1. INTRODUCTION

- The Q(uestion)-particle *da* in (matrix) simple questions in Sinhala:
  - Wh-questions (WhQs): *da* mandatorily attaches to the wh-phrase
    
    (1) Chitra monəwa da gatte
    Chitra what da bought.E
    ’What did Chitra buy?’
    
    [Slade 2011: (2) p. 19]
  
  - Alternative questions (AltQs): *da* mandatorily attaches to each of the contrasting disjuncts
    
    (2) oyaa maalu.də mas.də kannə?
    you fish.də meat.də eat.E
    ’Did you eat meat† or fish?’
    
    [Weerasooriya 2019: (36) p. 12]
  
  - Polar questions (PolQs): *da* can attach to a specific XP (narrow focus) or be placed at the end of the clause (broad focus):
    
    (3) Chitra ee potə da kieuwe?
    Chitra that book da read.E
    ’Was it that book that Chitra read?’
    
    [Kishimoto 2005: (21a) p. 11]
  
    (4) Chitra ee potə kiuwa də?
    Chitra that book read.A də
    ’Did Chitra read that book?’
    
    [Kishimoto 2005: (21b) p. 11]

---

† The particle *da* is also used in declaratives with indefinites and with (exclusive) disjunction. For a recent analysis, see Weerasooriya (2019).
The Q-particle *da* in questions containing islands in Sinhala:

(Gair 1983, Cable 2010, Slade 2011, a.o.)

- **Wh-questions (WhQs):** *da* cannot occur inside the island; it attaches instead at the edge of the island (Cable 2010, Slade 2011):

'What, did Chitra hear the rumor that Ranjit bought it?' [Slade 2011: (9) p. 21]

(6) і Chitra [[Ranjit [monəwa] gatta] kiəna] katakataawə *da* æhuwe?
'What, did Chitra hear the rumor that Ranjit bought it?' [Slade 2011: (10) p. 21]

- **Alternative questions (AltQs): ****

- **Polar questions (PolQs): ****

The goal of this talk is two-fold:
- To fill the empirical gap and present novel data on the Q-particle *da* in AltQs and PolQs containing islands, and
- To develop a –so far tentative!– unified analysis of the meaning of *da* in all three question types that accounts for its distribution in the island cases

Idea in a nutshell:
- The semantic contribution of the Q-particle *da* –heading QP– is to mediate between the two “legs” of a semantic dependency (Hagstrom 1998, Cable 2010, Slade 2011).
- Previously: focus [.]f (Rooth 1992) + binding via a choice function variable
- Proposal here: focus [.]f (Rooth 1992) + focus [.]h (Kratzer 1991, Beck 2006)

Roadmap:
- §2 Previous analyses
- §3 Novel data
- §4 Proposal
- §5 Conclusions and outlook
2. Previous Analyses

2.1. Cable (2010) on WhQs

- Back to the data on WhQ with islands:

(9) *Chitra [Ranjit monəwa də gatta kianə kata kataawə] æhuwe?
    Chitra [Ranjit what də bought.A that rumour] heard.E
    'What, did Chitra hear the rumor that Ranjit bought t?'
    [Slade2011: (9) p. 21]

(10) Chitra [Ranjit monəwa gatta kianə kata kataawə] də æhuwe?
    Chitra [Ranjit what bought.A that rumour] də heard.E
    'What, did Chitra hear the rumor that Ranjit bought t?'
    [Slade2011: (10) p. 21]

- Syntax:
  - Seemingly wh-movement is not movement of the WhP per se but of the QP [[... wh...] də]. The movement of QP is triggered by the need to check a syntactic feature in the left periphery of WhQs: (11).
  - In simple WhQs, QP is typically located immediately above the WhP; but, due to s-selection constraints, it sometimes contains more material: e.g. [WhP P]-də in (12).
  - In WhQs with islands, the QP projected by də has to include the entire island, since no syntactic dependency –including movement– can hold across an island: (9)-(10).

