Journal of Applied Non-Classical Logics, 2014
Vol. 24, Nos. 1–2, 86–117, http://dx.doi.org/10.1080/11663081.2014.911521

Taylor & Francis
Taylor & Francis Group

Some remarks on certain trivalent accounts of presupposition projection

Benjamin R. George*

Department of Linguistics and Department of Philosophy, Yale University, New Haven, Connecticut, United States of America

This paper discusses some formal properties of trivalent approaches to presupposition projection, and in particular of the middle Kleene system of Peters (1977) and Krahmer (1998). After exploring the relationship between trivalent truth-functional accounts and dynamic accounts in the tradition of Heim (1983), I show how the middle Kleene trivalent account can be formulated in a way which shows that it meets the explanatory challenge of Schlenker (2006, 2008a,b), and provide some results relating to the application of the middle Kleene approach to generalised quantifiers.

Keywords: presupposition projection; presupposition; trivalent logic; explanatory theories; semantics

1. Introduction

This paper presents a number of observations related to trivalent truth-functional approaches to presupposition projection in the tradition of Peters (1977), and their relationship to dynamic approaches of the sort used by Heim (1983) and Beaver (2001).

I present three major lines of investigation. First, in Section 2, I show that, although dynamic approaches generally have a great deal of power to define arbitrary projection patterns for connectives, a plausible generalisation of the dynamic toolbox employed by Heim (1983) can only build dynamic connectives that are equivalent to trivalent truth functions. This means that any connectives built using this toolbox will inherit certain formal properties of trivalent truth-functional approaches.[1] This also provides tools that may be useful in transferring results derived in a trivalent truth-functional setting to Heim-type dynamic accounts. While both the Peters (1977) and Heim (1983) connectives are already known to derive the projection patterns described by Karttunen (1973, 1974) (when the latter is augmented with the disjunction of Beaver, 2001), and so to make the same predictions for projection under propositional connectives, I offer the more general result that a larger class of possible dynamic connectives have equivalent trivalent truth functions.

Next, in Section 3, I show how the tools of George (2008c) allow us to make the intuition behind Peters's particular choice of trivalent connectives fully explicit and rigorous, and to generalise it to arbitrary truth functions (and eventually to other functions as well). The core intuition rests on the idea that the Peters (1977) connectives are essentially asymmetrical versions of the strong trivalent connectives of Kleene (1952). This link was already explicitly recognised by Krahmer (1998) (Krahmer calls the Peters connectives 'middle Kleene' connectives), and here I take the step of formalising this intuition fully by using the

*Email: brgeorge@marblesandunicorns.net

George (2008c) approach to incrementalising the principle behind the strong Kleene logic. This allows me to show how a general principle for incrementally applying bivalent truth functions to trivalent arguments derives the Peters connectives. That is, Krahmer's insight provides the conceptual basis for situating the Peters connectives within a framework that meets the 'explanatory challenge' of Schlenker (2008a,b).

Finally, in Section 4, I describe a natural generalisation of this idea to quantifiers. Proposals for generalising strong Kleene and related systems to particular quantifiers have a long history in the literature, but the present proposal gives a principled way of applying the generalised middle Kleene approach to arbitrary extensional determiner-type quantifiers, and so makes it possible to derive general results about which quantifiers will give rise to which sorts of presuppositional inferences. The predictions here deviate from the standard projection facts, in ways that are sometimes plausible, and sometimes more questionable. This provides an area where the predictions of a non-augmented middle Kleene approach diverge from those of most of its competitors. To the extent that these predictions are a better fit for our observations, this will count as an advantage of the middle Kleene approach. To the extent that they are not, they constitute strong evidence that it will need to be revised or complicated.[2]

Before turning to these formal observations, I will begin by briefly reviewing the projection problem, some of the explanatory criteria that have been important in debates about presupposition projection, and the basic architecture of the Peters and Heim-Beaver connectives.

1.1. The projection problem for presuppositions

The projection problem for presuppositions is the problem of deriving the presuppositions of a complex sentence from the presuppositions of its parts. For example, (1-a) presupposes (1-b), but (2-a) carries only a weak presupposition, along the lines of (2-b):[3]

(1) a. The Czar of the United States is bald.
    b. The United States has a Czar.

(2) a. If the United States is a monarchy, then the Czar of the United States is bald.
    b. If the United States is a monarchy, then the United States has a Czar.

The projection problem has been studied within both semantic and pragmatic frameworks, using a variety of different formal tools.

1.2. Explanatory criteria

Debates in the study of presupposition projection have often turned on what have typically been called 'explanatory' properties of competing accounts, and I want to briefly review some of the issues.[4]

To pick a representative statement of explanatory criteria, Heim (1983), echoing Gazdar (1979), complains that the account of Karttunen and Peters (1979) 'merely describes the projection facts instead of explaining them' (Heim, 1983, as reprinted in Davis, 1991, p. 397). Heim elaborates the issue as follows:

> G.'s [Gazdar's] point of criticism is that the K.&P.-theory treats these three properties [triggering, projection, and classical content] as mutually independent. None of them is derived from the other two. The theory thus implies—implausibly—that someone who learns the word 'if' has to learn not only which truth function it denotes and that it contributes [i.e., triggers] no

> presupposition, but moreover that it has the heritage [i.e., projection] property specified in (4) [Heim's statement of the standard projection facts for the conditional]. It also implies that there could well be a lexical item—presumably not attested as yet—whose [classical] content and presupposition [triggering] properties are identical to those of 'if', while its heritage [projection] property is different. We have to agree with G. that a more explanatory theory would not simply stipulate (4) as a lexical idiosyncrasy of 'if', but would somehow derive it on the basis of general principles and the other semantic properties of 'if'. (Heim, 1983, as reprinted in Davis, 1991, p. 398)

These explanatory criteria are a key component of Heim's rejection of the Karttunen and Peters (1979) framework in favour of a dynamic account of presupposition projection.[5]

One component of this complaint relates to *overgeneration*. The Karttunen and Peters (1979) framework recognises any combination of a truth function and a projection pattern as a possible connective meaning, so it fails to rule out many implausible unattested connectives. A framework that ruled out such deviant connectives would appear more desirable: by limiting possible connectives, such an account might hope to address learnability, identify universals of connective meaning, and provide a predictively stronger scientific theory. For these sorts of reasons a more *constraining* account of projection is to be preferred.

Unfortunately, the Heim (1983) framework itself overgenerates. For example, as pointed out by Soames (1989), nothing in the Heim (1983) framework excludes the possibility of defining a connective *and** which is truth-conditionally identical to ordinary *and*, but has linearly reversed presupposition projection properties. That is, (3-b), like (3-a), would presuppose (3-c), and (4-a) and (4-b) would both presuppose (4-c):

(3) a. The king of France is bald and France is a monarchy.
    b. France is a monarchy and* the king of France is bald.
    c. France has a king.

(4) a. France is a monarchy and the king of France is bald.
    b. The king of France is bald and* France is a monarchy.
    c. If France is a monarchy then France has a king.

Because *and** is deemed implausible, this suggests that Heim (1983) is vulnerable to criticism on the grounds that it is insufficiently constraining. This limitation is often regarded (e.g., by Schlenker, 2006, 2008a) as an explanatory weakness of the Heim (1983) account. Overgeneration is not an all-or-nothing matter, and nothing above shows that the framework of Heim (1983) is not *more* constraining than Karttunen and Peters (1979), only that it is not as constraining as we might hope.

How constraining an account is critically depends not just on the connectives it defines, but on the space of *possible* connectives from which those are understood as being drawn. When, in what follows, I describe one framework as more constraining than another, I mean that its space of possible connectives is more limited.

The passage from Heim (1983) above suggests another explanatory complaint: for Karttunen and Peters (1979), the projection properties and ordinary truth-functional meanings of a connective are treated as independent facets of its meaning – neither is derived from the other, nor are they both derived from some shared meaning. Heim's dynamic semantics purports to be more explanatory in that it replaces these two separate components with unified dynamic meanings (see, in particular, the first paragraph of Heim, 1983, Section 2.2). This notion of unified meaning is a relatively weak criterion of explanatory adequacy – in particular, it is also enjoyed by the Peters (1977) system, which treats the meaning of each connective as a single function.

Schlenker takes up these themes from Heim and Soames, and explicitly suggests another criterion, which is summarised as follows in Schlenker (2008b, p. 287):

> *Explanatory challenge*: Find an algorithm that predicts how any operator transmits presuppositions once its syntax and its classical semantics have been specified.

This challenge is met by transparency theory, as described by Schlenker (2006, 2008a), which derives presuppositions as conversational implicatures within a bivalent semantics, but not by the Heim-Beaver framework. Part of the goal of this paper is to show that Schlenker's explanatory challenge can, in large part, be met with a trivalent, semantic treatment of presuppositions, suggesting that, even if we accept Schlenker's explanatory challenge, it does not commit us to Schlenker's pragmatic treatment of presupposition (or to dynamic connective meanings).

I use the word 'explanatory' throughout this paper in the spirit of Heim and Schlenker. I do not wish to commit myself to any strong position on the relevance or irrelevance of these properties to explicating the general notion of explanation, but only note that they seem like desirable, or at least appealing, features for a theory of presupposition projection (for reasons discussed by Gazdar, Heim, Soames, and Schlenker, among others), for which I will continue to use this established terminology.[6]

1.3. Two projection systems

Below, I briefly sketch the Peters and Heim-Beaver projection systems. Sections 1.3.1 and 1.3.2 summarise these familiar systems (sometimes in a slightly modified format, suitable for our present purposes), and explicitly note certain established or obvious facts for later use. Finally, Section 1.3.3 offers a preliminary exploration of the explanatory properties of the two systems.

1.3.1. The Peters connectives

Peters (1977) adds to the customary 0 and 1 a third truth value, which I will write #, associated with presupposition failure. That is, a sentence can take any of the three truth values 0, 1, and #. A sentence's presuppositions are met by any circumstance in which its truth value is not # (that is, in which its truth value is either 0 or 1), and its assertion is true in any circumstance in which its truth value is 1.

To give the connectives suitable projection properties, Peters offers trivalent truth functions that extend the classical negation, conjunction, disjunction, and conditional operators, given below in Table 1. Each of these truth functions tells us, given the truth value of each of its arguments (truth, falsehood, or presupposition failure), what the truth value of the compound sentence will be. We can then determine the presuppositions of the compound sentence, seeing which circumstances will cause it to suffer presupposition failure (take the truth value #), and which will not.

If we take $\tilde{\neg}$, $\tilde{\wedge}$, $\tilde{\rightarrow}$, and $\tilde{\vee}$ as interpretations for English *not*, *and*, *if…then…*, and *or*, respectively, then they give us an account of presupposition projection. For example, $\tilde{\neg}x = \#$ iff $x = \#$, so if *not* is analysed with $\tilde{\neg}$, the prediction we derive is that a sentence and its negation will take the truth value # in the same circumstances, which is to say they will have the same presuppositions.

If we let W be our set of worlds, we can situate the Peters connectives within a possible worlds semantics, and give notions of proposition, presupposition, and assertion in terms of the three truth values. The following definitions give what I take to be a reasonable

Table 1. Peters's connectives.

x	y	$\tilde{\neg}$ x	x $\tilde{\rightarrow}$ y	x $\tilde{\wedge}$ y	$x\tilde{\vee}y$
1	1	0	1	1	1
1	0	0	0	0	1
1	#	0	#	#	1
0	1	1	1	0	1
0	0	1	1	0	0
0	#	1	1	0	#
#	1	#	#	#	#
#	0	#	#	#	#
#	#	#	#	#	#

formulation of the generally accepted versions of these notions in a trivalent semantics of presupposition.

First a proposition is a function from worlds to truth values. Now that there are three truth values, a proposition differentiates between three cases: truth, falsehood, and presupposition failure.

Definition 1 *A* trivalent proposition *is any function in* $\{0, 1, \#\}^W$ *(that is, any function from the set W of possible worlds into the set* $\{0, 1, \#\}$*). The set of all trivalent propositions is* $P_{\{0,1,\#\}} = \{0, 1, \#\}^W$.

A trivalent proposition has both a normal truth-conditional or assertive component, and a presuppositional component. A proposition suffers presupposition failure whenever it has the truth value #, so its presupposition is simply the set of worlds under which it does not take the truth value #, while the assertive component is, as in a bivalent system, simply the set of worlds in which the proposition is true.

Definition 2 *For any trivalent proposition* $p \in P_{\{0,1,\#\}}$*, the* presupposition of p *is the set* $\{w \in W \mid p(w) \neq \#\}$.

Definition 3 *For any trivalent proposition* $p \in P_{\{0,1,\#\}}$*, the* assertion of p *is the set* $\{w \in W \mid p(w) = 1\}$.

As an example, let $p \in P_{\{0,1,\#\}}$ be the proposition expressed by (5):

(5) Ada knows that a unicorn stole her martini.

For all $w \in W$, $p(w) = 1$ iff in w Ada's martini was stolen by a unicorn and Ada knows that it was, $p(w) = 0$ iff in w Ada's martini was stolen by a unicorn but Ada does not know that it was, and $p(w) = \#$ iff it is not the case that a unicorn stole Ada's martini. Thus, the presupposition of p is the set of worlds $\{w \in W \mid$ a unicorn stole Ada's martini in $w\}$, and the assertion of p is the set of worlds $\{w \in W \mid$ in w, Ada's martini was stolen by a unicorn and Ada knows that it was$\}$.

