each} \rrbracket$ on the arguments $\llbracket \text{student} \rrbracket$ and $\llbracket \text{has stopped smoking} \rrbracket$. Now, since $\llbracket \text{student} \rrbracket$ has no presuppositions, its only possible repair is itself, so we just need to identify the repairs of $p = \llbracket \text{has stopped smoking} \rrbracket$. The relevant set of $\llbracket \text{each} \rrbracket$ and $\llbracket \text{student} \rrbracket$ is just the set of students, so if at least one student is a former smoker, then p maps at least one member of the relevant set to 1, and the only available repair is $p^{0/\#} \approx \llbracket \text{used to smoke but doesn't smoke at present} \rrbracket$. If none of the students is a former smoker, then p maps everything in the relevant set to 0 or #, so the available repairs will be both $p^{1/\#} \approx \llbracket \text{doesn't smoke} \rrbracket$ and $p^{0/\#}$. We can now identify the predicted truth, falsehood, and failure conditions of the sentence.

First, when is (28) predicted to take the truth value 1? For this to be the case, it must be that $\llbracket \text{each} \rrbracket(\llbracket \text{student} \rrbracket, p') = 1$, for every p' that's an available repair of p . This is the case iff $\llbracket \text{each} \rrbracket(\llbracket \text{student} \rrbracket, p^{0/\#}) = 1$, since then (by monotonicity of $\llbracket \text{each} \rrbracket$) $\llbracket \text{each} \rrbracket(\llbracket \text{student} \rrbracket, p^{1/\#}) = 1$, so for establishing truth we only need to consider $p^{0/\#}$, which is always an available repair. Now, $\llbracket \text{each} \rrbracket(\llbracket \text{student} \rrbracket, p^{0/\#}) = 1$ iff every student is a former smoker, so that's when (28) is true. Thus if (28) is true then every student has smoked, so the inference that's normally described as a universal presupposition comes through as an entailment.

Next, when is (28) predicted to take the truth value 0? This can come about in two ways. First, if even one student is a former smoker, then, as noted above, the only available repair is $p^{0/\#}$, so if there is one student former smoker and $\llbracket \text{each} \rrbracket(\llbracket \text{student} \rrbracket, p^{0/\#}) = 0$, then we predict (28) is false. That is, the sentence is false when there is one student who is a former smoker, but it is not the case that every student is a former smoker. This means we predict that the sentence can take the truth value 0 even if some students have never smoked, provided that there is one student who used to smoke and has given it up. On the other hand, if no student is a former smoker, then both $p^{0/\#}$ and $p^{1/\#}$ are available repairs, so for the (28) to be false it needs to be false when we substitute either of these as an argument, so it needs to be false that every student is a former smoker and it also needs to be false that every student is a nonsmoker - that is, there needs to be one student who still

smokes.

Finally, when do we predict (28) takes the truth value #? The sentence suffers presupposition failure if it is neither true nor false. More explicitly, we have presupposition failure when both repairs are available (which is the case iff no student is a former smoker) and they produce different results, so the propositions that every student is a former smoker and that every student is a nonsmoker must have different truth values. Since in this case no student is a former smoker, it follows, so long as there are any students at all, that it must be false that every student is a former smoker, but then to get the disagreement that leads to failure, it must be true that every student is a nonsmoker. Thus, to have presupposition failure, it must be the case that no student presently smokes, but also that no student is a former smoker, which is to say it must be the case that no student has ever smoked. Thus, the present approach predicts that (28) presupposes the opposite of this: it presupposes that at least one student has smoked at some point.

Thus, the system as described so far predicts only an existential presupposition from the nuclear scope of a universal quantification. As already noted, it also predicts a symmetrical projection behavior for *and*. A revision of the deployment operation will strengthen presuppositions to address these issues.

