At least some distributive items aren't distributive items[*]

Haoze Li & Jess H.-K. Law

New York University, Rutgers University

1. Introduction

Since Link (1987), the distributivity operator (D-Op), as in (1) (based on Link (1987), represented in Link (1998:109-110)), has become a key to understanding distributivity.

(1) $[\![D]\!] = \lambda P \lambda y \forall x[x \leq y \rightarrow P(x)]$

The D-Op takes a predicate and a distributive key and distributes the predicate over every subpart of the distributive key. In natural languages, some distributive items have been argued to lexically encode the D-op, like adverbial *each* in English (e.g., Link 1987) and *dou* in Mandarin (Lin 1998), as exemplified in (2) and (3).

(2) [Everyone/They]$_{Key}$ **each** [built boats]$_{Share}$. (English)

(3) [Meigeren/Tamen]$_{Key}$ **dou** [kan-le shu]$_{Share}$
everyone/they DOU read-ASP book
'Everyone/They each read books.' (Mandarin)

Many more lexical items in natural language have been said to contribute distributivity, like binominal *each* in English (Choe 1987, Safir & Stowell 1988, Zimmermann 2002), and *ge* in Mandarin (Lin 2004, Lee et al. 2009), although these lexical items have peculiar properties not predicted by the definition of the D-Op.

A more recent trend on distributivity, however, is to rethink the relation between the D-op and overt distributivity along the following lines: perhaps a lot of the distributive items just work in tandem with the real D-Op, but do not serve as D-Ops themselves. A recent study with this vision is Champollion (2015a), who, drawing on the distinct distribution of

[*]We are grateful to Lucas Champollion for his inspiring seminar on mereology in Fall 2014. We would also like to thank the audience of Rutgers Semantics Group for their comments on a previous version of this work.

bi-nominal *each*, argues that this lexical item introduces dependent numerals parasitic on a covert D-Op.

In this paper, we defend the same general view by placing under scrutiny *ge*, an adverb in Mandarin considered as the distributive item by many scholars (e.g., Lin 2004; Lee et al 2009). In Section 2, we discuss a few peculiar properties of *ge* that are unexpected from a D-Op analysis, which serve as the basis for not assimilating *ge* to a D-Op. In Section 3, we propose that *ge* is parasitic on a D-Op, as it contributes a test, in the sense of dynamic semantics, on the output of D-Op. Section 4 lays out the formal details of the framework and how it is implemented and Section 5 concludes.

2. Selectional restrictions of *ge*

The following example shows two properties of *ge*, namely, it correlates with a distributive interpretation.

(4) [Meigeren/Tamen]$_{Key}$ **ge** [fu-le yi-zhang zhangdan]$_{Share}$
everyone/they GE pay-Asp one-CL bill
'Everyone/They each paid a bill.'

However, in addition to requiring a plural distributive key, *ge* imposes a selectional restriction on the distributive share, i.e., the VP following *ge*. In particular, it requires that the VP do not contain a bare noun phrase. So, while *yi-zhang zhangdan* 'one bill' is fine in (4), the bare noun version *zhangdan* 'bills' is not, as shown below:

(5) *Meigeren **ge** fu-le zhangdan.
everyone GE pay-ASP bill
'Everyone each paid bills.'

The unacceptability of (5) can also be rescued by introducing an adverbial indefinite *time*-phrase, which is used to count events (Landman 2004), as shown in (6). In Mandarin, the indefinite *time*-phrase can occur between a verb and an object. For a concrete analysis of the syntax of *time*-phrases in Mandarin, we refer the reader to Huang et al. (2009).

(6) Meigeren **ge** fu-le yi-ci zhangdan.
everyone GE pay-Asp one-CL bill
'Everyone each paid bills once.'

Another way to save the bare noun is to introduce a possessive reflexive like *ziji de* 'self's':

(7) Meigeren ge fu-le ziji-de zhangdan.
everyone GE pay-Asp self-DE bill
'Everyone paid their own bill.'

These selectional restrictions on the distributive share are not shared by other distributive items, like adverbial *each*, as in (2), and *dou*, as in (3). Nor are they enforced by the definition of the D-Op in (1).