To see how these connectives derive projection effects, consider $\tilde{\neg}p$ (i.e., the function that maps every $w \in W$ to $\tilde{\neg}(p(w))$). $\tilde{\neg}p(w) = \#$ iff $p(w) = \#$, because in general $\tilde{\neg}(x) = \#$ iff $x = \#$, so we derive that $\tilde{\neg}p$ and p have the same conditions of failure, and hence the same presuppositions. By analysing natural language negation with $\tilde{\neg}$ (and analysing natural language conjunction, disjunction, and conditionals with $\tilde{\wedge}$, $\tilde{\vee}$, and $\tilde{\rightarrow}$), Peters (1977) provides an account of presupposition projection under propositional connectives.

1.3.2. The Heim-Beaver system

In this section, I present Heim's semantics in a way that borrows heavily from Schlenker (2006). Everything here is a straightforward restatement of the propositional fragment system in Heim (1983) (augmented with Beaver's disjunction), adjusted in matters of presentation in order to aid the comparison with trivalent truth-functional accounts in later sections.

The basic concept of declarative sentence meaning in Heim (1983) is *context change potential*. That is, the meaning of a sentence is its capacity to change the state of information in discourse. A context is, roughly, the sum total of facts already established in discourse, so, like most such bodies of information, it is formalised as a set of worlds. Sometimes, when evaluating a sentence, we can find ourselves unable to derive a new context. Following the presentation in Schlenker (2006), let us represent this by treating the error value # as a special context. Our notion of context is then captured by the following definition:

Definition 4 *A* context *is any member of* $\mathcal{P}(W) \cup \{\#\}$. *(That is, a context is a set of worlds, or the error value #.)*

The semantic value of a sentence, for Heim (1983), is its potential to update the context in which it is uttered, producing a new context. Thus, the semantic value of a sentence can be regarded as a function from contexts to contexts – hereafter an *update*.

Definition 5 *An* update *is any function from contexts to contexts (i.e., any member of* $\mathcal{P}(W) \cup \{\#\}^{\mathcal{P}(W)\cup\{\#\}}$*).*

In keeping with the tradition of writing updates to the right of contexts, for any update u and any context C, the update notation $C[u]$ will be understood as equivalent to the function-application notation $u(C)$.

Updating with a nontrivial context with an atomic sentence can be thought of in two steps. First, for the sentence to be admissible, the context must entail whatever is presupposed by the sentence. Second, if the sentence is admissible, its assertion is added to the stock of information in the context. This is, in typical possible-worlds style, done by intersection of world sets.

Definition 6 *The* Heim-style update *for atomic sentence* φ *is the unique* u_φ *such that, for every context* C*:*

- *If* $C = \#$ *then* $C[u_\varphi] = \#$.
- *If there is* $w \in C$ *s.t.* w *does not satisfy the presuppositions of* φ*, then* $C[u_\varphi] = \#$.
- *In all other cases* $C[u_\varphi] = \{w \in C \mid w \text{ satisfies the assertions of } \varphi\}$.

This formulation assumes that presuppositions of atomic sentences are properties of worlds, and that checking presuppositions involves checking to see if the context entails that the world has the presupposed property. This seems to me to be in the spirit of the way atomic sentences are treated by Heim (1983) and many others, and with the conception of presuppositions as facts taken as given. Still, it is worth noting that this is in no way an essential feature of the Heim-style projection system – if there are atomic sentences that do not have updates of this form, then matters get more complicated in ways that I will not explore here.

Let us now turn to the connectives. Heim (1983) defines the result of updating a context with a compound sentence in terms of the update effects of its parts and some basic set-theoretic machinery. For example, Heim's analysis of *and* states that evaluating the compound sentence φ *and* ψ in context C amounts to performing the sequence of updates $C[u_\varphi][u_\psi]$ (that is, conjunction is just updating with the first conjunct and then the second conjunct), and the analysis of negation states that evaluating *not* φ in C yields the new context $C \setminus C[u_\varphi]$ (that is, negation involves excluding as possibilities all worlds that would satisfy the negated sentence).

To have a notation more readily comparable with a trivalent truth-functional system, I want to be able to speak of connectives as operations on updates more directly, without explicit mention of contexts. In order to do this, I will present Heim's connectives in a way that makes use of a null update ε, which maps every context to itself, along with the notion of sequencing of updates (function-composition) and set-theoretic operations on updates (understood in terms of the corresponding set-theoretic operations on contexts).

DEFINITION 7 *For all contexts C and all updates u and u'…*

- $C[\varepsilon] = C$. *(Null update.)*
- $C[u \setminus u'] = C[u] \setminus C[u']$ *if neither* $C[u]$ *nor* $C[u']$ *is #, otherwise #. (Difference of updates.)*
- $C[u; u'] = C[u][u']$. *(Sequencing of updates.)*
- $C[u \cup u'] = C[u] \cup C[u']$ *if neither* $C[u]$ *nor* $C[u']$ *is #, otherwise #. (Union of updates.)*
- $C[u \cap u'] = C[u] \cap C[u']$ *if neither* $C[u]$ *nor* $C[u']$ *is #, otherwise #. (Intersection of updates.)*

Difference and sequencing, along with the null update, will be used for Heim's connectives. Union and intersection are included because they seem to belong to the same natural class of operations as difference, and because the former is used in some formulations of dynamic disjunction (cf. Schlenker, 2006). The operations in Definition 7 can be thought of as a basic toolbox for defining operations on updates, which serves as a natural generalisation of the toolbox employed in Heim (1983).

Heim (1983) gives definitions of negation, conjunction, and the conditional. Below, I restate Heim's definitions in terms of the operations from Definition 7:

DEFINITION 8 *For all updates u and u'…*

- $\mathbf{not}\ u = \varepsilon \setminus u$.
- $u\ \mathbf{and}\ u' = u; u'$.
- $\mathbf{if}\ u, u' = \varepsilon \setminus (u \setminus (u; u'))$.

Beaver (2001) adds a disjunction intended to match standard descriptive claims about projection, and equivalent to, for example, the disjunction of Schlenker (2008a, 2006), Karttunen (1973, 1974), and Peters (1977).

DEFINITION 9 *For all updates u and u',*

- $u\ \mathbf{or}\ u' = \mathbf{not}\ ((\mathbf{not}\ u)\ \mathbf{and}\ (\mathbf{not}\ u'))$.

1.3.3. Explanatory characteristics of the two systems

Having reviewed implementations of the Heim-Beaver and Peters systems, let us take a moment to look at their explanatory characteristics. The explanatory qualities of a system (at least as understood in terms of constrainingness and of Schlenker's explanatory challenge) are determined not just by the meanings it employs, but by the space of possibilities from which it draws its meanings. Since neither Peters (1977) nor Heim (1983) is explicit about what space of meanings it presumes, we can assess these properties only given some assumptions about the space of alternatives. Here, I'll start with the least charitable such assumptions for each system, and then discuss some more constraining approaches. Most of my observations about the Heim-Beaver system are reformulations or minor extensions of points made elsewhere (especially by Schlenker, 2006, 2008a,b), but the discussion of the Peters system, although in many cases obvious, does not, as far as I know, restate previous observations from the literature, except for the brief discussion of Krahmer's 'middle Kleene' insight, which is anticipated by Krahmer (1998), and is to a large extent implicit in Peters (1977).

Peters (1977) identifies connective meanings with trivalent truth functions. If we understand the Peters connectives as belonging to a framework in which *any* trivalent truth function is a possible connective meaning then, trivially, we get a system that does not meet Schlenker's explanatory challenge, since the lexical meaning encodes non-bivalent material (the five new lines of the trivalent truth table), and is free to make different choices of non-bivalent material in choosing an extension of a given classical connective.

This space of possible meanings is modestly constraining. From a point of view of cardinality, it limits us to $3^9 = 19,683$ possible connectives, and for any given classical binary connective, there will be $3^5 = 243$ ways of extending it by 'filling in' the new lines of the truth table. If we assume other reasonable constraints on natural trivalent truth functions, we can lower the number further.[7] These numbers are larger than we might like, and in particular compare unfavourably to any theory that meets Schlenker's explanatory challenge: such a theory will allow only $2^4 = 16$ possible binary connectives, and only 1 presuppositional behaviour for each classical binary connective. Still, even in this trivial choice of framework, the number of possibilities has the virtue of being finite.

Counting possible connectives is not the only way to assess the constrainingness. One interesting constraint imposed by the use of trivalent truth functions is that they make presupposition projection *extensional*, in the sense that the presuppositions of compound sentences will be determined on a world-by-world basis. If $w \neq w'$ then, under the Peters approach of interpreting every (binary) connective $\star$ with a trivalent truth function, the question of whether w satisfies the presuppositions of $\varphi \star \psi$ can only depend on whether φ and ψ are true, false, or suffer presupposition failure in w, and the status of the assertions and presuppositions of φ and ψ at w' will not enter into the matter (provided that we assume that φ and ψ express trivalent propositions, which is to say that their presuppositions can be understood as sets of worlds).

Extensionality is a substantial constraint that denies us the power to assign arbitrary projection behaviours to connectives. As a simple example of a formally possible non-extensional unary operator, let $\ddagger$ be the operation on a trivalent proposition p such that $\ddagger(p)(w) = 0$ if there is any world w' such that $p(w') = \#$, and $\ddagger(p)(w) = p(w)$ otherwise. '$\ddagger p$' can be read as 'p has no presuppositions, and also p'.[8] We will see other examples later on, but, critically, this distinguishes the class of trivalent truth functions from the class of arbitrary operations on updates: it is perfectly straightforward to define $\ddagger$ as an operation on updates.[9]

The above observations pertain to the framework that takes the possible connective meanings to be exactly the trivalent truth functions. But certain patterns, and especially certain order asymmetries, in Peters (1977) suggest further constraints. Krahmer (1998) calls the Peters connectives the 'middle Kleene' connectives, because of their resemblance to the weak Kleene and strong Kleene trivalent logics of Kleene (1952), in their first and second argument positions, respectively. Understanding the Peters connectives in the way that Krahmer suggests turns out to allow us to derive the Peters connectives in a way that meets Schlenker's explanatory challenge. I will return to this in Section 3.

What about the Heim-Beaver system? If, uncharitably, we take any operation on updates as a possible connective meaning, then we have a framework that is notably less constraining than the trivalent truth-functional framework, and which does not meet Schlenker's explanatory challenge.[10] Furthermore, the space of possible functions of this kind is not finite (at least if there are infinitely many worlds), and need not support the extensionality property we noted for trivalent truth functions, so, given this space of meanings, the operations-on-updates framework is significantly less constraining than the trivalent truth-functional framework.

The above assessment is formally consistent with Heim (1983), but it is insufficiently charitable to the common format of Heim's connectives. All of Heim's connectives can be built using only two operations: $\backslash$ and $;$. In the next section we will see that limiting ourselves to these (or even adding $\cup$ and $\cap$) yields a framework that is at least as constraining as an account which draws its meanings from arbitrary trivalent truth functions. Thus, although the Heim-Beaver system does not meet Schlenker's explanatory challenge, it can still plausibly be said to be reasonably constraining.

2. The Heim-Beaver dynamic system as a trivalent truth-functional system

In this section, I want to explore the relationship between the Heim-Beaver toolbox of ε, $;$, $\backslash$, $\cup$, and $\cap$ and systems based on trivalent truth functions. The main result will be that, if we begin with atomic updates that correspond with trivalent propositions (i.e., that are *trivalentisable*), this toolbox only allows us to define operations that correspond to trivalent truth functions. This is a substantial and, as far as I know, new constraining result for this toolbox. It is widely known that the propositional fragment of the Heim-Beaver system is equivalent to the Peters system, but what is shown here is that the pieces from which the Heim-Beaver connectives are built can define only connectives that correspond to trivalent connectives. This implies that the Heim-Beaver framework, understood in this way, is at least as constraining as the framework that allows arbitrary trivalent truth functions as connectives, and in particular that it inherits any constraints that can be proved for this framework, including the extensionality constraint discussed above.

2.1. Commensurability with trivalent truth functions

In order to compare the Heim-Beaver system with one based on trivalent truth functions, I will need some (more or less trivial) definitions and observations.

To begin, it will be necessary to be explicit with regard to the intended dynamic role of static trivalent propositions. The dynamic effect of a trivalent proposition should be first to check that the context satisfies its presuppositions, and then, if it does, to update the context with its assertions. In more formal terms, I define the update for a static trivalent proposition as follows:

Definition 10 *For any trivalent proposition p, the update for p, written $\mathtt{u}_p$ is the unique update with the following properties:*

- *If $C = \#$, or if there is $w \in C$ s.t. $p(w) = \#$, then $C[\mathtt{u}_p] = \#$.*
- *Otherwise, $C[\mathtt{u}_p] = \{w \mid w \in C \text{ and } p(w) = 1\}$.*

This is just a more formal restatement of the ideas that a presupposition is what must be satisfied by the common ground in order to evaluate the assertion, and acceptance of an assertion involves adding the information it provides to the common ground.

The above definition has the following important formal feature:

Remark 1 The 'update for' function (i.e., the function that maps p to $\mathtt{u}_p$) is one-to-one.

We will, in particular, be concerned with those updates that correspond to static trivalent propositions (i.e., that are *trivalentisable*):

DEFINITION 11 *An update u is* trivalentisable *iff there is a trivalent proposition p s.t. $u = \mathtt{u}_p$.*

Trivalentisability may also be formulated as follows:

Remark 2 An update u is trivalentisable iff there are two sets of worlds u^π (the presupposition of u) and u^α (the assertion of u) such that the following conditions hold:

- $u^\alpha \subseteq u^\pi$.
- $\#[u] = \#$.
- For $C \neq \#$, $C[u] = \#$ iff $C \nsubseteq u^\pi$.
- If $C[u] \neq \#$, $C[u] = C \cap u^\alpha$.