2.3. Revising Function Deployment

The function deployment I want to advocate here modifies the version presented above to regain universal presuppositions for some quantifiers, including the universal quantifier, and to make the projection behavior of *and* asymmetrical. The underlying intuition is that, in reasoning about presuppositions, we consider separately the preconditions for truth and the preconditions for falsehood, and that we want, on top of the deployment discussed so far, to demand that the preconditions for truth are (incrementally) satisfied before we permit any non-failure truth value, even 0. Put another way, if we move through the arguments of a function one at a time, we demand that we should never be forced to conclude that the output of the function is not 1 based on the presuppositions of an argument - truth may be ruled out, of course, but it must be ruled out by the assertive, rather than the presuppositional, component of an argument, and if the presuppositions ever rules out truth then the entire expression suffers presupposition failure.

The above will be codified in terms of the notion of disappointing arguments, where disappointment is defined in terms of the function deployment developed above. To define disappointment, we need to be able to talk about which aspects of an argument's behavior are caused by its presuppositional rather than its assertive component. The key to this is the notion of presupposition-equivalence.

Definition 5 (Presupposition-equivalence) *Functions f and g of the same type are presupposition-equivalent iff they map exactly the same arguments to #. (In the case of zero-ary truth functions (i.e. truth values) this means that 0 and 1 are presupposition-equivalent to each other and to themselves, but # is presupposition-equivalent only to itself.)*

Thus, for example, the following predicates are all presupposition-equivalent to each other, since, although they may be true or false of different entities, they all map the same entities (those without advisors) to #:

- (29) admires his advisor.
 (30) admires his advisor and is allergic to grapefruit.
 (31) admires his advisor and drinks heavily.

With this notion of equivalence, we can define disappointment:

Definition 6 (Disappointment) *For any function f and any initial sequence of arguments a_1, \dots, a_{m-1} , the argument b_m has disappointing presuppositions with respect to f and a_1, \dots, a_{m-1} iff the following two criteria both hold:*

- *For every b'_m that is presupposition-equivalent to b_m , and for all c_{m+1}, \dots, c_n , $f[a_1, \dots, a_{m-1}, b'_m, c_{m+1}, \dots, c_n] \neq 1$.*
- *There are d_m, \dots, d_n such that $f[a_1, \dots, a_{m-1}, d_m, \dots, d_n] = 1$.*

The first criterion says that, based only on b_m 's presuppositions and what has come before, we can rule out any hope for truth, and the second says that the arguments before b_m hold out hope for truth, so that we can be sure b_m is the source of the disappointment.

To see how disappointment works, suppose that there are three students, that one of them smokes and has smoked for some time, and that the other two don't smoke and never have. Now consider the function $\llbracket \text{each} \rrbracket$ and the initial argument $\llbracket \text{student} \rrbracket$. In this context, the nuclear scope argument $\llbracket \text{has stopped smoking} \rrbracket$ is disappointing. The first condition for disappointment is met because $\llbracket \text{each} \rrbracket [p, q] = 1$ iff everything that has property p has property q , so in particular nothing with property p can cause anything with property q to suffer presupposition failure, but the two never-smoking students do cause $\llbracket \text{has stopped smoking} \rrbracket$ to suffer presupposition failure, and will likewise for any predicate that is presupposition-equivalent to $\llbracket \text{has stopped smoking} \rrbracket$. The second condition of disappointment just demands that there be a way to fill in the nuclear scope that makes a sentence starting *each student...* come out true - here $\lambda x.(1)$ will work. Having seen the quantifier and the restrictor, there is hope of truth, but the presuppositional aspects of the nuclear scope dash these hopes.

Now all that is left is to build a disappointment check into a revised notion of deployment:

Definition 7 (Deployment with Concern for Disappointment) *For any function f and any full sequence a_1, \dots, a_n of arguments for f , $f:(a_1, \dots, a_n)$; the deployment of f on a_1, \dots, a_n with concern for disappointment, is defined as follows:*

- *If there is $m \leq n$ such that a_m is disappointing with respect to f and a_1, \dots, a_{m-1} , then $f:(a_1, \dots, a_n) := \#$.*

- *Otherwise*, $f:(a_1, \dots, a_n): = f[a_1, \dots, a_n]$.

That is, deployment with concern for disappointment always yields failure when any of the arguments is disappointing in light of the preceding arguments, and otherwise it behaves like normal deployment by finding all the repairs for each argument (based on the relevant set calculation discussed above) and seeing if every choice of repairs produces the same results.