3. Proposal

We propose that *ge* is a VP modifier parasitic on a D-Op, rather than the D-Op itself. This explains why the presence of *ge* entails a distributive interpretation. Consider the example (4), which is taken to have a silent D-Op, as illustrated in (8). We spell out in more detail how the D-Op is introduced and interpreted in Section 4.2.

(8) Everyone D-Op [GE [$_{\text{VP}}$ pay one-CL bill]]

In our proposal, *ge* takes VP as argument and contributes an 'anti-cumulativity' condition on the output of D-Op to the effect that **no sum of non-singleton subsets of the events in the output of D-Op is an event predicate denoted by VP**. We provide an informal definition of *ge* in (9), which will be modified in Section 4.2.

(9) $ge \rightsquigarrow \lambda P \lambda e[P(e) \wedge^{\neg\exists e,e'[e \in E \,\wedge\, e' \in E \,\wedge\, e \neq e' \,\wedge\, P(e \oplus e')]}]$, in which E is an anaphoric element referring to a set of events in the output of the D-Op.

According to this definition, *ge* takes an event predicate as an argument, i.e., the VP in (8). While it does not have any interpretive effects in the at-issue component, it checks whether the meaning of the whole sentence fulfills the condition indicated by the superscript. This condition is formalized as a post-supposition in the sense of Brasoveanu (2013).

In (8), the presence of the D-Op gives rise to a distributive reading. Suppose that the witness set of *everyone* includes three individuals *Libai, Dufu* and *Zilu*. The distributive reading can informally be represented as: for each of these three people, there is a paying event whose agent is him and whose theme is one bill. As a result, we get three such events, as in (10).

(10)

x	e
Libai	$\mathbf{pay}(e_1) \wedge \mathbf{bill}(\mathbf{th}(e_1)) \wedge \lvert\mathbf{th}(e_1)\rvert = 1$
Dufu	$\mathbf{pay}(e_2) \wedge \mathbf{bill}(\mathbf{th}(e_2)) \wedge \lvert\mathbf{th}(e_2)\rvert = 1$
Zilu	$\mathbf{pay}(e_3) \wedge \mathbf{bill}(\mathbf{th}(e_3)) \wedge \lvert\mathbf{th}(e_3)\rvert = 1$

Following the cumulativity assumption for thematic roles (Krifka 1998; Landman 2000), as stated in (11), the theme role of the sum of any two or more events in (10) cannot be one bill, but two or more bills.

(11) **Cumulativity assumption for thematic roles**
For any thematic role θ, it holds that $\theta(e \oplus e') = \theta(e) \oplus \theta(e')$
(The θ of the sum of two events is the sum of their θs.)

The VP in (8) is an event predicate that can be characterized as a set of paying-of-one-bill events. The sum of any two or more events in (10) is not a paying-of-one-bill event. Therefore, the output of the D-Op fulfills the requirement of *ge*. This is why having a numeral phrase inside a VP helps to satisfy the anti-cumulativity requirement of *ge*. The same reasoning can be extended to the example with an indefinite *time*-phrase, i.e., (6). However, for reasons of space, we have to leave the details to the reader.

In contrast, if the numeral phrase is replaced with a bare noun, as in (5), the requirement of *ge* cannot be satisfied. In the same context described above, the D-Op results in three paying-of-bills events, as in (12).

(12)

x	e
Libai	$\textbf{pay}(e_1) \wedge \textbf{bills}(\textbf{th}(e_1))$
Dufu	$\textbf{pay}(e_2) \wedge \textbf{bills}(\textbf{th}(e_2))$
Zilu	$\textbf{pay}(e_3) \wedge \textbf{bills}(\textbf{th}(e_3))$

In this example, the sum of any two or more paying-of-bills events is another paying-of-bills event. The requirement of *ge* is not fulfilled. Hence, (5) is not acceptable.

The fact that possessive reflexives can help satisfy the anti-cumulativity requirement of *ge*, as shown in (7), is due to the dependency between distributed subjects and reflexives. Reflexives are bound variable anaphora. In (7), the possessive reflexive is bound by the subject, which is taken as the distributive key by the silent D-Op, as in (13a). As a result, the value of the reflexive varies with the distributive key. Simply speaking, the sentence is understood as 'Libai paid his bills, Dufu paid his bills and Zilu paid his bills' in the context mentioned above. Therefore, the set of events output by the D-Op looks like (13b).