This brings us to some unsurprising points about the use of trivalent propositions and truth functions in a dynamic setting. Because we can give a general definition of $\mathtt{u}_p$ for arbitrary p, we can interpret any trivalent proposition dynamically, so we can employ a static theory of sentence meaning and still use the resulting propositions to do updates on contexts. Further, by Remark 1, we do not lose any ability to individuate propositions by converting them to updates, so, for every operation on trivalent propositions (and in particular for every trivalent truth function whatsoever, regardless of whether it bears any relation to the Peters connectives) there is a corresponding operation on updates.

The trivalentisable updates have a number of properties not shared by all functions on contexts, including the following:

Remark 3 If u is trivalentisable, then for all C, $C[u] = \#$ or $C[u] \subseteq C$.

Remark 4 If u is trivalentisable, $C, C' \neq \#$, $C[u] \neq \#$, and $C' \subseteq C$, then $C'[u] \neq \#$.

These remarks allow us to see some examples of non-trivalentisable but logically possible updates. Let V be any non-empty proper subset of the set of worlds W (which is assumed to contain at least two possible worlds), and observe that u_1 and u_2 defined below are not trivalentisable, by Remarks 3 and 4 respectively:

(6) a. For all contexts C, $C[u_1] = V$.
b. For all contexts C, if $C = \#$ or $C \cap V = \emptyset$, then $C[u_2] = \#$. Otherwise, $C[u_2] = C$.[11]

That is, no trivalentisable update implements the operation 'discard whatever context we had before, and start fresh with new context V', and none implements 'let us hereby presuppose that it is already consistent with the contextual information that we are in one of the V-worlds'. Other examples exist, but for now it suffices to observe that trivalentisable updates are a special, limited, class of updates, when compared with the full array of possible functions from contexts to contexts.

For succinctness, call the set of trivalentisable updates $U_{\pi,\alpha}$.

A little reflection shows that, given the Heim-Beaver system's treatment of atomic sentences, all atomic sentences will be associated with trivalentisable updates:

Remark 5 For every atomic sentence φ, the Heim update u_φ is trivalentisable, and in particular is equal to $\mathtt{u}_{p_\varphi}$ where p_φ is the unique trivalent proposition such that $p_\varphi(w) = \#$ if w does not satisfy the presupposition of φ, $p_\varphi(w) = 1$ if w satisfies the assertion (and presupposition) of φ, and $p_\varphi(w) = 0$ otherwise.

Note that this is not an essential feature of a Heim-style projection system, but is instead a feature of the way we treat presupposition-laden atomic sentences in terms of separate presuppositions (or admittance conditions) and assertions (or at-issue content), both understood as propositional. To the extent that the update associated with every atomic sentence updates a context by first checking to make sure it satisfies these presuppositions, and then by adding the assertions to the information content of the context, it is trivalentisable.

2.2. Isomorphism and related results

Above, I defined the Heim-Beaver connectives in terms of the basic operations of ε, ;, \, $\cup$, and $\cap$. I will now show that for every operation on updates in the Heim-Beaver toolbox, there is an equivalent trivalent truth function that can be applied to trivalent propositions. The proposed correspondence is summarised in Table 2.

That is, ε is equivalent to truth, $\cup$ and $\cap$ are equivalent to weak Kleene disjunction and conjunction ($\mathring{\vee}$ and $\mathring{\wedge}$), \ is equivalent to a weak Kleene trivalentisation of the natural truth-functional difference operation ($\mathring{-}$), and sequencing is equivalent to Peters's conjunction ($\tilde{\wedge}$). The last of these equivalences is a consequence of the widely recognised equivalence of the Heim and Peters connectives but, as far as I know, this type of equivalence has not been applied to characterising the constraining power of a Heim-style framework.[12]

The key equivalence result is that $U_{\pi,\alpha}$ (the set of trivalentisable updates) under the operations ε, ;, \, $\cup$, and $\cap$ is isomorphic to $P_{\{0,1,\#\}}$ (the set of trivalent propositions) under the operations $\top$, $\tilde{\wedge}$, $\mathring{-}$, $\mathring{\vee}$, and $\mathring{\wedge}$,[13] under the 'update for' function $\mathtt{u}$, which can reasonably be understood as preserving at-issue and presuppositional meaning.

Here, at last, is the key isomorphism result:

PROPOSITION 1 *The 'update for' function* $\mathtt{u}$ *is an isomorphism between the structures* $(P_{\{0,1,\#\}}, \top, \tilde{\wedge}, \mathring{-}, \mathring{\vee}, \mathring{\wedge})$ *and* $(U_{\pi,\alpha}, \varepsilon, ;, \backslash, \cup, \cap)$.

To see that this is the case, first note that the 'update for' function $\mathtt{u}$ is surjective from $P_{\{0,1,\#\}}$ onto $U_{\pi,\alpha}$, and so (by Remark 1 above) is a bijection between $P_{\{0,1,\#\}}$ and $U_{\pi,\alpha}$. All

Table 2. Truth-functional equivalents of operations from the Heim-Beaver toolbox.

p	q	$\top$	$p\tilde{\wedge}q$	$p\mathring{-}q$	$p\mathring{\vee}q$	$p\mathring{\wedge}q$
equiv:		ε	;	\	$\cup$	$\cap$
1	1	1	1	0	1	1
1	0	1	0	1	1	0
1	#	1	#	#	#	#
0	1	1	0	0	1	0
0	0	1	0	0	0	0
0	#	1	0	#	#	#
#	1	1	#	#	#	#
#	0	1	#	#	#	#
#	#	1	#	#	#	#

that remains is to establish that $\mathtt{u}$ is a homomorphism. That is, that for all $p, q \in P_{\{0,1,\#\}}$, the following equalities hold:

- $\mathtt{u}_{\top} = \varepsilon$.
- $\mathtt{u}_{p \tilde{\wedge} q} = \mathtt{u}_p ; \mathtt{u}_q$.
- $\mathtt{u}_{p \mathring{-} q} = \mathtt{u}_p \setminus \mathtt{u}_q$.
- $\mathtt{u}_{p \mathring{\vee} q} = \mathtt{u}_p \cup \mathtt{u}_q$.
- $\mathtt{u}_{p \mathring{\wedge} q} = \mathtt{u}_p \cap \mathtt{u}_q$.

The equivalences for ; and \ are given in Appendix A as examples – the reader may without too much difficulty supply the others.

The equivalences described in the discussion of the homomorphism property imply that trivalentisable updates are closed under all the operations of the Heim-Beaver toolbox:

Corollary 1 *$U_{\pi,\alpha}$ is closed under the operations of* ;, \, ∪, *and* ∩, *and further* $\varepsilon \in U_{\pi,\alpha}$.

When we consider that all atomic formulas are assigned trivalentisable updates, and that the Heim-Beaver connectives build the updates of complex sentences using only combinations of the operations listed in Corollary 1, the following is immediate:

Proposition 2 *Every update derived by the propositional fragment of the Heim-Beaver system, or by any assortment of propositional connectives defined in terms of* ε, ;, \, ∪, *and* ∩, *is trivalentisable (assuming atomic sentences are given only trivalentisable updates).*

Returning to Proposition 1, we have a number of interesting new observations. We find that *any* hypothetical dynamic connective we might build with ;, \, ε, ∪, and ∩ corresponds, under the 'update for' isomorphism, to a trivalent truth function. This is general result about possible connectives in this framework, and it helps us to see that the framework based on these operations on updates is substantially constraining. If we take the possible connective meanings to be exactly those which can be built with this toolbox, we get something at least as constraining as the framework that allows arbitrary trivalent truth functions to serve as connective meanings, and we inherit all the relevant bounds of the general trivalent truth-functional framework. These include the finite cardinality bounds on the number of possible connectives of a particular arity, and also more interesting constraints like the extensionality property.

Recall that the extensionality property says that every connective treats trivalent propositions on a world-by-world basis. That is, in the binary case, for any connective $\star$ and any sentences φ and ψ interpreted as trivalent propositions, the question of whether a world w satisfies the presuppositions of $\varphi \star \psi$ can only depend on whether φ and ψ are true, false, or suffer presupposition failure in w, and not on their truth values or presuppositional status with respect to any other world. In the truth-functional case, this is an immediate consequence of the fact that trivalent truth functions are functions over truth values, and not over propositions in any intensional or hyperintensional sense. The isomorphism result shows us that connectives built with the Heim-Beaver toolbox have this property as well.

This means that, although the Heim-Beaver toolbox does not meet Schlenker's explanatory challenge, it is constraining in an important way. Although it can define both the standard **and** and Soames's deviant **and***, it cannot define a non-extensional connective like **and**†:

(7) a. $C[u \textbf{ and } u'] = C[u][u']$.
b. $C[u \textbf{ and}^* u'] = C[u'][u]$.

c. $C[u\ \textbf{and}^{\dagger}\ u'] = C[u][u']$ if u and u' have exactly the same presuppositions,[14] and otherwise $C[u\ \textbf{and}^{\dagger}\ u'] = \#$.

Of course, $\textbf{and}^{\dagger}$ is just one example – what is important is that it represents a whole class of binary operations on updates that is excluded, and a class of ways of assigning a projection pattern to classical conjunction (or any other classical connective) that are not generated by the Heim-Beaver toolbox. Extensionality is one important constraining feature of the Heim-Beaver toolbox, but it is, presumably, not the only one.

Before moving on, let us note one more minor consequence of the isomorphism result: it makes possible a new kind of proof of the (widely known) equivalence of the Heim-Beaver and Peters connectives. If we simply take the isomorphic images of the operations used to build the Heim-Beaver connectives, we can verify that the resulting trivalent truth functions are the Peters connectives confirming the equivalence.[15]

As long as attention is restricted to trivalentisable updates, we can use the Peters connectives on trivalent propositions and then interpret those trivalent propositions as updates, and derive the same updates as we would using the Heim-Beaver connectives. This observation is not new, but the isomorphism result gives us the tools necessary to prove it by a straightforward comparison of truth tables, and it also calls attention to the possibility of using the Peters connectives to study the logical properties of the Heim-Beaver connectives, for example, by proving logical equivalences for the Peters connectives, and then inferring that the same equivalences hold for the Heim-Beaver connectives.

3. The middle Kleene intuition and Schlenker's explanatory challenge

Understood as arbitrarily chosen trivalent truth functions, the Peters connectives do not meet Schlenker's explanatory challenge: nothing we have said so far accounts for why *and* should be analysed as $\tilde{\wedge}$, rather than any of the other 242 trivalent truth functions that agree with classical $\wedge$ when given arguments in $\{0, 1\}$.[16] Thus, the Peters system, like the Heim-Beaver system, is arguably at an explanatory disadvantage when compared with a system like Transparency Theory (Schlenker, 2008a, 2006), which meets this challenge, and is also more constraining.

Here, I derive the Peters connectives in a manner consistent with Schlenker's explanatory challenge by giving a uniform (and, I hope, intuitively appealing) rule for reading trivalent functional behaviour from a classical connective meaning. The rule that I will employ, which I call *Peters-Kleene function deployment*, exploits Krahmer's 'middle Kleene' intuition, combining the intuition behind the strong Kleene logic (cf. Kleene, 1952; Beaver and Krahmer, 2001) with a left-to-right incremental evaluation strategy. With this new, fully explicit implementation of these familiar intuitions, we can show that the Peters connectives can be derived in a way compatible with Schlenker's explanatory challenge.

The plan is to define $f[\vec{x}]$ the Peters-Kleene deployment of a function f on an argument-sequence $\vec{x}$, and then show the following equivalences, where $f_{\neg}$, $f_{\wedge}$, $f_{\vee}$, and $f_{\rightarrow}$ are the functions classically associated with the subscripted connectives:

(8) For all $x, y \in \{0, 1, \#\}$:

- $f_{\neg}[x] = \tilde{\neg} x$.
- $f_{\wedge}[x, y] = x \tilde{\wedge} y$.
- $f_{\vee}[x, y] = x \tilde{\vee} y$.
- $f_{\rightarrow}[x, y] = x \tilde{\rightarrow} y$.

The shape of the proposal is similar to that of transparency theory: the distinctive (arbitrary/lexical/learned) meaning of each connective is simply a bivalent truth function. The Peters-Kleene deployment rule is supposed to be a universal principle of how these functions are to be combined with arguments drawn from the set {0, 1, #}, which, unless some further semantic mechanisms are postulated, will fully determine how each connective projects presuppositions, leaving no freedom to employ other projection patterns. The arguments are assumed to be incrementally evaluated in some canonical order that is determined by the syntax. The exact notion of order involved requires further exploration, but here I will assume that the order involved is linear order of the arguments at the level of logical form.[17] For brevity, in the remainder of the paper I will simply speak of linear or left-to-right order, without delving further into the details.

Because Peters-Kleene deployment meets Schlenker's explanatory challenge, it shows that this challenge can be met within a static account that treats presuppositions as a semantic phenomenon, indicating that considerations related to Schlenker's challenge do not by themselves determine whether a semantic or pragmatic approach to presupposition is preferable.

3.1. *Peters-Kleene deployment*

This section sketches the idea of Peters-Kleene asymmetrical function deployment. The intuition, expressed with a function-as-agent metaphor, is as follows: the meaning of every connective is a classical truth function. This function combines with multiple arguments by taking these arguments one at a time, in a canonical left-to-right order. When the argument is 1 or 0, the function takes in that argument and moves on to the next. When the argument is #, there are two options: if the function has enough information, on the basis of prior arguments, to determine that the current argument position 'doesn't matter', then it simply ignores the # and proceeds to the next argument (if there are further arguments). Otherwise, the function does not know what to do, and it resolves to return presupposition failure value #. This incremental approach accounts for the order asymmetry seen in the Peters truth functions, and the notion of an argument position 'not mattering' will receive a kind of strong Kleene implementation.