2.3.1. *Deployment with Disappointment and the Connectives*

To see how deployment with disappointment works, consider the case of *and*, and in particular the following two sentences:

(32) Smith is a fool and he knows that he is a fool.

(33) Smith knows that he is a fool and he is a fool.

In assessing these sentences, there are three cases to consider. First, there is the case where Smith is a fool and is aware of his own foolishness. In this case, both conjuncts are true, so both sentences take the truth value $\llbracket \text{and} \rrbracket:(1, 1)$. Now, 1 is not disappointing in either of these argument positions since in both cases its presuppositional content is compatible with truth: 1 is presupposition-equivalent to 1, and $\llbracket \text{and} \rrbracket[1, 1] = \llbracket \text{and} \rrbracket(1, 1) = 1$. Since the arguments are not disappointing, we use regular function deployment. The only repair for 1 in either context is 1, so the sentence is true.

Next, consider the case where Smith is a fool, but doesn't know it. In this case, (32) takes the truth value $\llbracket \text{and} \rrbracket:(1, 0)$. Here, the initial 1 is not disappointing because it is compatible with truth (in the case where hypothetical the second argument is 1), and 0 is not disappointing because it is presupposition-equivalent to 1, which will yield truth here. With (33), the truth value is given by $\llbracket \text{and} \rrbracket:(0, 1)$. The initial 0 is not disappointing because it is presupposition-equivalent to 1, and 1 in the first argument position of $\llbracket \text{and} \rrbracket$ is compatible with an outcome of truth. For the second argument position there is no hope of truth: the presence of 0 in the first position makes falsehood a certainty, and there is no second argument that could save us, so the second argument is not the source of disappointment. In both cases, we have ruled out disappointment, and so we compute the deployment of $\llbracket \text{and} \rrbracket$ onto 0 and 1, in either order, and compute a truth value of 0.

Finally, there is the case where Smith is not a fool at all. Here the truth value of (32) is $\llbracket \text{and} \rrbracket:(0, \#)$. Now, the initial 0 is not disappointing (since it is presupposition-equivalent to 1), and the subsequent # is not disappointing (because the preceding 0 means we had no hope left to lose). Thus $\llbracket \text{and} \rrbracket:(0, \#) = \llbracket \text{and} \rrbracket[0, \#]$, which is 0, since $\llbracket \text{and} \rrbracket(0, 0) = \llbracket \text{and} \rrbracket(0, 1) = 0$. This time, though, we do find some asymmetry with (33): here the truth value is given by $\llbracket \text{and} \rrbracket:(\#, 0)$, and now the first argument is disappointing. To see this, note that $\llbracket \text{and} \rrbracket[\#, 0] = 0$ (since substituting either 0 or 1 for # will produce the same false result) and $\llbracket \text{and} \rrbracket[\#, 1] = \#$ (since substituting 0 and substituting 1 would produce different outcomes, meaning there is a

lack of consensus between the repairs). Further, # is the only thing presupposition-equivalent to itself, so just knowing its presupposition-equivalence class we can rule out an outcome of 1. Before, we saw #, we still had hopes to disappoint, since knowing only the arguments before # (that is, the empty argument list), there was no way to rule out an outcome of 1. This means that (33) suffers presupposition failure when Smith is not a fool. This is exactly the asymmetry associated with traditional accounts of the projection of *and*, on which (33) presupposes that Smith is a fool while (32) does not.

Since it is easier to get truth out of disjunction, *or* remains symmetrical after the disappointment modification: since $\llbracket \text{or} \rrbracket [\#, 1] = 1$ (because $\llbracket \text{or} \rrbracket (0, 1) = \llbracket \text{or} \rrbracket (1, 1) = 1$), a # in the first argument position of a disjunction is not disappointing like it is for a conjunction.

3. Some Predictions for Quantifiers

At last, we can explore the projection predictions for a few different quantificational cases. We find that the system predicts universal presuppositions for *each* and *none* quantification, but otherwise shares with the more basic deployment system that was first developed the appealingly weak presupposition effects for most other quantifiers, and for presuppositional restrictors. The predictions that we obtain here are distinctive in that they involve sharp contrasts between different quantifiers, and they are also compatible with my own judgments.