(13) a. Everyone$_1$ D-op GE [$_{\text{VP}}$ paid *ziji*$_1$'s bills]

b.

x	e
Libai	$\textbf{pay}(e_1) \wedge \textbf{th}(e_1) = \textbf{Libai's bill}$
Dufu	$\textbf{pay}(e_2) \wedge \textbf{th}(e_2) = \textbf{Dufu's bill}$
Zilu	$\textbf{pay}(e_3) \wedge \textbf{th}(e_3) = \textbf{Zilu's bill}$

The events in the set characterized by the denotation of VP are paying events and take atomic individuals' bills as theme arguments. However, the sum of any two or more events in (13b) cannot take an atomic individual's bills as its theme argument. For example, the theme of $e_1 \oplus e_2$ is **Libai's bill** $\oplus$ **Dufu's bill**, instead of **Libai's bill** or **Dufu's bill**.

4. Formalization

To formalize the current proposal, we will use dynamic plural predicate logic (DPlL) (van den Berg 1996; Nouwen 2003) enriched with post-suppositions (Brasoveanu 2008, 2013; Henderson 2014). This framework is well-suited to the modeling of dependency between the D-Op and *ge*. Under DPlL, the D-Op can generate a set of events, and the dynamic mechanism allows the requirement of *ge* to be anaphoric to the output of the D-Op.

4.1 Dynamic plural predicate logic with post-suppositions

The basic innovation of DPlL as compared with dynamic predicate logic is that instead of evaluating formulas with respect to pairs of assignments $\langle i, j\rangle$, they are evaluated wth respect to pairs of sets of assignments $\langle I, J\rangle$. Specifically, the values of a variable under a set of assignments can be collected in the following way:

(14) $I(x) := \{i(x) : i \in I\}$

Moreover, the variable assignment $I[x]J$ in DPlL can be defined based on $i[x]j$ in dynamic predicate logic, as illustrated in (15).

(15) $I[x]J := \forall i \in I \exists j \in J[i[x]j] \wedge \forall j \in J \exists i \in I[i[x]j]$

Adding post-suppositions enriches evaluation contexts. In particular, post-suppositions are formulas introduced at certain points in the interpretation that are passed on from local context to local context and that need to be satisfied only globally, relative to the final output context (for a useful metaphor for post-suppositions, see Champollion (2015a)). Therefore, a context is a set of assignments I indexed with a set of tests ζ, represented as $I[\zeta]$. That is, output contexts not only contain the values of the previously introduced variables but also conditions on these values. If the input context I is indexed with ζ, the latter is passed on to the output context J. In this framework, the interpretation function is not simply $[\![\cdot]\!]^{\langle I,J\rangle}$, but $[\![.]\!]^{\langle I[\zeta],J[\zeta']\rangle}$ in which ζ and ζ' are possibly empty sets of tests and $\zeta \subseteq \zeta'$. We mark a formula ϕ as a post-supposition by superscripting it, as shown in (16). It indicates that the post-supposition ϕ does not update the input assignment I in any way. It is simply added to the input set of tests.

(16) $[\![^{\phi}]\!]^{\langle I[\zeta],J[\zeta']\rangle}$ is true iff ϕ is a test, $I = J$ and $\zeta' = \zeta \cup \{\phi\}$

The definition in (16) interacts with the definition of truth in (17) to guarantee that post-suppositions are evaluated after the at-issue update. The definition of truth treats the formulas $\psi_1, ... \psi_n$ as post-suppositions, which is performed on the final output context J.

(17) Truth: ϕ is true relative to an input context $I[\emptyset]$ iff there is an output set of assignments J and a (possibly empty set) of tests $\{\psi_1, ..., \psi_n\}$ s.t. $[\![\phi]\!]^{\langle I[\emptyset],J[\{\psi_1,...,\psi_n\}]\rangle}$ is true and $[\![\psi_1 \wedge ... \wedge \psi_n]\!]^{\langle J[\emptyset],J[\emptyset]\rangle}$ is true.

We assume standard models $M = \langle D_e, D_v, D_s, F\rangle$, where D_e is the domain of individuals, D_v is the domain of events, D_s is the domain of assignments, and F is the basic interpretation function assigning each n-ary relation of type τ_1 ... τ_n a subset of $D_{\tau 1} \times ... \times D_{\tau n}$. Basic formulas are interpreted in the following way.