As an example of how this will work, $\wedge$ is the meaning that, combined with the principle, gives us the behaviour of $\tilde{\wedge}$. The reason that $0\tilde{\wedge}\# = 0$ is that when we know the first argument of $\wedge$ is 0, we do not (classically) need to inspect the second argument to know that the truth value of the conjunction is 0.[18] This is our notion of the second argument position not mattering. This assumes that truth functions approach things in bivalent terms, so $\wedge$ determines that the second argument after 0 will always yield an output of 0 by considering 0 and 1 as possible second arguments, but not considering the unfamiliar #. In contrast with $0\tilde{\wedge}\#$ above, $1\tilde{\wedge}\# = \#$ because when the first argument of $\wedge$ is 1, the second argument *does* matter: $1 \wedge 0 \neq 1 \wedge 1$, so $\wedge$ cannot compensate for an unexpected # by ignoring its second argument. The reason that $\#\tilde{\wedge}0 = \#$ is that, going from left to right, when $\wedge$ sees the first # it does not yet 'know' that the next argument is 0, so being aware of the possibility that the next argument could be 1, it concludes that the first argument position *does* matter, and thus that the # cannot be ignored, with the result of presupposition failure for the whole conjunction. The remainder of this section is an attempt to give a more rigorous form to these ideas, without relying on the imprecise but intuitively useful personification of such set-theoretic objects as functions.

In what follows, I understand an n-ary (truth) function to be a function defined over a domain of n-ary sequences of (truth) values. The trivial case of this will be a 0-ary function, defined over sequences of length 0 – since there is only one sequence of length 0, 0-ary functions will have exactly one item in their image, and so in what follows I will talk as if each truth value simply *is* the 0-ary function that maps the empty sequence onto it.

To implement incrementality, it will be useful to have a way to combine an n-ary function with an argument to fill its first argument position, yielding an $(n-1)$-ary function. This is achieved by the following definition.

Definition 12 *For any $n \geq 1$, any n-ary function f with domain $\{0, 1\}$, and any $x \in \{0, 1\}$, the* reduction of f by x *(written f/x) is the unique $(n-1)$-ary truth function such that for any $(n-1)$-ary sequence $\vec{y}$ of classical truth values, $(f/x)(\vec{y}) = f(x, \vec{y})$.*

Given the understanding of 0-ary functions described above, we have the following:

Remark 6 For any unary truth function f, and any $x \in \{0, 1\}$, $f/x = f(x)$.

This reduction operation can be thought of as an implementation of Currying that ensures that the arguments are fed into the function from left to right. It allows us to give the truth functions a traditional flat presentation, but still feed in arguments incrementally according to linear order.

The definition above restricts attention to truth functions for simplicity of exposition, but is readily generalised to functions of any type, and will be used for other types in the discussion of quantifiers in Section 4.

I next define a rule for attempting to reduce a function by a set of values, representing indeterminacy between alternative argument values. The idea is that reduction of a function by a set of alternatives succeeds if giving the function any member of the set of alternatives will yield the same result (in which case we reduce by an arbitrary member of the set of alternatives), and fails otherwise (in which case presupposition failure occurs).

Definition 13 *For any n-ary truth function f defined over the classical truth values, and any non-empty set X of classical truth values, the* Peters-Kleene reduction of f by X *(written $f///X$), is defined as follows:*

- *If there is a function g such that for all $x \in X$, $f/x = g$, then $f///X = g$.*
- *Otherwise,*[19] *$f///X$ is the $(n-1)$-ary function that maps all sequences of classical truth values to the failure value #.*

When we have a singleton set, this corresponds to ordinary reduction by the sole member of the set:

Remark 7 For all truth functions f and all $x \in \{0, 1\}$, $f///\{x\} = f/x$.

The idea here is that sets containing more than one value represent a kind of indeterminacy between possible values, used to test that the exact value chosen 'doesn't matter'. Thus, we get a usable function only if all the different values under consideration would give us the same result. If we have values that would produce different outcomes, we give up and return the failure value #. The intuition here is similar to the strong Kleene intuition, except that it is carried out looking at only the first argument in the sequence. This is the source of all the asymmetry in the theory: because we handle argument positions one at a time, we address uncertainty with the prior arguments 'locked in', but we must have a single new function after the reduction, meaning we must consider all possible (bivalent) values for the arguments not yet seen. Since f is a truth function defined only on classical truth values, g will be as well: since our functions are not defined for non-bivalent arguments, the notion of possible later arguments implicit in this definition only considers sequences of classical truth values.

Like the definition of reduction, the definition of Peters-Kleene reduction may be straightforwardly generalised beyond truth functions.

Reduction was our source of incrementality, and Peters-Kleene reduction generalised reduction to sets of possibilities. As a final ingredient, we need to associate trivalence with a set of possibilities: the *repair set* of that truth value. The repair set of every truth value in $\{0, 1, \#\}$ is a non-empty set of classical truth values. The idea is that presupposition failure (the truth value #) is treated by the semantics as an inability to resolve a classical truth value, so that the repair set of # contains both 0 and 1. The significance of this will become clearer in the context of the definition of Peters-Kleene deployment below.

Definition 14 *For any truth value x, the* repair set of x *(written $\mathcal{REP}(x)$) is defined as follows: if $x \in \{0, 1\}$ then $\mathcal{REP}(x) = \{x\}$, and if $x = \#$ then $\mathcal{REP}(x) = \{0, 1\}$.*

I will return to the possibility of generalising $\mathcal{REP}()$ to other types in Section 4.

I can now define Peters-Kleene deployment. Deploying a truth function at a particular truth value means Peters-Kleene reducing the function by that truth value's repair set; for classical truth values, this will boil down to ordinary reduction, but for #, this results in irreparable failure unless reduction by 0 would yield the same results as reduction by 1 – that is # is tolerable only in argument positions whose values are irrelevant.

Definition 15 *For any f a function defined over n-ary sequences of classical truth values and x a (possibly non-classical) truth value (that is $x \in \{0, 1, \#\}$), the* Peters-Kleene deployment of f on x*, written $f[x]$, is $f /\!/\!/ \mathcal{REP}(x)$.*

Peters-Kleene deployment combines a function with a single argument, but it is often useful to speak of giving a function multiple arguments. For this reason, I adopt the notational convention of writing $f[x_1, \ldots, x_n]$ for $f[x_1] \ldots [x_n]$.[20] Peters-Kleene deployment is readily generalised to other types of arguments, provided that we can find a suitable generalisation of the notion of repair set.

Connectives are, as a matter of their lexical meaning, functions that are concerned only with classical truth values (and something analogous presumably holds true for functions of other types), and the architecture of the grammar demands that they always combine with their arguments by left-to-right Peters-Kleene deployment. That is, for any clause ψ derived by combining an n-ary connective c (associated with an n-ary bivalent truth function f_c), and n syntactic arguments $\varphi_1, \ldots, \varphi_n$, ordered by linear precedence (with truth value (drawn from $\{0, 1, \#\}$) $v_{\varphi_1}, \ldots, v_{\varphi_n}$, respectively), the truth value of ψ is $f_c[v_{\varphi_1}, \ldots, v_{\varphi_n}]$.

Because Peters-Kleene deployment gives us a way to combine a classical truth function with arbitrary trivalent values, and because it has long been known how to understand trivalent truth functions as generating projection facts, Peters-Kleene deployment meets Schlenker's explanatory challenge. It is, as a result, more constraining than a hypothetical framework that allows arbitrary trivalent truth functions.

3.2. Peters connectives derived by Peters-Kleene deployment

I now confirm that Peters-Kleene deployment derives the Peters connectives from the classical connectives. The full proof of this is a somewhat laborious exploration of the possible combinations of truth values. I present a natural grouping of the main observations here, and refer the reader to Appendix B for representative proofs.

Let us begin by noting the obvious for the purely bivalent case.

Remark 8 If f and g are unary and binary truth functions, respectively, then for any $x, y \in \{0, 1\}$, $f[x] = f(x)$ and $g[x, y] = g(x, y)$.

Remark 8 says that Peters-Kleene deployment gives us the accustomed results for the four classical lines of the trivalent truth table. We now need to handle the five new lines where # comes into play.

First, all our classical truth functions, when Peters-Kleene is deployed on # as a first argument, result in presupposition failure:

Remark 9 $f_\neg[\#] = \#$.

Remark 10 For any truth value x, $f_\wedge[\#, x] = f_\vee[\#, x] = f_\rightarrow[\#, x] = \#$.

This fills in the bottom three lines of the truth table, so we have the facts reported in Table 3.

There are now only six cells of the truth table left to worry about.

Next, the following three facts can be confirmed by similar lines of reasoning:

Remark 11 $f_\wedge[0, \#] = 0$, $f_\rightarrow[0, \#] = 1$, and $f_\vee[1, \#] = 1$.

The following facts, which can again be proved in similar ways, yield the last three cells:

Remark 12 $f_\wedge[1, \#] = \#$, $f_\rightarrow[1, \#] = \#$, and $f_\vee[0, \#] = \#$.

Filling in the above, we now have all the cases, shown in Table 4.

Observe that values computed for Peters-Kleene deployment of the classical truth functions are identical to those stipulated for the Peters trivalent truth functions, thus:

Proposition 3 *For all* $x, y \in \{0, 1, \#\}$:

- $f_\neg[x] = \tilde{\neg}x$.
- $f_\wedge[x, y] = x \tilde{\wedge} y$.
- $f_\vee[x, y] = x \tilde{\vee} y$.
- $f_\rightarrow[x, y] = x \tilde{\rightarrow} y$.

That is, when the meanings of the connectives are taken to be the classical bivalent truth functions, we can derive trivalent truth-functional behaviour identical to the Peters connectives by adopting the general principle that a connective always combines with

Table 3. Derivations of the Peters connectives – results so far.

x	y	$f_\neg[x]$	$\tilde{\neg}x$	$f_\rightarrow[x, y]$	$x \tilde{\rightarrow} y$	$f_\wedge[x, y]$	$x \tilde{\wedge} y$	$f_\vee[x, y]$	$x \tilde{\vee} y$
1	1	0	0	1	1	1	1	1	1
1	0	0	0	0	0	0	0	1	1
1	#	0	0		#		#		1
0	1	1	1	1	1	0	0	1	1
0	0	1	1	1	1	0	0	0	0
0	#	1	1		1		0		#
#	1	#	#	#	#	#	#	#	#
#	0	#	#	#	#	#	#	#	#
#	#	#	#	#	#	#	#	#	#

Table 4. Equivalence of Peters connectives and Peters-Kleene deployment of classical connectives.

x	y	$f_{\neg}[x]$	$\tilde{\neg}x$	$f_{\rightarrow}[x, y]$	x $\tilde{\rightarrow}y$	$f_{\wedge}[x, y]$	$x\tilde{\wedge}y$	$f_{\vee}[x, y]$	$x\tilde{\vee}y$
1	1	0	0	1	1	1	1	1	1
1	0	0	0	0	0	0	0	1	1
1	#	0	0	#	#	#	#	1	1
0	1	1	1	1	1	0	0	1	1
0	0	1	1	1	1	0	0	0	0
0	#	1	1	1	1	0	0	#	#
#	1	#	#	#	#	#	#	#	#
#	0	#	#	#	#	#	#	#	#
#	#	#	#	#	#	#	#	#	#

its arguments by Peters-Kleene deployment, meaning that Peters-Kleene deployment, in addition to meeting Schlenker's explanatory challenge, delivers the standard projection facts, just as the Peters connectives do.

3.3. On the nature of Peters-Kleene deployment

From the point of view of Schlenker's challenge, what is important about Peters-Kleene deployment is that it is a general rule that tells us, for an arbitrary bivalent truth function f, how to apply f to a sequence of truth values from the set $\{0, 1, \#\}$, but this tells us nothing about why Peters-Kleene deployment in particular is used.

From one point of view, this is in keeping with typical semantic practice: we do not, for example, ask for a deeper explanation of why function application is typically the primary mode of semantic composition, but simply note its utility. As Peters-Kleene deployment is just a proposed enrichment of function application to allow us to apply bivalent truth functions to trivalent inputs, it is not obvious why it would need more elaboration.

Still, Peters-Kleene deployment *is* based on two potentially appealing intuitions and these are worth spelling out. The first of these is that of ordered evaluation: information about earlier arguments of a function can inform the evaluation of later arguments, but not vice versa. This has the normal appeal of incrementality, and a similar principle is explicit in transparency theory, and implicit in the original formulation of the Peters connectives and in many dynamic accounts.

The second intuition, closely connected with the motivation of the strong Kleene logic, is that presupposition failure is a breakdown of the natural order of things (embodied by the classical connectives), and that nontrivial projection facts result from the tools the semantics uses to work around it. This is what lies behind the choice of the term *repair set*. The rule that $\mathcal{REP}(0) = \{0\}$ and $\mathcal{REP}(1) = \{1\}$ is our 'if it ain't broke, don't fix it' principle. The rule that $\mathcal{REP}(\#) = \{0, 1\}$ says that cases of # can be ignored, but only when they are in positions for which the truth value is irrelevant to the evaluation of the whole. Presupposition failure in the complex sentence occurs when it is impossible to ignore the subparts that involve presupposition failure and still derive a classical truth value.

Peters-Kleene deployment results from one reasonably simple way of formally implementing these intuitions, so I think I may reasonably claim that it is not *completely* arbitrary (for other implementations, see Fox, 2008 and the supervaluation-based approach discussed in Schlenker, 2008).

Having noted the above, I want to point that, as these ideas are presented here, their relevance to any account of semantic processing remains unclear. We would not want to say that hearers are computing Peters-Kleene deployment on the truth values of sentence parts as they process the sentence. This view would lack credibility for the simple reason that hearers often do not know the truth values of the clauses they are hearing.[21] We can probably do better by saying that hearers are doing Peters-Kleene deployment in parallel in all worlds (or in some finite collection of representative world-types), but this idea, although plausible in its outlines, would require further development. This kind of limitation, however, is hardly unique to Peters-Kleene deployment, but is shared by many formal semantic proposals.