3.1. Restrictors

Consider the following example of a presuppositional restrictor:

- (34) Every student in the course who stopped smoking complained about the exam.

Suppose that even one student was a former smoker. Since the relevant set of $\llbracket \text{every} \rrbracket$ is the whole universe of entities, $\llbracket \text{student in the course who stopped smoking} \rrbracket$ maps at least one entity in that relevant set (in particular, the student who stopped smoking) to 1, so we only need to consider the repair that substitutes 0 for # (i.e. the predicate $\llbracket \text{student in the course who had smoked in the past, and stopped} \rrbracket$). Thus, if there was one student former smoker, the sentence is true if every student former smoker complained about the exam, and false otherwise. We don't have to worry about disappointment here, since no restrictor could rule out the possibility of an outcome of 1 (we could always put the predicate $\lambda x.(1)$ in the nuclear scope), and the nuclear scope value is non-presuppositional, so it is presupposition-equivalent to $\lambda x.(1)$ (since neither maps anything to #), so it does not rule out hope for an outcome of truth by virtue of its presuppositions. We thus avoid undesirably strong projection of restrictor presuppositions.

3.2. at least one

Now let's look at the projection properties of the nuclear scope of some quantifiers. First, consider simple existential quantification:

(35) At least one of these ten students has stopped smoking.

Let's begin by figuring out regular deployment and then see where disappointment comes in. Under normal deployment if at least one of the students is a former smoker, then the sentence gets truth value 1, since then it is true for either repair of the nuclear scope $p = \llbracket \text{has stopped smoking} \rrbracket$ that at least one of the ten students has the property identified by that repair (although in fact only one repair will be considered in this case). If none of the ten students is a former smoker, then (since the relevant set is the set of students), p maps nothing in the relevant set to 1, so both $p^{0/\#} \approx \llbracket \text{is a former smoker} \rrbracket$ and $p^{1/\#} \approx \llbracket \text{doesn't smoke} \rrbracket$ are acceptable repairs. We know that the quantification is false when we substitute $p^{0/\#}$, since none of the students are former smokers, so, if it is also false when we substitute $p^{1/\#}$, the deployment is false, but otherwise we get the truth value # - that is, under simple deployment, we get 0 if none of the students are former smokers and all of the students still smoke, and # if none of the students are former smokers but some don't smoke (because they've never smoked).

Disappointment doesn't change this. On the assumption that there is a group of ten students to talk about, the restrictor is not disappointing (because it's just the set of ten students, which could always hold out hope for truth if combined with a predicate like $\lambda x.(1)$). Meanwhile, if at least one student has smoked in the past, then the nuclear scope is presupposition-equivalent to a predicate that maps that student to 1, which is enough to make the deployment true. Finally, if none of the students have smoked in the past, then we do get disappointment, but then we would have had failure under deployment anyway.

Thus, the sentence is true if one of the students is a former smoker, but for it to be false the nuclear scope must be false of all the students, so its presupposition must be true of all the students. Thus, we predict a presupposition that either at least one student is a former smoker or all of the students persist in smoking.

3.3. each

Now consider the (36), with truth value determined by (37):

(36) Each of these ten students has stopped smoking.

(37) $\llbracket \text{each} \rrbracket : (\lambda x.(x \text{ is one of these ten students}), \llbracket \text{has stopped smoking} \rrbracket)$:

First, let's consider when the arguments are disappointing. The first argument (the restrictor) is never disappointing, since there is always some next argument that will yield an outcome of 1, since $\llbracket \text{each} \rrbracket [p, \lambda x.(1)] = 1$ for any choice of p , including any p presupposition-equivalent to $\lambda x.(x \text{ is one of these ten students})$.