(18) a. $[\![P(x_1, ...x_n)]\!]^{\langle I[\zeta],J[\zeta']\rangle}$ is true iff $I = J, \zeta = \zeta'$and $\forall j_s \in J(P(j(x_1), ...j(x_n)))$

b. $[\![\phi \wedge \psi]\!]^{\langle I[\zeta], J[\zeta']\rangle}$ is true iff there is a K and ζ'' s.t. $[\![\phi]\!]^{\langle I[\zeta], K[\zeta'']\rangle}$ is true and $[\![\psi]\!]^{\langle K[\zeta''], J[\zeta']\rangle}$ is true.
c. $[\![\neg\phi]\!]^{\langle I[\zeta], J[\zeta']\rangle}$ is true iff $I = J, \zeta = \zeta'$ and for any non-empty subset H of J, there is no K and ζ'' s.t. $[\![\phi]\!]^{\langle H[\zeta], K[\zeta'']\rangle}$ is true

Within the framework, there are two kinds of cardinality predicates: one at the domain-level (DC) and one at the evaluation-level (EC). Domain-level cardinality predicates like $DC(x) = 1$ and $DC(x) = 2$ are evaluated by checking the cardinality of the set of atomic parts of an individual, as shown in (19). Evaluation-level cardinality predicates like $EC(x) = 1$ and $EC(x) = 2$ work by gathering all values of a variable under a set of assignments and checking the cardinality of the resulting set, as in (20).

(19) $[\![DC(x)=n]\!]^{\langle I[\zeta], J[\zeta']\rangle}$ is true iff $I=J, \zeta=\zeta'$ and $\forall j \in J |\{x' : x' \leq j(x) \wedge \text{atom}(x')\}| = n$

(20) $[\![EC(x) = n]\!]^{\langle I[\zeta], J[\zeta']\rangle}$ is true iff $I = J, \zeta = \zeta'$ and $|J(x)| = n$

According to Brasoveanu (2008), universal quantification is decomposed into a a maximization operation over the restrictor formula and a D-Op over the nuclear scope formula, i.e., $\forall x[\phi](\psi)$ abbreviates $\mathbf{max}^x(\phi) \wedge D(\psi)$. The **max** operator introduces a new variable x and stores in J the maximal set of individuals satisfying the formula it scopes over:

(21) $[\![\mathbf{max}^x(\phi)]\!]^{\langle I[\zeta], J[\zeta']\rangle}$ is true iff $[\![[x] \wedge \phi]\!]^{\langle I[\zeta], J[\zeta']\rangle}$ is true and $\neg\exists J'[J(x) \subset J'(x) \wedge [\![[x] \wedge \phi]\!]^{\langle I[\zeta], J'[\zeta']\rangle}]$

The D-Op is defined as in (22) (Henderson 2014). It essentially splits its input context into singleton sets of assignments, which serve as local contexts for the elements in the scope of D-Op. Then, the D-Op applies the expression in its scope to each of these singleton sets and collects the outputs back together into a global context.

(22) $[\![D(\phi)]\!]^{\langle I[\zeta], J[\zeta']\rangle}$ is true iff $\zeta = \zeta'$ and there exists a partial function F from assignments to sets of assignments such that:
a. $I = \text{Dom}(F)$ and $J = \bigcup \text{Ran}(F)$
b. there is a possibly empty set of tests ζ'' such that $\forall i \in I[[\![\phi]\!]^{\langle \{i\}[\zeta], F(i)[\zeta \cup \zeta'']\rangle} \wedge [\![\bigwedge \zeta'']\!]^{\langle J[\zeta], J[\zeta]\rangle}]$

Since $\zeta = \zeta'$, any of the post-suppositional tests must be introduced within the scope of the D-Op, must be satisfied relative to the output $J[\zeta]$ and are not passed on.

4.2 Definitions and derivations

In this subsection, we work through several derivations to show how the system works. We include two derivations, one in which *ge* scopes over a numeral phrase, and one in which *ge* scopes over a reflexive. Mandarin *ge* is taken to be an event predicate modifier that introduces a post-supposition, as defined in (23) and (24) using DPlL.