4. Peters-Kleene deployment for quantifiers

I now turn to quantifiers, and introduce a natural way of generalising Peters-Kleene deployment to quantification. Specific quantifiers have been discussed already in strong Kleene and middle Kleene settings (cf. especially Krahmer, 1998), but here I discuss Peters-Kleene deployment for arbitrary determiner-type quantifiers. As far as nuclear scopes are concerned, the resulting trivalent behaviour converges with the 'supervaluation quantifiers' of van Eijck (1996). What is novel here is that, given a formal identification of the principle (Peters-Kleene deployment) behind the Peters connectives, we are able to see what kinds of projection it predicts for quantifiers, on a natural generalisation of the notion of repair sets. This yields some non-standard predictions (some more appealing, some less so), which give us a way to evaluate the basic Peters-Kleene approach against its competitors, and to search for evidence regarding what kinds of refinements and complications the approach might need.

Generalised Peters-Kleene deployment also makes it possible to offer general results about which quantifiers and classes of quantifiers give rise to which patterns of presupposition projection, something that has not, as far as I know, previously been explored for this type of trivalent projection system.

4.1. Introduction and examples

As has long been recognised, the strong Kleene approach can be used with the quantifiers $\forall$ and $\exists$ by treating them as forms of conjunction and disjunction, respectively. The idea behind strong Kleene conjunction (or disjunction) is that the complex sentence is true if *every* way of assigning classical truth values to the indeterminate (#) arguments of the connective would make it true, false if every way of assigning such truth values would make it false, and lacks a classical truth value (takes the truth value #) otherwise. This means that a strong Kleene conjunction sentence is true if all the conjuncts are true (because even a single conjunct with truth value # could be assigned 0 by a repair, meaning that the conjunction would not be true in all repairs), false if at least one of the conjuncts is false (because knowing that one conjunct is false is, in the classical case, enough to guarantee that the conjunction is false), and takes the truth value # otherwise (i.e., if none of the conjuncts is false, and at least one of them is undefined).

Informally, taking $\forall x$ as conjunction over possible values for x, we find that a formula of the form $\forall x \varphi(x)$ is true if all assignments of a value to x make $\varphi(x)$ true, is false if at least one assignment of a value to x makes $\varphi(x)$ false, and suffers presupposition failure in all other cases. If, for example, we analyse the English quantificational expression *everybody* as $\forall$, and say that x *has stopped smoking* is true if x is a former smoker (that is, used to smoke but no longer does), is false if x has smoked and persists in doing so (here abbreviated as 'x smokes'), and suffers presupposition failure otherwise (i.e., if x has never smoked), we derive that (9) is true under the conditions given by (10-a), is false under the conditions given by (10-b), and suffers presupposition failure under the circumstances given by (10-c).

(9) Everybody has stopped smoking.

(10) a. Everybody is a former smoker.
b. Somebody smokes.
c. Nobody smokes and somebody has never smoked.

Thus, what (9) presupposes is that (10-c) is not true, or, equivalently, that either (10-a) or (10-b) is true. This presupposition is given by (11):

(11) Somebody smokes or everybody is a former smoker.

Analogous reasoning applies to $\exists$, understood as disjunction. A classical disjunction is false iff all of its disjuncts are false, so $\exists x\varphi(x)$ will come out false, on the strong Kleene approach, iff $\varphi(x)$ comes out false for every possible value of x. A disjunction is true as long as at least one disjunct is true, so $\exists x\varphi(x)$ will be true as long as some assignment of a value to x makes $\varphi(x)$ true, even if $\varphi(x)$ suffers presupposition failure for some other values of x. In all other cases, we have presupposition failure. Analysing *somebody* as $\exists$, (12) will be true if (13-a) is true, false if (13-b) is true, and suffer presupposition failure if (13-c) is true, so it will presuppose exactly (14) (the disjunction of (13-a) and (13-b)).

(12) Somebody has stopped smoking.

(13) a. Somebody is a former smoker.
b. Everybody smokes.
c. Somebody does not smoke and not everybody is a former smoker.

(14) Somebody is a former smoker or everybody smokes.

These non-standard predictions will be discussed later, but, first, let us see how to implement this kind of reasoning in terms of Peters-Kleene deployment.

To deal with arbitrary natural language quantifiers using Peters-Kleene deployment as defined in the previous section, we will want to approach things using generalised quantifier theory (hereafter GQT). Here, quantifiers like $\exists$ and $\forall$ are understood as expressing properties of predicate-extensions (sets or characteristic functions of sets).[22] The reader is referred to, for example, the first few chapters of Peters and Westerståhl (2006) for a general overview of the main ideas of GQT. Framing this in functional terms, we get the following, corresponding to typical proposed extensions for *everything* and *something*:

Definition 16 *Given a universe U and a function $A \in \{0,1\}^U$, $\forall(A) = 1$ iff for all $x \in U$, $A(x) = 1$, and $\forall(A) = 0$ otherwise.*

Definition 17 *Given a universe U and a function $A \in \{0,1\}^U$, $\exists(A) = 0$ iff for all $x \in U$, $A(x) = 0$, and $\exists(A) = 1$ otherwise.*

That is $\forall$ is true of only properties that are themselves true of everything in the universe of discourse, while $\exists$ is true of any property that is true of something in the universe.

Quantificational natural language determiners like *some*, *each*, and *no* will, as usual, be treated as functions of two such arguments, so, for example, *no* will have as its extension the quantifier Q_{no}.

Definition 18 *Given a universe* U *and functions* $A, B \in \{0,1\}^U$, $Q_{no}(A,B) = 1$ *iff there is no* $x \in U$ *such that* $A(x) = B(x) = 1$, *and otherwise* $Q_{no}(A,B) = 0$.

That is, *no student laughed* is true iff there is nothing in both the extension of the first argument (*student*) and the extension of the second argument (*laughed*).

Having adopted GQT apparatus, we now need to generalise Peters-Kleene deployment to act on functions of this type. Peters-Kleene deployment was defined above only for truth functions, but the definition is readily generalised to functions for any kind of argument for which a $\mathcal{REP}()$ operation is defined. If we define $\mathcal{REP}()$ for other types, those definitions, combined with the Peters-Kleene deployment rule (adjusted in the obvious way), will give us predictions for presupposition projection for a wider range of functions.[23] If we wish to extend this approach to the quantifiers of the types discussed above, all we need to do is extend $\mathcal{REP}()$ to apply to functions from entities to truth values. For this case, and indeed for arbitrary functions into $\{0, 1, \#\}$, the most natural generalisation is to repair the output values of the function pointwise – that is, to say that the repairs of a function agree with the function for all arguments for which the function takes classical values, but can take any classical value for arguments where the function being repaired takes the failure value #. This is codified in the following definition:

Definition 19 *For all sets* A *and functions* $f \in \{0,1,\#\}^A$ *(that is,* f *has domain* A *and codomain* $\{0,1,\#\}$*),* $\mathcal{REP}(f) = \{g \in \{0,1\}^A \mid \text{for every } x \in A, g(x) \in \mathcal{REP}(f(x))\}$, *where, as before,* $\mathcal{REP}(\#) = \{0,1\}$, $\mathcal{REP}(0) = \{0\}$, *and* $\mathcal{REP}(1) = \{1\}$.[24]

So, for example, if $f \in \{0,1,\#\}^{\{a,b,c\}}$ is such that $f(a) = 0$ and $f(b) = f(c) = \#$, then $\mathcal{REP}(f) = \{g \in \{0,1\}^{\{a,b,c\}} \mid g(a) = 0\}$. That is, $\mathcal{REP}(f)$ will contain four functions, one for each choice of values for $f(b)$ and $f(c)$ from the set $\{0, 1\}$. The repairs of a function f are all the functions that agree with f where f is (classically) defined.

Peters-Kleene deployment is supposed to differ from the strong Kleene approach only in that it takes arguments one at a time. Thus, for quantifiers like $\forall$ and $\exists$ that take only one argument, it ought to replicate the results of the strong Kleene existential and universal quantifier that we discussed above. To get a feel for how this works, let us look at $\forall$.

The interesting case is when a quantifier (here $\forall$) takes as an argument a presupposition-laden predicate-extension. Such an extension will be a function from the universe U into $\{0, 1, \#\}$. For example, the extension of *has stopped smoking* will be such a function (mapping former smokers to 1, current smokers to 0, and everybody else to #), and its repairs will be all the functions in $\{0,1\}^U$ that map all former smokers to 1, and all current smokers to 0 (anything else in U may be either 0 or 1). This corresponds, in our previous discussion of the universal quantifier as conjunction, to the idea that when the scope of $\forall$ suffers presupposition failure at a particular value of the variable, we can repair the truth value of the scope at that value as either 0 or 1.

Now, let $A \in \{0,1,\#\}^U$ be the extension of *has stopped smoking*, and restrict attention to a case where U contains only people, so that (9) can be analysed in terms of $\forall$. Then, with Peters-Kleene deployment, to compute the truth value of (9) we evaluate $\forall /// \mathcal{REP}(A)$.

We have one argument: $\forall /// \mathcal{REP}(A) = 1$ iff for each $A' \in \mathcal{REP}(A)$, $\forall(A') = 1$. Likewise, $\forall /// \mathcal{REP}(A) = 0$ iff for each $A' \in \mathcal{REP}(A)$, $\forall(A') = 0$. In all other cases, $\forall /// \mathcal{REP}(A) = \#$. Now, if A maps a single element of U to # then it will have a repair A' that maps that element of U to 0, so that $\forall(A') = 0$. Likewise, if A maps any element of U to 0, then all of its repairs will map that element to 0. In either case, the condition for $\forall /// \mathcal{REP}(A) = 1$ is not satisfied. If, on the other hand, A maps every element of U to 1, then $\mathcal{REP}(A) = \{A\}$ and $\forall(A) = 1$, so $\forall /// \mathcal{REP}(A) = 1$. Thus, $\forall /// \mathcal{REP}(A) = 1$ iff A

maps every element of U to 1, which, where A is the extension of *has stopped smoking*, is the case in which everybody is a former smoker.

Now, when does $\forall///\mathcal{REP}(A) = 0$? This happens whenever there is $x \in U$ such that $A(x) = 0$, because then all $A' \in \mathcal{REP}(A)$ will be such that $A'(x) = 0$, which will mean that $\forall(A') = 0$. On the other hand, if there is no such x, then A maps every element of U to either 1 or #, so there will be an $A' \in \mathcal{REP}(A)$ that maps everything in U to 1, meaning $\forall(A') = 1$, so $\forall///\mathcal{REP}(A) \neq 0$. Thus, $\forall///\mathcal{REP}(A) = 0$ iff A maps at least one thing to 0. In the case where A is the extension of *has stopped smoking*, this will be the case in which at least one person has smoked and still does so.

Finally, we will have $\forall///\mathcal{REP}(A) = \#$ in all cases where neither of the above conditions is met. Thus, we get presupposition failure in any case where A maps at least one element of U to #, and maps no element of U to 0. And what is presupposed will be the disjunction of the two conditions above, that is, that either A maps everything to 1, or else maps something to 0. In the case of (9), this is our presupposition that somebody smokes or everybody is a former smoker.

As seen above, we recover exactly the semantics we did with our initial discussion of $\forall$. The reasoning for $\exists$ is analogous, and again recovers the observations made by understanding the existential quantifier as a kind of disjunction.

So far, we have just recovered the already familiar natural strong Kleene generalisations of the universal and existential quantifiers. What the specification of $\mathcal{REP}(A)$ for arbitrary A does is allow us to derive a Peters-Kleene generalisation of any generalised quantifier of a type that takes (characteristic functions of) sets as arguments. The results are, as far as nuclear scopes are concerned, largely the same as what van Eijck (1996) calls 'supervaluation quantifiers'.

This paper is concerned primarily with the formal issues, but, although empirical matters are not the main concern, we should note that this system derives some unorthodox predictions. In contrast with the propositional case, generalised Peters-Kleene deployment for quantifiers diverges from the standard projected presuppositions as reported by, for example, Heim (1983), on which both (9) and (12) have the universal presupposition (15):

(15) Everybody has smoked.

On the Peters-Kleene approach, neither (12) nor (9) presupposes (15), although (9) does entail it. I am not able to review the empirical advantages and disadvantages of these sorts of predictions in detail, but it is worth noting that the predicted presuppositions are, at least, not obviously absurd (for some relevant discussion, see, among others, Chemla, 2009; Krahmer, 1998; George, 2008c). In particular, Chemla (2009) finds a great deal of variation in the frequency of universal inferences for sentences quantifying over presuppositional predicates: it was found that almost all subjects found that a sentence corresponding to (15) was suggested by cases of *each* and *no* quantification, but only about half accepted it as suggested by existential (*at least 3…*) quantification.[25] Note that the main finding described is that (15) is suggested in some sense, which appears consistent with it being a presupposition, an ordinary entailment, an implicature, or some combination of these. In light of this, Chemla's data might be seen as indicating that universal quantification gives rise to inferences like (15) as at least entailments (and possibly presuppositions), while existential quantification gives rise to these inferences only as implicatures. The observation that these universal inferences are stronger for universal than for existential quantification mirrors the predictions of generalised Peters-Kleene deployment where they are derived as entailments (though not presuppositions) for universals but not existentials.[26]

Having said this, I will, again, be concerned here primarily with exploring the properties of this generalisation of Peters-Kleene deployment – such an exploration will, I hope, be of theoretical interest in its own right, and be valuable for its empirical assessment and for the development of principled revisions. One natural family of revisions, which I will not discuss here, involves more complex or context-dependent rules for computing repair sets. For a discussion of some of these, the reader is referred to George (2008a,c)

4.2. Taxonomy of quantifiers with universal inferences

Having given a few examples, I now want to turn to new results that offer more general characterisations of the sorts of quantification that give rise to certain kinds of universal inferences. I will restrict attention to permutation-invariant quantifiers of one argument (hereafter *perm-quantifiers*) and to the nuclear scope positions of permutation-invariant conservative determiners of two arguments (hereafter *pc-determiners*).[27] The permutation-invariance constraint implements one notion of topic-neutrality in generalised quantifier theory, so it limits us to a class of quantifiers including those with a relatively 'logical' character, and conservativity is plausibly a property of all natural language determiners (see Keenan and Stavi, 1986). I will further restrict attention to finite, non-empty universes.