Turning to the second argument (the nuclear scope), if even one of the ten students has never smoked, $\llbracket \text{has stopped smoking} \rrbracket$ is disappointing with respect to $\llbracket \text{each} \rrbracket$ and $\lambda x.(x \text{ is one of these ten students})$, because if there is a student y who has never smoked, $\llbracket \text{has stopped smoking} \rrbracket(y) = \#$, so $q^{0/\#}(y) = 0$ for any q that is presupposition-equivalent to $\llbracket \text{has stopped smoking} \rrbracket$. Thus, since $q^{0/\#}$ is always among the repairs of q , it's always one of the repairs we'd consider in deployment, so the quantified sentence might evaluate to 0 or $\#$ (depending on whether there's another repair and what it is), but never to 1. Since the nuclear scope is disappointing whenever any of the students doesn't smoke and never has, we get presupposition failure (truth value $\#$) whenever any of the students has never smoked. This is the universal presupposition that every student has smoked before. If, on the other hand, this presupposition is met (i.e. if every student used to smoke), then there is no fear of disappointment, since $\llbracket \text{has stopped smoking} \rrbracket$ doesn't fail for any of the students, so it is presupposition-equivalent to the function that maps all the students to 1 but behaves identically to $\llbracket \text{has stopped smoking} \rrbracket$ for all other arguments, and this predicate will make the sentence true.

In the case where the nuclear scope of (36) is not disappointing, we find that all of the students have smoked in the past, so for the nuclear scope $p = \llbracket \text{has stopped smoking} \rrbracket$, both $p^{0/\#}$ and $p^{1/\#}$ behave identically so far as the students are concerned, so (36) gets truth value 1 if all the students are former smokers, and 0 if all of the students have smoked and at least one of them still does, and $\#$ (by disappointment) in all the cases where at least one student has never smoked.

3.4. none

Next, consider (38), the truth value of which is evaluated by computing the deployment (39):

(38) None of these ten students has stopped smoking.

(39) $\llbracket \text{none} \rrbracket : (\lambda x.(x \text{ is one of these ten students}), \llbracket \text{has stopped smoking} \rrbracket)$:

As with the discussion of *each*, the first argument here is never disappointing, so long as the reference to the ten students is unproblematic. However, the second argument is again disappointing iff there is a student who has never smoked. To see this, suppose some student y has never smoked, and let q be any predicate presupposition-equivalent to $\llbracket \text{has stopped smoking} \rrbracket$. We know that $q(y) = \#$, but we have two cases to consider. If there is any student z such that $q(z) = 1$ then in evaluating the deployment $\llbracket \text{none} \rrbracket[\lambda x.(x \text{ is one of these ten students}), q]$, we would only need to consider $q^{0/\#}$, but $q^{0/\#}(z) = 1$, so it's not the case that $q^{0/\#}$ is true of none of the students, so the deployment would evaluate to 0. If, on the other hand, q maps none of the students to 1, then (since the students are the relevant set here), both $q^{0/\#}$ and $q^{1/\#}$ are available as repairs, but $q^{1/\#}(y) = 1$, so $q^{1/\#}$ is true of some of the students, so the deployment of $\llbracket \text{none} \rrbracket$ can't give us truth here either. Since we encounter disappointment if there is any student who has never smoked, we have a universal presupposition that each of the students has smoked in the past. (Note

that there was hope of truth before we looked at the nuclear scope, satisfying the other requirement for disappointment.)

If we avoid disappointment, no other presuppositional issues arise, since in that case the nuclear scope suffers presupposition failure for none of the students, so we have a bivalent case, and the sentence is true iff all of the students persist in smoking, and false iff all of the students have smoked and at least one has stopped.

3.5. exactly three

Finally, consider the quantifier *exactly three*:

(40) Exactly three of these ten students have stopped smoking.

Here, it will be best to work out the regular deployment and then check for disappointment. For regular deployment. The restrictor, being non-presuppositional, has only one repair. If any of the students are former smokers, then the nuclear scope is true of something in the relevant set (the restrictor), so it has only one repair (roughly \llbracket is a former smoker \rrbracket) as well - in this case, the deployment comes out 1 if the number of student former smokers is three, and 0 otherwise. On the other hand, if none of the students are former smokers, then we must consider two repairs, given roughly by \llbracket is a former smoker \rrbracket and \llbracket doesn't smoke \rrbracket ; since in this case none of the students are former smokers, the substitution of \llbracket is a former smoker \rrbracket produces truth value 0, so here we get 0 for the deployment if the number of students who have never smoked is anything other than three, and # if the number of students who have never smoked is exactly three.