(23) $ge \rightsquigarrow \lambda V \lambda e[V(e) \wedge {}^{\neg \exists e'[e' \sqsubseteq e \wedge EC(e')>1 \wedge \exists e''[e'' = \oplus e' \wedge V(e'')]]}]$

(24) a. $[\![e' \sqsubseteq e]\!]^{\langle I[\zeta],J[\zeta']\rangle}$ is true iff $I = J, \zeta = \zeta'$ and $J(e') \subseteq J(e)$
b. $[\![e'' = \oplus e']\!]^{\langle I[\zeta],J[\zeta']\rangle}$ is true iff $I = J, \zeta = \zeta'$ and $\forall j \in J[j(e'') = \oplus J(e')]$

The post-supposition brought by *ge* states that there is no plural event variable e' such that it is a non-singleton subset of another plural event variable e, and the sum of the values assigned to e' is in V. In the following derivations, we will see that e is anaphoric to the output of the D-Op.

The fragment that we will use in the derivations is given in (25). Inspired by Champollion (2015b), we assume two functional items *th* and v_{ag}, which encode the theme role and the agent role respectively, and determine the scope of quantifiers.[1]

(25) Fragment (Types: **e** := *se*; **v** := *sv*; **t** := *st((st)t)*)

Lexical item	**Translation**	**Type**
fu 'pay'	$\lambda e[\mathbf{pay}(e)]$	**vt**
zhangdan 'bill'	$\lambda x[\mathbf{bill}(x)]$	**et**
yi-zhang 'one'	$\lambda P \lambda Q \exists x[P(x) \wedge DC(x) = 1 \wedge Q(x)]$	**et((et)t)**
meigeren 'everyone'	$\lambda Q[\mathbf{max}^x(\mathbf{person}(x)) \wedge D(Q(x))]$	**(et)t**
ziji$_x$ 'self$_x$'	$\lambda P[P(x)]$	**(et)t**
Possessive *de*	$\lambda \mathscr{Q} \lambda P \lambda Q[\mathbf{max}^y(P(y) \wedge \mathscr{Q}(\lambda x[\mathbf{of}(y,x)])) \wedge Q(y)]$	**((et)t)((et)((et)t))**
th	$\lambda \mathscr{Q} \lambda V \lambda e[\mathscr{Q}(\lambda x[V(e) \wedge \mathbf{th}(e) = x])]$	**((et)t)((vt)vt)**
v_{ag}	$\lambda V \lambda \mathscr{Q} \lambda \mathscr{V}[\mathscr{Q}(\lambda x[\mathscr{V}(\lambda e[V(e) \wedge \mathbf{ag}(e) = x])])]$	**(vt)(((et)t)(((vt)t)t))**
∃-closure	$\lambda V \exists e[V(e)]$	**(vt)t**

Now, we are in a position to derive the relevant examples. (26) provides the composition of the example (4), in which *ge* is licensed by the numeral phrase.

(26) a. [$_{\text{CP}}$ ∃-closure [$_{\text{IP}}$ Everyone [$_{v\text{P}}$ v_{ag} [$_{\text{VP1}}$ *ge* [$_{\text{VP2}}$ pay [$_{\text{TH}}$ th [one bill]]]]]]]
b. TH $\rightsquigarrow \lambda V \lambda e \exists x[\mathbf{bill}(x) \wedge V(e) \wedge \mathbf{th}(e) = x]$

[1] For simplicity, *meigeren* 'everyone' in our fragment encodes a D-Op. We are aware that this is different from the classic view that *meigeren* 'everyone' is a plural entity and the D-Op is introduced by something else in a sentence (e.g., Lin 1998). Letting the D-Op not be an inherent part of *meigeren* 'everyone' is perfectly compatible with our analysis, as long as there is a D-Op that *ge* can be parasitic to.

c. $\text{VP1} \rightsquigarrow \lambda e \exists x \begin{bmatrix} \mathbf{bill}(x) \wedge DC(x) = 1 \wedge \mathbf{pay}(e) \wedge \mathbf{th}(e) = x \wedge \\ \neg\exists e' \begin{bmatrix} e' \sqsubseteq e \wedge \ EC(e') > 1 \wedge \\ \exists e''[e'' = \oplus e' \ \wedge \ \exists x'[\mathbf{bill}(x') \ \wedge \ DC(x') = 1 \wedge \ \mathbf{pay}(e'') \ \wedge \ \mathbf{th}(e'') = x'] \end{bmatrix} \end{bmatrix}$

d. $\text{IP} \rightsquigarrow \lambda \mathscr{V}[\mathbf{max}^y(\mathbf{person}(y)) \wedge D(\mathscr{V}(\lambda e[\mathbf{VP1}(e) \wedge \mathbf{ag}(e) = y]))]$

e. $\text{CP} \rightsquigarrow \mathbf{max}^y(\mathbf{person}(y)) \wedge D(\exists e[\mathbf{VP1}(e) \wedge \mathbf{ag}(e) = y])]$