I begin by introducing some terminology. I will say, as is standard, that a quantifier has a *universal presupposition* if it gives rise to a presupposition that everything in the domain over which it quantifies (the whole universe for a quantifier of one argument, the restrictor for a determiner of two arguments) has whatever property is presupposed by its (nuclear) scope. That is, where Q is a quantifier with a universal presupposition, *Q students have stopped smoking* will presuppose that every student smoked. We saw above that Peters-Kleene deployment predicts that some quantifiers do not have universal presuppositions, but do have the same inferences as entailments (for example *each student has stopped smoking* is predicted to entail, but not presuppose, that every student has smoked). I will say that a quantifier with this property has a *universal entailment* (quantifiers with universal presuppositions will trivially have universal entailments as well). Finally, for symmetry, say a quantifier has a *universal anti-entailment* when it has the other half of a universal presupposition – that is, when the would-be universal presupposition must be true in order for the corresponding quantified sentence to be false. In more formal detail, these notions are defined as follows:

Definition 20 *A one-argument quantifier Q over U has a* universal entailment *iff, for every $A \in \{0, 1, \#\}^U$, if $Q///\mathcal{REP}(A) = 1$, then $A(x) \neq \#$ for all $x \in U$. A two-argument quantifier D over U has a* universal entailment *iff, for every $B \in \{0, 1\}^U$ and every $A \in \{0, 1, \#\}^U$, if $Q///\mathcal{REP}(B)///\mathcal{REP}(A) = 1$ then, for all x s.t. $B(x) = 1$, $A(x) \neq \#$.*

Definition 21 *A one-argument quantifier Q over U has a* universal anti-entailment *iff, for every $A \in \{0, 1, \#\}^U$, if $Q///\mathcal{REP}(A) = 0$, then $A(x) \neq \#$ for all $x \in U$. A two-argument quantifier D over U has a* universal anti-entailment *iff, for every $B \in \{0, 1\}^U$ and every $A \in \{0, 1, \#\}^U$, if $Q///\mathcal{REP}(B)///\mathcal{REP}(A) = 0$ then, for all x s.t. $B(x) = 1$, $A(x) \neq \#$.*

Definition 22 *A perm-quantifier or pc-determiner has a* universal presupposition *if it has both a universal entailment and a universal anti-entailment.*

Let us adopt the convention that, for any non-empty universe U, and any $A \in \{0, 1\}^U$, $S_A = \{x \in U | A(x) = 1\}$. For any perm-quantifier Q over a finite, non-empty universe U and any $A, A' \in \{0, 1\}^U$ such that $|S_A| = |S_{A'}|$, $Q(A) = 1$ iff $Q(A') = 1$,[28] so we can adopt the common practice of talking about quantifiers of one argument being true of numbers (for representative presentations of this type of approach for quantifiers of one and especially two arguments, see for example, Peters and Westerståhl, 2006; van Benthem, 1986).

Definition 23 *For any perm-quantifier Q over a finite, non-empty universe U, and any $n \leq |U|$, Q* is true of *n iff $Q(A) = 1$ whenever $|S_A| = n$, and Q* is false of *n iff $Q(A) = 0$ whenever $|S_A| = n$.*

Note that, for every such n, Q will be either true of n or false of n.

Analogously, for any pc-determiner D over a finite, non-empty U, for all $A, B \in \{0, 1\}^U$, the truth value of $D(A, B)$ depends only on $|A - B|$ and $|A \cap B|$, so we can describe D as being true or false of pairs of numbers.

Definition 24 *For any pc-quantifier D over a finite, non-empty U and any m and n such that $m + n \leq |U|$, D* is true of *(m, n) iff $D(A, B) = 1$ whenever $|A - B| = m$ and $|A \cap B| = n$, and D* is false of *(m, n) iff $D(A, B) = 0$ whenever $|A - B| = m$ and $|A \cap B| = n$.*

Again, for every such (m, n) pair, D will be either true or false of (m, n).

Given this, we can give conditions for universal entailments, anti-entailments, and presuppositions in terms of what numbers a quantifier is true of. To begin, consider the perm-quantifiers with universal entailments.

Proposition 4 *For all perm-quantifiers Q on U, Q has a universal entailment iff there is no $n < |U|$ such that Q is true of both n and $n + 1$.*

Proof To see this, first suppose that there is such an n. Now, let A be any function in $\{0, 1, \#\}^U$ that maps n elements of U to 0, $|U| - (n + 1)$ elements of U to 1, and 1 element of U to #. Let x be the unique element of U such that $A(x) = \#$. Because there is only one such x, there are only two elements of $\mathcal{REP}(A)$. That is, $\mathcal{REP}(A) = \{A', A''\}$, where $A'(x) = 0$, $A''(x) = 1$, and for all $y \neq x$ $A(y) = A'(y) = A''(y)$. Now, A' maps n elements of 1, so $Q(A') = 1$, and A'' maps $n + 1$ elements of U to 1, so $Q(A'') = 1$. So we have $Q///\mathcal{REP}(A) = 1$, but there is $x \in U$ s.t. $A(x) = \#$, which is to say Q does not have a universal entailment.

On the other hand, suppose there is no such n, and consider any $A \in \{0, 1, \#\}^U$ such that there is at least one $x \in U$ for which $A(x) = \#$. Let $j = |\{x \in U \mid A(x) = 1\}|$, $k = |\{x \in U \mid A(x) = \#\}|$.

If there is no n' such that $j \leq n' \leq j + k$ and Q is true at n', then, for all $A' \in \mathcal{REP}(A)$, $Q(A') = 0$, so $Q///\mathcal{REP}(A) = 0$, hence $Q///\mathcal{REP}(A) \neq 1$. Meanwhile, if there is such an n', then there are two cases to consider: either $j \leq n' < j + k$ or $j < n' = j + k$.

If $j \leq n' < j + k$, then there is $A' \in \mathcal{REP}(A)$ that maps exactly n' elements of U to 1 (so $Q(A') = 1$, because Q is true of n'), and there will further be $A'' \in \mathcal{REP}(A)$ that maps exactly $n' + 1$ elements of U to 1 (so $Q(A'') = 0$, because by our supposition Q cannot be true of both n' and $n' + 1$). Because $Q(A') \neq Q(A'')$, $Q///\mathcal{REP}(A) = \# \neq 1$.

If $j < n' = j + k$, there are $A', A'' \in \mathcal{REP}(A)$ s.t. A' maps n' elements of U to 1, and A'' maps $n' - 1$ elements of U to 1, so, by reasoning analogous to the above $Q(A') = 1$ and $Q(A'') = 0$, so again $Q///\mathcal{REP}(A) = \# \neq 1$.

This exhausts all cases, so, whenever there is no n such that Q is true at both n and $n+1$, we find that Q has a universal entailment (in that, for all $A \in \{0, 1, \#\}^U$, if $A(x) = \#$ for any $x \in U$ then $Q///\mathcal{REP}(A) \neq 1$). ■

Analogous reasoning yields the following result for pc-determiners:

Proposition 5 *For any pc-determiner D, D has a universal entailment iff there are no m and n s.t. Q is true of both* (m, n) *and* $(m+1, n-1)$.

Reversing truth and falsehood everywhere, we further get the corresponding results for anti-entailments.

Proposition 6 *For all perm-quantifiers Q on U, Q has a universal anti-entailment iff there is no* $n < |U|$ *such that Q is false of both n and* $n+1$.

Proposition 7 *For any pc-determiner D, D has a universal anti-entailment iff there are no m and n s.t. Q is false of both* (m, n) *and* $(m+1, n-1)$.

Thus, perm-quantifiers with universal entailments will include ∀, ¬∃, *exactly three things*, *an odd number of things*, etc. and pc-determiners with universal entailments will include *every*, *no*, *exactly three*, and *an even number of*, etc.

In the special case of monotonicity, we can say more. Whenever Q is a positive-monotone perm-quantifier and $n+1 \leq |U|$, if Q is true at n, then Q is true at $n+1$. So, if a positive-monotone perm-quantifier has universal entailments and is true at n, it cannot be true at $n+1$, which is to say it must be that $n = |U|$, hence:

Remark 13 ∀ is (up to extensional equivalence) the only nontrivial positive-monotone perm-quantifier with a universal entailment.

The trivial quantifier $\mathbf{0}_U$ (defined $\mathbf{0}_U(A) = 0$ for all $A \in \{0, 1\}^U$) is the only other example of a positive-monotone perm-quantifier with a universal entailment.

By analogous reasoning, we can draw a similar conclusion for pc-determiners: they associate every restrictor argument with either a restricted universal quantifier or trivial quantification of the $\mathbf{0}_U$ sort.[29]

Remark 14 Every positive-monotone pc-determiner D with a universal entailment is one such that, for every $A \in \{0, 1\}^U$, either $D/A = \textit{each}/A$ or $D/A = \mathbf{0}_U$.

Similarly, for negative monotonicity, we find the following:

Remark 15 ¬∃ is the only nontrivial negative-monotone perm-quantifier with a universal entailment.

Remark 16 Every negative-monotone pc-determiner D with a universal entailment is one such that, for every $A \in \{0, 1\}^U$, either $D/A = \textit{no}/A$ or $D/A = \mathbf{0}_U$.

Similarly for anti-entailments:

Remark 17 ∃ is the only nontrivial positive-monotone perm-quantifier with a universal anti-entailment.

($\mathbf{1}_U$, defined $\mathbf{1}_U(A) = 1$ for all $A \in \{0, 1\}^U$, is the only other positive-monotone perm-quantifier with a universal anti-entailment.)

Remark 18 Every positive-monotone pc-determiner D with a universal anti-entailment is one such that for every $A \in \{0, 1\}^U$, either $D/A = \textit{some}/A$ or $D/A = \mathbf{1}_U$.

Remark 19 ¬∀ is the only nontrivial negative-monotone perm-quantifier with a universal anti-entailment.

Remark 20 Every negative-monotone pc-determiner D with a universal anti-entailment is one such that for every $A \in \{0, 1\}^U$, either $D/A = \textit{not every}/A$ or $D/A = \mathbf{1}_U$.

Putting together our observations about universal entailments and universal anti-entailments, we find that we get universal presuppositions only for those perm-quantifiers Q such that there is no n such that Q is true of both n and $n+1$ or Q is false of both n and $n+1$, thus:

Remark 21 The only perm-quantifiers with universal presuppositions are *an even number of things* and *an odd number of things*.

Remark 22 If any pc-determiner D has a universal presupposition, then, for every $A \in \{0, 1\}^U$, D/A is equivalent to either *an even number of*$/A$ or *an odd number of*$/A$.

The above facts rule out universal presuppositions for most linguistically interesting quantifiers. If this is deemed problematic then some revision will be needed. Looking at the above, one natural approach suggests itself: the determiners with the strongest intuitions in favour of universal presuppositions are *each/every/all* and *no/none*. Under Peters-Kleene deployment, these do not have universal presuppositions, but they do have universal entailments. One approach, then, would be to define a principled rule to strengthen universal entailments into universal presuppositions. Something along these lines is possible, and is advanced in George (2008b) and George (2008c, Appendix A), but I do not have the space to pursue the matter further here. There are, of course, numerous other ways of departing from this approach, at the expense of greater complexity. The exploration of the full range of possibilities is beyond the scope of this paper, which is chiefly concerned with formal aspects of the generalised Peters-Kleene approach. The results above are of interest because this approach serves as an appealing baseline, and because they give us a sense of its character and its limitations.

5. Concluding remarks

This paper has presented a number of formal results related to static, trivalent treatments of presupposition projection, and in particular to Peters-Kleene function deployment. We have seen that Peters-Kleene deployment, combined with classical meanings, offers a static account of semantic presuppositions that is reasonably constraining and meets Schlenker's explanatory challenge, while deriving predictions that coincide with the standard projection claims for propositional connectives. This is of interest because it shows, by example, that whatever advantages dynamic accounts (like Heim, 1983) or accounts that treat presuppositions as implicatures (like transparency theory) may have, neither is needed to provide a constraining account or meet Schlenker's explanatory challenge. This means, in particular, that the question of whether presupposition should be treated as semantic or pragmatic must be decided on grounds other than Schlenker's explanatory challenge. We have also seen that the components used to build the Heim-Beaver connectives all correspond to trivalent truth functions, with implications for the kinds of operations on updates that can be built with this toolbox (such as the fact that they allow only extensional projection patterns, at least with regard to trivalentisable updates). Turning to the case of quantifiers, we have seen that, as a formal matter, Peters-Kleene deployment is readily generalised to this case, but that its predictions diverge from those of established accounts. I have also given some general results regarding which quantifiers generate universal inferences under Peters-Kleene deployment, in the hope that these observations may inform future work on assessing or improving the Peters-Kleene approach.