Now we turn to disappointment. The restrictor is again unproblematic, having no presuppositional content that concerns us. Further, if at least three students have smoked before, then the nuclear scope is not disappointing, since then it is presupposition-equivalent to some function that maps three students to 1 and the rest to 0 or #, which under the deployment above will make the truth value come out 1. However, if fewer than three of the students have ever smoked, we do get disappointment, since in this case any predicate q presupposition-equivalent to the nuclear scope maps at most two of the students to truth values other than #, so it maps at most two of the students to 1, so $q^{0/\#}$ maps no more than two of the students to 1, ruining any hope of q being true of exactly three of the students.

Under deployment with concern for disappointment, (40) suffers presupposition failure by disappointment if fewer than three of the students have smoked, and it suffers presupposition failure from the deployment system if exactly three of the students have never smoked and the rest presently smoke, and it is true if exactly three of the students are former smokers (regardless of the smoking history, or lack thereof, of the remaining seven students), and false in all other cases. This is a desirably weaker-than-universal presupposition.

References

- Beaver, David and Emiel Kraahmer: 2001, 'A Partial Account of Presupposition Projection', *Journal of Logic, Language and Information* **10**, 147–182.
- Chemla, Emmanuel: 2007, 'Presuppositions of Quantified Sentences: Experimental Data'. Manuscript, ENS.
- Chemla, Emmanuel: 2008, 'Similarity: Towards a Unified Account of Scalar Implicatures, Free Choice Permission and Presupposition Projection'. Manuscript, ENS and MIT.
- van Fraassen, Bas: 1969, 'Presuppositions, Supervaluations and Free Logic', in K. Lambert (ed.), *The Logical Way of Doing Things*, 67–92. Yale University Press, New Haven.
- George, Benjamin R.: 2008a, 'Predicting Presupposition Projection: some alternatives in the strong Kleene tradition'. Manuscript, UCLA.
- George, Benjamin R.: 2008b, 'Presupposition Repairs: a Static, Trivalent Approach to Predicting Projection'. Master's thesis, UCLA.
- Heim, Irene: 1983, 'On the Projection Problem for Presuppositions', in M. Barlow, D. Flickinger, and M. Westcoat (eds.), *Second Annual West Coast Conference on Formal Linguistics*, 114–126. Stanford University.
- Heim, Irene: 1990, 'Presupposition Projection', in R. van der Sandt (ed.), *Reader for the Nijmegen Workshop on Presupposition, Lexical Meaning, and Discourse Processes*. University of Nijmegen.
- Keenan, Edward L. and Jonathan Stavi: 1986, 'A Semantic Characterization of Natural Language Determiners', *Linguistics and Philosophy* **9**, 253–326.
- Kleene, Stephen: 1952, *Introduction to Metamathematics*. North Holland, Amsterdam.
- LaCasse, Nicolas R.: 2008, 'Constraints on Connectives and Quantifiers: Solving the over-generation problem of dynamic semantics'. Manuscript, UCLA.
- Peters, Stanley: 1977, 'A Truth-Conditional Formulation of Karttunen's Account of Presupposition', in *Texas Linguistic Forum* **6**. University of Texas at Austin.
- Schlenker, Philippe: 2006, 'Anti-Dynamics: Presupposition Projection Without Dynamic Semantics', *Journal of Logic, Language and Information* **16**, 325–356.
- Schlenker, Philippe: 2008a, 'Be Articulate: A Pragmatic Theory of Presupposition Projection', *Theoretical Linguistics*.
- Schlenker, Philippe: 2008b, 'Local Contexts: Presuppositions, Dynamic Semantics and Transparency'. Manuscript, Institut Jean-Nicod and NYU.
- Soames, Scott: 1989, 'Presupposition', in D. Gabbay and F. Guenther (eds.), *Handbook of Philosophical Logic*, Vol. IV, 553–616. Reidel, Dordrecht.