The dynamic version of (26e) can be represented as (27), in which the post-supposition is discharged in the scope of the D-Op.

(27) $$\mathbf{max}^y(\mathbf{person}(y)) \wedge \\ D\left(\begin{array}{l} [e] \wedge [x] \wedge \mathbf{bill}(x) \wedge DC(x) = 1 \wedge \mathbf{pay}(e) \wedge \mathbf{ag}(e) = y \wedge \mathbf{th}(e) = x \\ \wedge \ \neg([e'] \wedge \ e' \sqsubseteq \ e \ \wedge \ EC(e') > 1 \wedge \ [e''] \ \wedge \ e'' = \oplus e' \ \wedge \ [x] \wedge \ \mathbf{bill}(x) \ \wedge \ DC(x) = 1 \wedge \ \mathbf{pay}(e'') \ \wedge \ \mathbf{th}(e'') = x]) \end{array} \right)$$

Briefly, the at-issue component of (27) says: given an input state I, $\mathbf{max}^y(\mathbf{person}(y))$ introduces the maximum set of people across the cells of the index y; the D-Op divides the state into substates restricted by the value of y, checks that the value of y in each row is the agent of an paying-of-one-bill event, then collects the substates again. This update process can be illustrated as in (28), which is exactly what we want (see (10)).

(28) $$\begin{array}{|c|} \hline y \\ \hline \text{Libai} \\ \hline \text{Dufu} \\ \hline \text{Zilu} \\ \hline \end{array} \xrightarrow{\text{D}} \begin{array}{c} \begin{array}{|c|} \hline y \\ \hline \text{Libai} \\ \hline \end{array} \rightarrow \begin{array}{|c|c|} \hline y & e \\ \hline \text{Libai} & \mathbf{pay\text{-}one\text{-}bill}(e_1) \\ \hline \end{array} \\ \\ \begin{array}{|c|} \hline y \\ \hline \text{Dufu} \\ \hline \end{array} \rightarrow \begin{array}{|c|c|} \hline y & e \\ \hline \text{Dufu} & \mathbf{pay\text{-}one\text{-}bill}(e_2) \\ \hline \end{array} \\ \\ \begin{array}{|c|} \hline y \\ \hline \text{Zilu} \\ \hline \end{array} \rightarrow \begin{array}{|c|c|} \hline y & e \\ \hline \text{Zilu} & \mathbf{pay\text{-}one\text{-}bill}(e_3) \\ \hline \end{array} \end{array} \rightarrow \begin{array}{|c|c|} \hline y & e \\ \hline \text{Libai} & \mathbf{pay\text{-}one\text{-}bill}(e_1) \\ \hline \text{Dufu} & \mathbf{pay\text{-}one\text{-}bill}(e_2) \\ \hline \text{Zilu} & \mathbf{pay\text{-}one\text{-}bill}(e_3) \\ \hline \end{array}$$

According to (22), the post-suppositional test introduced by *ge* checks the union of all the substates resulting from the D-Op, i.e., the collection of the substates. The post-supposition says: take the output states (28) as input states J, then generate possible output states J' such that for any subset of J', there is no output J'' in which (i) $[e']$ introduces a plurality of events across the cells of the index e'; (ii) $e' \sqsubseteq e$ checks that the plurality of events is a subset of the events in e stored in (28); (iii) $EC(e') > 1$ checks that there are at least two distinct values of e'; (iv) the sum of the values of e' is a paying-of-one-bill event.

The example with a possessive reflexive, i.e., (7), is derived as in (29). Following Brasoveanu (2008), we simply assume that reflexives introduce a variable indexed with their antecedents. In this example, the reflexive is indexed with a subscript and its antecedent is indexed with a superscript. As pointed out by Brasoveanu (2008), Quantifier Raising and Quantifying In are not needed in DPlL. Coindexation is enough because binding in a dynamic framework is taken care of by the explicit quantification over assignments built into the dynamic meaning of quantificational determiners.