A few other issues in the study of presupposition projection deserve a brief mention in connection with these themes. A common limitation of many approaches built to address

Schlenker's explanatory challenge, such as Peters-Kleene deployment and transparency theory, is that they treat presupposition as a single, more-or-less homogeneous phenomenon. There are many arguments that this is not the case, and that, for example, we may need to distinguish between semantic and pragmatic presuppositions. This would suggest that the kind of unified account described here will not serve as an all-purpose theory of projection. One reply to this is to abandon any hope of addressing this explanatory challenge. Another is to conclude that current theories of this kind are grossly inadequate, but to seek some general projection mechanism that is compatible with the heterogeneity of presupposition. Instead of either of these options, we might conclude that, if different kinds of presupposition are in fact different phenomena, they should be handled separately, with the goal of matching each genuinely distinct type of presupposition with a suitable projection account (where each such account tries to meet Schlenker's explanatory challenge for its kind of presupposition). For this approach, we might seek to handle semantic presuppositions (or at least one class of more or less propositional semantic presuppositions) with something like Peters-Kleene deployment, to handle presuppositional implicatures with something like transparency theory, and so on. I have presented the present approach as an account of a single phenomenon, which I have called 'presupposition', but by doing this I do not mean to deny the possibility of presupposition being heterogeneous, or to suggest that a recognition of such heterogeneity is incompatible with the kind of project that Peters-Kleene deployment and transparency theory represent.

Another issue is the empirical investigation of variation in presupposition projection with different quantifiers, and the search for empirical generalisations about which properties of quantifiers associate with which kinds of projected presuppositions, with the goal of giving an empirically robust account of the full range of quantification data. There is a great deal of descriptive and theoretical work still to be done here.

The formal points presented above have not in any sense provided a complete account of presupposition projection, but it is my hope that they have helped to elucidate some aspects of the theoretical landscape, and in particular to give a better sense of the formal properties of systems based on Peters-Kleene deployment and related ideas.

Acknowledgements

The ideas in this paper, along with a number of aspects of their presentation, owe a great deal to conversations with a number of people, including Philippe Schlenker, Ed Keenan, Daniel Büring, Ed Stabler, Sam Cumming, Emmanuel Chemla, Jessica Rett, Benjamin Spector, Danny Fox, Paul Égré, and Dave Ripley. I am also indebted to Robert van Rooij and two anonymous reviewers for many valuable criticisms and suggestions. Some of the research appearing in this paper was supported in part by the UCLA Graduate Division through a Graduate Research Mentorship and Graduate Summer Research Mentorship, by a Partner University Fund Grant from the FACE foundation, and by NSF grant BCS-0617316. I am indebted to Dominique Sportiche and Philippe Schlenker in connection with the role the last two grants played in supporting my research. The errors in this paper are, of course, my sole responsibility.

Notes

1. These including a property which I call 'extensionality' which guarantees that, under certain assumptions, the projection behaviour of any definable connective can be analysed on a world-by-world basis.
2. Such complications are certainly possible. For reasons of space, I will be unable to discuss them in detail, but, for example, George (2008b,c) offers some possible adjustments. It should be noted,

of course, that the more specialised adjustments to the general framework we include, the more dubious the explanatory value of such a framework becomes.

3. These examples are provided for purposes of illustrating the standard picture. The full empirical details of the projection facts are complex and often controversial, but, for the most part, I will not be able to explore those controversies here.
4. My primary concern is not to *motivate* these various explanatory criteria, but only to identify and differentiate them. For motivating arguments, readers are referred to the cited articles by the main advocates of these explanatory criteria.
5. Another major advantage of the Heim (1983) dynamic account over many of its contemporaries is that it can handle quantification and the presuppositions of sub-clausal constituents, but this achievement does not really differentiate dynamic and static accounts, as Heim's projection claims for the quantifiers are easily re-derived by building suitable definitions of the quantifiers into a static framework where predicates and quantifiers are given their usual functional types, but with respect to three truth values instead of two. Dynamic accounts have many other uses in, for example, the treatment of unbound anaphoric dependencies, but this does not tell us much about how much of the dynamics of meaning should be written into the meanings of individual connectives or other expressions, and does not tell us whether there is any special dynamic machinery associated with presupposition projection.
6. If the reader doubts the value of these properties, they will, I hope, at least appreciate the value of showing that these properties are not as decisive as they have been said (or implied) to be between static and dynamic accounts, or between semantic and pragmatic accounts.
7. For example, if we assume $f(\#, \#) = \#$ for every linguistically plausible binary truth function f, then we have only 3^4 possible extensions of each classical binary truth function. We might also propose, for example, that it must be that for all $x, y \in \{0, 1\}$, if $f(x, \#) = y$ then $f(x, 0) = y$ (this partly enforces a kind of requirement that presupposition failure is more akin to falsehood than to truth) – different constraints of this kind would reduce the cardinality of the class of possible connectives in different ways.
8. To see that $\ddagger$ is non-extensional, let the set of worlds be $\{w_1, w_2\}$, and let $p_1(w_1) = p_1(w_2) = 1$, and let $p_2(w_1) = 1$ and $p_2(w_2) = \#$. $\ddagger(p_1)(w_1) = p_1(w_1) = 1$, but, because $p_2(w_2) = \#$, $\ddagger(p_2)(w_1) = \#$. That is, even though $p_1(w_1) = p_2(w_1)$, $\ddagger(p_1)(w_1) \neq \ddagger(p_2)(w_1)$, so $\ddagger$ is non-extensional in my sense.
9. The relevant definition states that $C[\ddagger(u)] = \emptyset$ if there is any world w such that $\{w\}[u] = \#$, and otherwise $C[\ddagger(u)] = C[u]$.
10. As we will see below, every update generated by the Heim-Beaver system acts on contexts in a particular way: it checks to see if context satisfies its presuppositions, and, if the answer is 'no', it maps the context to #, but if the context does satisfy the presuppositions of the update, the update updates the context by removing all worlds that do not satisfy its assertions. That is, every update in the Heim-Beaver system is determined by the choice of a combination of a presupposed proposition and an asserted proposition (at least if we assume that atomic sentences are assigned updates of this sort). Further, if we require that the asserted proposition be entailed by the presupposed one, it will be possible to recover the presuppositions and assertions from the update (this is a consequence of the result that the 'update of' function is one-to-one, as noted in Remark 1). That is, a function over the class of updates will be able to map arbitrary pairs of presupposition and assertion to other arbitrary pairs of the same sort, subject (at best) to the restriction that the presupposition must always be strictly weaker than the assertion. From a perspective of constrainingness, this is almost as bad as the Karttunen and Peters (1979) system.
11. To see that u_2 is trivalentisable, suppose that it is. Now, note that $W[u_2] = W$ by the second clause, and hence, by Remark 4, for all $C \neq \#$, $C[u_2] \neq \#$, so in particular $(W \setminus V)[u_2] \neq \#$, but, by the first clause of the definition of u_2, we find that $(W \setminus V)[u_2] = \#$, so we have a contradiction, so u_2 cannot be trivalentisable.
12. The template-based dynamic semantics of LaCasse (2008) has some important similarities, but does not employ the type of isomorphism result seen here.
13. For trivalent propositions p and q and a binary trivalent truth function f, I understand $f(p, q)$ to be the unique trivalent proposition r such that $r(w) = f(p(w), q(w))$ for all $w \in W$, and I understand the unary and 0-ary cases analogously. So, for example, I will treat $\top$ as the function with the property that, for all $w \in W$, $\top(w) = 1$.

14. That is, for every context C, $C[u] = \#$ iff $C[u'] = \#$.

15. That is, we begin by recalling the definitions of the Heim-Beaver connectives: **not** $u = \varepsilon \setminus u$, u **and** $u' = u; u'$, u **or** $u' =$ **not** ((**not** u) **and** (**not** u')), and **if** $u, u' = \varepsilon \setminus (u \setminus (u; u'))$. Then we observe, by inspection of the truth tables, that the corresponding trivalent truth functions derive the Peters connectives. That is, for all $x, y \in \{0, 1\#\}$, $x \tilde{\wedge} y = x \tilde{\wedge} y$, $\top \mathring{-} x = \tilde{\neg} x$, $\tilde{\neg}(\tilde{\neg} x \tilde{\wedge} \tilde{\neg} y) = x \tilde{\vee} y$, and $\top \mathring{-} (x \mathring{-} (x \tilde{\wedge} y)) = x \tilde{\rightarrow} y$. Then we are able to confirm that for all $p, q \in P_{\{0,1,\#\}}$, and all contexts C, the following equalities hold:

   - $C[$**not** $\mathtt{u}_p] = C[\mathtt{u}_{\tilde{\neg} p}]$.
   - $C[\mathtt{u}_p$ **and** $\mathtt{u}_q] = C[\mathtt{u}_{p \tilde{\wedge} q}]$.
   - $C[\mathtt{u}_p$ **or** $\mathtt{u}_q] = C[\mathtt{u}_{p \tilde{\vee} q}]$.
   - $C[$**if** $\mathtt{u}_p, \mathtt{u}_q] = C[\mathtt{u}_{p \tilde{\rightarrow} q}]$.

16. This is not to say that no hints of an answer can be found in Peters (1977) – the left-to-right intuition underlying the Peters (1977) proposal, among other natural patterns, already suggests a way of narrowing things down, as we shall see later in this section.

17. For truth-functional connectives, linear surface order would be adequate, but for the discussion of quantifiers later on, it appears that the restrictor of the quantifier is always evaluated before its nuclear scope, which will (e.g., for quantificational phrases in object positions) require a deviation from surface order, but is consistent with linear order at LF in a theory that has obligatory leftward QR for all complement-position quantificational phrases. Other approaches to reading linear order of arguments from the syntax deserve attention as well, but I will not resolve the matter here. What is important for us is that each distinct syntactic analysis of a sentence determines a single ordering of the arguments of each function, which corresponds at least roughly to linear order.

18. That is, $0 \wedge 1 = 0 \wedge 0 = 0$; some readers may note that the intuition here is similar to the idea of lazy evaluation in some programming languages.

19. That is, if there are $x, y \in X$ such that $f/x \neq f/y$.

20. That is, $f[\vec{x}]$ is defined as follows:

   - Where $\vec{x}$ is of length 1, $\vec{x}$ is a single value x, so $f[\vec{x}] = f[x] = f /\!/\!/ \mathcal{REP}(x)$.
   - Where $\vec{x}$ is of length greater than 1, there are x and $\vec{y}$ such that $\vec{x} = x\vec{y}$, in which case $f[\vec{x}] = (f[x])[\vec{y}]$.

21. After all, an utterance of known truth value is uninformative, and therefore of dubious pragmatic felicity or communicative value.

22. Throughout this paper, I adopt what Peters and Westerståhl (2006) call the *local* perspective, rather than their *global* perspective, which is to say I am concerned with quantifiers primarily as functions from (characteristic functions of) sets to truth values.

23. The system resulting from any such choice of $\mathcal{REP}()$ will meet Schlenker's explanatory challenge, although the degree to which this is satisfying will, as always, depend on how complex and how ad hoc the definition of $\mathcal{REP}()$ is.

24. This definition can readily be made recursive, allowing us to generalize it to arbitrary conjoinable types, but what we have here will suffice for present purposes.

25. I am, for reasons of space, only able to offer this drastically simplified description of a small part of Chemla's rich and fascinating data – the reader is, again, referred to Chemla (2009) for a more complete picture.

26. It should be noted that there is at least one place where the predictions of the generalised Peters-Kleene approach diverge rather drastically from Chemla's data. Chemla finds roughly the same frequency of universal inferences for *exactly three* as for existential quantification, but it turns out that generalised Peters-Kleene deployment derives entailments like (15) in this case.

27. In classifying generalised quantifiers, I have generally attempted to follow the terminology of Peters and Westerståhl (2006).

28. If U is infinite, we also need to require that $|U - S_A| = |U - S_{A'}|$.

29. Whenever I refer to monotonicity properties of determiners, I mean monotonicity with respect to second argument position.

References

Beaver, D. (2001). *Presupposition and Assertion in Dynamic Semantics*. Stanford, CA: CSLI.

Beaver, D., & Krahmer, E. (2001). A partial account of presupposition projection. *Journal of Logic, Language and Information, 10*, 147–182.

Chemla, E. (2009). Presuppositions of quantified sentences: Experimental data. *Natural Language Semantics, 17*, 299–340.

Davis, S. (1991). *Pragmatics: A Reader*. Oxford: Oxford University Press.

Fox, D. (2008). Two short notes on Schlenker's theory of presupposition projection. *Theoretical Linguistics, 34*, 237–252.

Gazdar, G. (1979). *Pragmatics: Implicature, Presupposition and Logical Form*. New York, NY: Academic Press.

George, B. R. (2008a). A new predictive theory of presupposition projection. In T. Friedman, & S. Ito (Eds.), *Proceedings of the 18th Semantics and Linguistic Theory Conference, held March 21–23, 2008 at The University of Massachusetts, Amherst* (pp. 358–375). Ithaca, NY: Cornell University Press.

George, B. R. (2008b). *Predicting presupposition projection: Some alternatives in the strong Kleene tradition* (Unpublished paper). University of California, Los Angeles, USA. Retrieved from http://semarch.linguistics.fas.nyu.edu/Archive/DY0YTgxN/

George, B. R. (2008c). *Presupposition repairs: A static, trivalent approach to predicting projection* (Unpublished masters thesis). Los Angeles, USA: University of California.

Heim, I. (1983). On the projection problem for presuppositions. In M. Barlow, D. Flickinger, & M. Westcoat (Eds.), *WCCFL 2: Second Annual West Coast Conference on Formal Linguistics* (pp. 114–126). Stanford, CA: Stanford University Press. (As reprinted on pp. 397–405 of Davis (1991)).

Karttunen, L. (1973). Presuppositions of compound sentences. *Linguistic Inquiry, 4*, 167–193.

Karttunen, L. (1974). Presuppositions and linguistic context. Theoretical. *Linguistics, 1*, 181–194.

Karttunen, L., & Peters, S. (1979). Conventional implicatures in Montague grammar. In C.-K. Oh & D. Dineen (Eds.), *Syntax and Semantics 11: Presupposition* (pp. 1–56). New York, NY: Academic Press.

Keenan, E. L., & Stavi, J. (1986). A semantic characterization of natural language determiners. *Linguistics and Philosophy, 9*, 253–326.

Kleene, S. (1952). *Introduction to Metamathematics*. Amsterdam: North Holland.

Krahmer, E. (1998). *Presuppositions and Anaphora*. Stanford, CA: CSLI.

LaCasse, N. R. (2008). *Constraints on Connectives and Quantifiers: Solving the over-generation problem of dynamic semantics (Unpublished paper)*. Los Angeles, USA: University of California.

Peters, S. (1977). A truth-conditional formulation of Karttunen's account of presupposition. In Texas Linguistic (Ed.), *Forum 6* (pp. 137–149). Austin, TX: University of Texas.

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

Schlenker, P. (2006). Anti-dynamics: Presupposition projection without dynamic semantics. *Journal of Logic, Language and Information, 16*, 325–356.

Schlenker, P. (2008a). Be articulate: A pragmatic theory of presupposition projection. *Theoretical Linguistics, 34*, 157–212.

Schlenker, P. (2008b). Presupposition projection: Explanatory strategies. Theoretical. *Linguistics, 34*, 287–316.

Schlenker, P. (2008c). Presupposition projection: The new debate. In T. Friedman, & S. Ito (Eds.), *Proceedings of the 18th Semantics and Linguistic Theory Conference, held March 21–23, 2008 at The University of Massachusetts, Amherst* (pp. 655–693). Ithaca, NY: Cornell University Press.

Soames, S. (1989). Presupposition. In D. Gabbay, & F. Guenther (Eds.), *Handbook of Philosophical Logic*, Vol. IV (pp. 553–616). Dordrecht: D. Reidel.

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

van Eijck, J. (1996). Quantifiers and partiality. In J. van der Does & J. van Eijck (Eds.), *Quantifiers, Logic, and Language* (pp. 105–144). Chicago, IL: University of Chicago Press.

Appendix A. Homomorphism results

Recall that the key isomorphism result in Section 2.2 depends on the following equivalences, for all trivalent propositions p and q:

(16)
a. $\mathtt{u}_\top = \varepsilon$.
b. $\mathtt{u}_{p \tilde{\wedge} q} = \mathtt{u}_p; \mathtt{u}_q$.
c. $\mathtt{u}_{p \mathring{-} q} = \mathtt{u}_p \setminus \mathtt{u}_q$.
d. $\mathtt{u}_{p \mathring{\vee} q} = \mathtt{u}_p \cup \mathtt{u}_q$.
e. $\mathtt{u}_{p \mathring{\wedge} q} = \mathtt{u}_p \cap \mathtt{u}_q$.

Below, I sketch the proofs of (16-b) and (16-c) for illustration. The proof of (16-a) is trivial, and the proofs of (16-d) and (16-e) are analogous to the proof for (16-c).

Remark 23 For all $p, q \in P_{\{0,1,\#\}}$, $\mathtt{u}_{p \tilde{\wedge} q} = \mathtt{u}_p; \mathtt{u}_q$.

Proof First, observe that $C[\mathtt{u}_{p \tilde{\wedge} q}] = \#$ iff $C[\mathtt{u}_p; \mathtt{u}_q] = \#$.

Supposing $C[\mathtt{u}_{p \tilde{\wedge} q}] = \#$, either $C = \#$ (in which case $C[\mathtt{u}_p; \mathtt{u}_q] = \#[\mathtt{u}_p][\mathtt{u}_q] = \#$) or there is $w \in C$ s.t. $(p \tilde{\wedge} q)(w) = \#$, which (by inspection of the truth table for $\tilde{\wedge}$) is the case iff either $p(w) = \#$ or $p(w) = 1$ and $q(w) = \#$. If $p(w) = \#$ then $C[\mathtt{u}_p] = \#$ because $w \in C$, so $C[\mathtt{u}_p; \mathtt{u}_q] = C[\mathtt{u}_p][\mathtt{u}_q] = \#[\mathtt{u}_q] = \#$. If $p(w) = 1$ and $q(w) = \#$, then $w \in C[\mathtt{u}_p]$, so by definition of $\mathtt{u}_q$, $C[\mathtt{u}_p; \mathtt{u}_q] = C[\mathtt{u}_p][\mathtt{u}_q] = \#$. Thus, whenever $C[\mathtt{u}_{p \tilde{\wedge} q}] = \#$, $C[\mathtt{u}_p; \mathtt{u}_q] = \#$.

On the other hand, if $C[\mathtt{u}_p; \mathtt{u}_q] = \#$, then either $C = \#$ (in which case $C[\mathtt{u}_{p \tilde{\wedge} q}] = \#$), or $C[\mathtt{u}_p] = \#$ (in which case there is $w \in C$ such that $p(w) = \#$, so $(p \tilde{\wedge} q)(w) = \#$, so $C[\mathtt{u}_{p \tilde{\wedge} q}] = \#$), or $C[\mathtt{u}_p] \neq \#$ and $C[\mathtt{u}_p][\mathtt{u}_q] = \#$ (in which case there is $w \in C[\mathtt{u}_p]$ s.t. $q(w) = \#$, which is to say there is $w \in C$ such that $p(w) = 1$ and $q(w) = \#$, which means that $(p \tilde{\wedge} q)(w) = \#$, so $C[\mathtt{u}_{p \tilde{\wedge} q}] = \#$). Thus, if $C[\mathtt{u}_p; \mathtt{u}_q] = \#$ then $C[\mathtt{u}_{p \tilde{\wedge} q}] = \#$, giving us the needed biconditional.

Now, suppose $C[\mathtt{u}_{p \tilde{\wedge} q}] \neq \#$ (and so, as we just saw, $C[\mathtt{u}_p; \mathtt{u}_q] \neq \#$). In this case, $C[\mathtt{u}_{p \tilde{\wedge} q}] = C \cap \{w \mid p(w) = 1 \text{ and } q(w) = 1\} = C \cap \{w \mid p(w) = 1\} \cap \{w \mid q(w) = 1\}$, and $C[\mathtt{u}_p; \mathtt{u}_q] = C[\mathtt{u}_p][\mathtt{u}_q] = C[\mathtt{u}_p] \cap \{w \mid q(w) = 1\} = C \cap \{w \mid p(w) = 1\} \cap \{w \mid q(w) = 1\}$, so $C[\mathtt{u}_{p \tilde{\wedge} q}] = C[\mathtt{u}_p; \mathtt{u}_q]$.

Together, the above observations show that for all p and q, $C[\mathtt{u}_{p \tilde{\wedge} q}] = C[\mathtt{u}_p; \mathtt{u}_q]$, so $\mathtt{u}_{p \tilde{\wedge} q} = \mathtt{u}_p; \mathtt{u}_q$. ■

Remark 24 For all $p, q \in P_{\{0,1,\#\}}$, $\mathtt{u}_{p \mathring{-} q} = \mathtt{u}_p \setminus \mathtt{u}_q$.

Proof Again, we begin by showing that $C[\mathtt{u}_{p \mathring{-} q}] = \#$ iff $C[\mathtt{u}_p \setminus \mathtt{u}_q] = \#$. From the truth table for $\mathring{-}$, note that $(p \mathring{-} q)(w) = \#$ iff $p(w) = \#$ or $q(w) = \#$. Thus, $C[\mathtt{u}_{p \mathring{-} q}] = \#$ iff there is $w \in C$ s.t. $p(w) = \#$ or $q(w) = \#$. This is in turn the case iff $C[\mathtt{u}_p] = \#$ or $C[\mathtt{u}_q] = \#$, which is the condition under which $C[\mathtt{u}_p \setminus \mathtt{u}_q] = \#$.

Now, suppose $C[\mathtt{u}_{p \mathring{-} q}] \neq \#$ (and so $C[\mathtt{u}_p \setminus \mathtt{u}_q] \neq \#$). In this case, $C[\mathtt{u}_{p \mathring{-} q}] = C \cap \{w \mid (p \mathring{-} q)(w) = 1\} = C \cap \{w \mid p(w) = 1 \text{ and } q(w) = 0\} = C \cap \{w \mid p(w) = 1\} \cap \{w \mid q(w) = 0\} = (C \cap \{w \mid p(w) = 1\}) \cap (C \cap \{w \mid q(w) = 0\})$. From $C[\mathtt{u}_{p \mathring{-} q}] \neq \#$, it follows that $q(w) \neq \#$ for all $w \in C$, so we derive $C[\mathtt{u}_{p \mathring{-} q}] = (C \cap \{w \mid p(w) = 1\}) \cap (\{w \in C \mid q(w) \neq 1\}) = (C \cap \{w \mid p(w) = 1\}) \setminus (C \cap \{w \mid q(w) = 1\})$. Noting that $C[\mathtt{u}_p] \neq \#$ and $C[\mathtt{u}_q] \neq \#$, we find that $C[\mathtt{u}_p] = C \cap \{w \mid p(w) = 1\}$ and likewise $C[\mathtt{u}_q] = C \cap \{w \mid q(w) = 1\}$. Thus $C[\mathtt{u}_{p \mathring{-} q}] = C[\mathtt{u}_p] \setminus C[\mathtt{u}_q] = C[\mathtt{u}_p \setminus \mathtt{u}_q]$. ■

Appendix B. Peters-Kleene deployment of classical truth functions

Below I sketch some proofs of the key claims made in Section 3.2 about the Peters-Kleene deployment of the classical truth functions.

Remark 9 $f_\neg[\#] = \#$.

Proof By the definitions of Peters-Kleene deployment and repair sets, $f_\neg[\#] = f_\neg /// \mathcal{REP}(\#) = f_\neg ///\{0, 1\}$. Now, since $f_\neg/0 = 1 \neq 0 = f_\neg/1$, we find $f_\neg ///\{0, 1\}$ is the unique 0-ary function that maps all sequences of length 0 to #, which (by our convention of identifying singletons with 0-ary functions), is just # itself. ■

Remark 10 For any truth value x, $f_\wedge[\#, x] = f_\vee[\#, x] = f_\rightarrow[\#, x] = \#$.

Proof For $f_\wedge$, note that $f_\wedge(0, 1) = 0 \neq 1 = f_\wedge(1, 1)$. This in turn means that $f_\wedge/0 \neq f_\wedge/1$, since if these two were the same function, they would behave the same everywhere, but in fact they treat the value 1 differently. By the above definitions, this means that $f_\wedge /\!/\!/ \mathcal{REP}(\#) = f_\wedge /\!/\!/ \{0, 1\}$ is the unary function g that maps all truth values to #, so, since $g(z) = \#$ for all z, $g /\!/\!/ \mathcal{REP}(x) = \#$ for all x, hence $(f_\wedge /\!/\!/ \mathcal{REP}(\#)) /\!/\!/ \mathcal{REP}(x) = \#$ for all x, which is to say $f_\wedge[\#, x] = \#$ for all x. The arguments for $f_\vee$ and $f_\to$ are analogous, resting on the observations that $f_\vee(0, 0) = 0 \neq 1 = f_\vee(1, 0)$, and that $f_\to(0, 0) = 1 \neq 0 = f_\to(1, 0)$. ■

Remark 11 $f_\wedge[0, \#] = 0$, $f_\to[0, \#] = 1$, and $f_\vee[1, \#] = 1$.

Proof $f_\wedge(0, 0) = 0 = f_\wedge(0, 1)$, so $f_\wedge/0$ is the constant function that maps all truth values to 0. Further, since $\mathcal{REP}(0) = \{0\}$, $f_\wedge /\!/\!/ \mathcal{REP}(0) = f_\wedge/0$. Now, since we have a constant function, $(f_\wedge/0)(0) = 0 = (f_\wedge/0)(1)$, which means $(f_\wedge/0) /\!/\!/ \{0, 1\} = 0$, so $f_\wedge[0, \#] = (f_\wedge /\!/\!/ \mathcal{REP}(0)) /\!/\!/ \mathcal{REP}(\#) = (f_\wedge /\!/\!/ \{0\}) /\!/\!/ \{0, 1\} = 0$. Analogously, $f_\vee(1, 0) = 1 = f_\vee(1, 1)$, so $f_\vee /\!/\!/ \mathcal{REP}(1)$ is the constant function that maps all truth values to 1, so $f_\vee[1, \#] = 1$, and $f_\to(0, 0) = 1 = f_\to(0, 1)$, so $f_\to /\!/\!/ \mathcal{REP}(0)$ is the constant function that maps all truth values to 1, so $f_\to[0, \#] = 1$. ■

Remark 12 $f_\wedge[1, \#] = \#$, $f_\to[1, \#] = \#$, and $f_\vee[0, \#] = \#$.

Proof Since $f_\wedge(1, 1) = 1$ and $f_\wedge(1, 0) = 0$, $f_\wedge/1$ is the identity function on truth values (that is, it maps 1 to 1 and 0 to 0). Call this function I. Since $\mathcal{REP}(1) = \{1\}$, $f_\wedge /\!/\!/ \mathcal{REP}(1) = f_\wedge/1 = I$. Observe further that $f_\to /\!/\!/ \mathcal{REP}(1) = f_\vee /\!/\!/ \mathcal{REP}(0) = I$. Now $I /\!/\!/ \mathcal{REP}(\#) = I /\!/\!/ \{0, 1\}$, and $I(0) = 0 \neq 1 = I(1)$, so $I /\!/\!/ \{0, 1\} = \#$. This means that $f_\wedge[1, \#] = (f_\wedge /\!/\!/ \mathcal{REP}(1)) /\!/\!/ \mathcal{REP}(\#) = I /\!/\!/ \{0, 1\} = \#$, and likewise, by our above observation for I, $f_\to[1, \#] = \#$ and $f_\vee[0, \#] = \#$. ■