(29) a. $[_{\text{CP}}$ ∃-closure $[_{\text{IP}}$ Everyone^x $[_{v\text{P}}$ v_{ag} $[_{\text{VP1}}$ *ge* $[_{\text{VP2}}$ pay $[_{\text{TH}}$ th [self$_x$-*de* bill]]]]]]

b. self$_x$-*de* bill $\rightsquigarrow \lambda Q[\mathbf{max}^y(\mathbf{bill}(y) \wedge \mathbf{of}(y,x)) \wedge Q(y)]$

c. TH $\rightsquigarrow \lambda V \lambda e[\mathbf{max}^y(\mathbf{bill}(y) \wedge \mathbf{of}(y,x)) \wedge V(e) \wedge \mathbf{th}(e) = y]$

d. VP1 $\rightsquigarrow \lambda e \left[\begin{array}{l} \mathbf{max}^y(\mathbf{bill}(y) \wedge \mathbf{of}(y,x)) \wedge \mathbf{pay}(e) \wedge \mathbf{th}(e) = y \wedge \\ \neg\exists e' \left[\begin{array}{l} e' \sqsubseteq e \wedge EC(e') > 1 \wedge \\ \exists e''[e'' = \oplus e' \wedge \mathbf{max}^y(\mathbf{bill}(y) \wedge \mathbf{of}(y,x)) \wedge \mathbf{pay}(e'') \wedge \mathbf{th}(e'') = y] \end{array}\right] \end{array}\right]$

e. CP $\rightsquigarrow \mathbf{max}^x(\mathbf{person}(x)) \wedge D(\exists e[\mathbf{VP1}(e) \wedge \mathbf{ag}(e) = x])]$

The dynamic version that the CP in (29e) abbreviates is (30), which briefly says: given a set of input states I, $\mathbf{max}^x(\mathbf{person}(x))$ introduces the maximum set of people across the cells of the index x; the D-Op divides the state into substates restricted by the value of x, checks that the value of x in each row is the agent of an event of paying x's bill, then collects the substates again. This update process can be illustrated as in (31), which is precisely what we want (see (13b)).

(30)
$$\mathbf{max}^x(\mathbf{person}(x)) \wedge$$
$$D\left(\begin{array}{l} [e] \wedge \mathbf{max}^y(\mathbf{bill}(y) \wedge \mathbf{of}(y,x)) \wedge \mathbf{pay}(e) \wedge \mathbf{ag}(e) = x \wedge \mathbf{th}(e) = y \\ \wedge \neg([e'] \wedge e' \sqsubseteq e \wedge EC(e') > 1 \wedge [e''] \wedge e'' = \oplus e' \wedge \mathbf{max}^y(\mathbf{bill}(y) \wedge \mathbf{of}(y,x)) \wedge \mathbf{pay}(e'') \wedge \mathbf{th}(e'') = y]) \end{array}\right)$$

(31)
$$\begin{array}{c} \begin{array}{|c|} \hline x \\ \hline \text{Libai} \\ \hline \text{Dufu} \\ \hline \text{Zilu} \\ \hline \end{array} \xrightarrow{\text{D}} \begin{array}{c} \begin{array}{|c|} \hline x \\ \hline \text{Libai} \\ \hline \end{array} \rightarrow \begin{array}{|c|c|} \hline x & e \\ \hline \text{Libai} & \textbf{pay-L's-bill}(e_1) \\ \hline \end{array} \\ \begin{array}{|c|} \hline x \\ \hline \text{Dufu} \\ \hline \end{array} \rightarrow \begin{array}{|c|c|} \hline x & e \\ \hline \text{Dufu} & \textbf{pay-D's-bill}(e_2) \\ \hline \end{array} \\ \begin{array}{|c|} \hline x \\ \hline \text{Zilu} \\ \hline \end{array} \rightarrow \begin{array}{|c|c|} \hline x & e \\ \hline \text{Zilu} & \textbf{pay-Z's-bill}(e_3) \\ \hline \end{array} \end{array} \rightarrow \begin{array}{|c|c|} \hline x & e \\ \hline \text{Libai} & \textbf{pay-L's-bill}(e_1) \\ \hline \text{Dufu} & \textbf{pay-D's-bill}(e_2) \\ \hline \text{Zilu} & \textbf{pay-Z's-bill}(e_3) \\ \hline \end{array} \end{array}$$

Applying the post-suppositional test to the output in (31), we can get the following result: no sum of any two or more events in (31) can be an event whose theme role is the bill of a person's and this person refers to Libai, Dufu or Zilu.

5. Conclusion

In this paper, we have argued that *ge* in Mandarin is not a distributive item. Rather, it is parasitic on a D-Op and introduces a post-suppositional test to ensure that the event predicate in the scope the D-Op exhibits anti-cumulativity, i.e., the sum of any non-singleton subset of events does not fall in the denotation of the event predicate. This paper can be seen as part of a more recent attempt to do justice to lexical items that have been lumped together under the label of distributive items, but nonetheless show peculiar properties unexpected from the D-op. We hope more and more non-standard distributive items in natural language can be discovered and given analyses in their own right.

Haoze Li, Jess H.-K. Law
haozeli@nyu.edu, jesshklaw@rutgers.edu

References

van den Berg, Martin. 1996. Some aspects of the internal structure of discourse. Doctoral dissertation, University of Amsterdam, Amsterdam.

Brasoveanu, Adrian. 2008. Donkey pluralities: Plural information states versus non-atomic individuals. *Linguistics and Philosophy* 31:129–209.

Brasoveanu, Adrian. 2013. Modified numerals as post-suppositions. *Journal of Semantics* 30:155–209.

Champollion, Lucas. 2015a. Every boy bought two sausages each: Distributivity and dependent numerals. In *Proceedings of the 32nd West Coast Conference on Formal Linguistics*, ed. Ulrike Steindl, Thomas Borer, Huilin Fang, Alfredo García Pardo, Peter Guekguezian, Brian Hsu, Charlie O'Hara, & Iris Chuoying Ouyang, 103–110. Somerville, MA: Cascadilla Proceedings Project.

Champollion, Lucas. 2015b. The interaction of compositional semantics and event semantics. *Linguistics and Philosophy* 38:31–66.

Choe, Jae-Woong. 1987. Anti-quantifiers and a theory of distributivity. Phd, University of Massachusetts-Amherst.

Henderson, Robert. 2014. Dependent indefinites and their post-suppositions. *Semantics & Pragmatics* 7:1–58.

Huang, C.-T.James, Y.-H. Audrey Li, & Yafei Li. 2009. *The syntax of chinese*. Cambridge: Cambridge University Press.

Krifka, Manfred. 1998. The origins of telicity. In *Events and grammar*, ed. Susan Rothstein, 197–236. Dordrecht: Kluwer Academic Publishers.

Landman, Fred. 2000. *Events and plurality: The jerusalen lectures*, volume 76 of *Studies in Linguistics and Philosophy*. Dordrecht: Kluwer.

Landman, Fred. 2004. *Indefinites and the type of sets*. Blackwell Publishing.

Lee, Po-lun, lei Zhang, & Haihua Pan. 2009. The distributive operator *ge* and some related issues. *Language and Linguistics* 10:293–314.

Lin, Jo-Wang. 1998. Distributivity in Chinese and its implications. *Natural Language Semantics* 6:201–243.

Lin, Tzong-Hong Jonah. 2004. Aspect, distributivity, and *wh*/qp interaction in Chinese. *Language and Linguistics* 5:615–642.

Link. 1987. *Generalized quantifiers*, volume 31, chapter Generalized quantifiers and plurals, 151–180.

Link, Godehard. 1998. *Algebraic Semantics in Language and Philosophy*. Stanford, California: CSLI Publications.

Nouwen, Rick. 2003. Plural pronominal anaphora in context: Dynamic aspects of quantification. Doctoral dissertation, Utrecht Univeristy, Utrecht, The Netherlands.

Safir, Ken, & Tim Stowell. 1988. Binominal *each*. In *Proceedings of North East Linguistic Society 8*, ed. James Blevins & Julie Carter, 426–450. Toronto: GLSA.

Zimmermann, Malte. 2002. Boys buying two sausages each: On the syntax and semantics of distance distributivity. Doctoral dissertation, Universiteit van Amsterdam, Amsterdam.