(11) [Cable 2010:78]

(12) a. Chitra [kauru ekka] də kata kale?
    Chitra who with də talk did
    'Who did Chitra talk with?'

b. *Chitra [kauru də ekka] kata kale?
    Chitra who də with talk did

- Semantics (regardless of whether there is an island or not):
  - F(ocus)-marking XPr gives rise to a set of alternatives: [.]f (Rooth 1992).
    Interrogative wh-words are inherently F-marked (and have no ordinary value [.]p) (in the spirit of Beck 2006).
  - The Q-particle də bears an index i ranging over choice functions. The corresponding choice function f takes the Roothian [.]f of its syntactic sister and selects an element of that set.
  - The operator ForceQ binds the choice function f introduced by the index of də.
(13)

(14)

a. $\llbracket \text{who}_F \rrbracket^f = \{'\text{Chitra}', '\text{Guna}', '\text{Alis}', \ldots \}$

b. $\llbracket \text{who}_F \text{ with}_F \rrbracket^f = \{'\text{with Chitra}', '\text{with Guna}', '\text{with Alis}', \ldots \}$

c. $\llbracket \text{who}_F \text{ with}_F \text{ d}_1 \rrbracket^f = f (\{'\text{with Chitra}', '\text{with Guna}', '\text{with Alis}', \ldots\})$

d. $\llbracket \text{Chitra} [\text{who}_F \text{ with}_F \text{ d}_1 \text{ talk-did}_F] \rrbracket^f$
   $= \lambda w. \text{TALK}_{w} (\text{chitra}, f (\{'\text{with Chitra}', '\text{with Guna}', '\text{with Alis}', \ldots\})))$

e. $\llbracket \text{Force}_{Q,1} \text{ Chitra} [\text{who}_F \text{ with}_F \text{ d}_1 \text{ talk-did}_F] \rrbracket^f$
   $= \lambda p: \exists f[p = \lambda w. \text{TALK}_{w'} (\text{chitra}, f (\{'\text{with Chitra}', '\text{with Guna}', '\text{with Alis}', \ldots\}))) ]$
   $= \{'\text{that Chitra talked with Chitra}', '\text{that Chitra talked with Guna}', '\text{that Chitra talked with Alis}', \ldots \}$
2.2. Slade’s (2011) extension to AltQs and PolQs

**AltQs:**
- Challenge: Intuitively, in (15) we need to choose *once* from the set {gunapala, chitra}. Why then *two* occurrences of *do*?
- Slade’s (2011) attempt: (16)-(17)

(15) Gunapala *do* Chitra *do* gamata giye?
Gunapala *do* Chitra *do* village.Dat go.Past.E
'Did Gunapala or Chitra go to the village?' [Slade2011: (49) p. 100]

(16) [Slade2011: p. 101]

(17) Junction Rule [simplified here to match the types in (15)]
\[ \lambda x_e . \lambda f_{<et,e>}. \lambda y_e . \{y\} \cup \{ f(\{\lambda z_e . z\}(x)) \} \]

\[
\begin{align*}
gunapala & \quad \text{chitra} \\
\{\text{chitra}\} & \quad \text{chitra} \\
\{\text{chitra}\} & \quad \{\text{gunapala, chitra}\}
\end{align*}
\]

**Problem 1** for the choice function view of *do*:
No rationale or deeper explanation justifying multiple occurrences of *do*.

**PolQs**
- Challenge: No intuitive link between *do* and interrogativity in PolQs.
  Intuitively, (18) does not ask to choose from the set of alternatives {ranjit, chitra, alis…} of the *do*-marked DP; rather, (18) asks to choose between ‘yes’ and ‘no’.
- Slade’s (2011) idea: PolQs with narrow *do* are reduced to the corresponding AltQs (also with narrow *do*): (19)

(18) Ranjit* do* aawe?
Ranjit* do* come.Past.E
'Was it Ranjit who came?'

(19) [ Ranjit* do* came] (or) [not Ranjit* do* came]
3. **Novel Data on the Q-Particle in AltQs and PolQs with Islands**

- Recall again the data on WhQ with islands:

  (20) Complex NP-island:
  
  a. * Chitra [ranjit monowa da gatta kian katakatawa] aehuwe? (=9)
      Chitra [ranjit what da bought.A that rumour] heard.E
  

  'What did Chitra hear the rumor that Ranjit bought t?'

- AltQs:

  (21) Complex NP-island:
  
      John Chris da Ali da French speak do that rumour confirm did.E
  
  b. John [Chris French katha karanawa kiyana kathawa] da (nethnam)
      John Chris French speak do that rumour da (if not)
      Ali French speak do that rumour da confirm did.E

  'Did John confirm the rumour that Chris speaks French or that Ali speaks French?'

- Adjunct-island:

      John Chris da Ali da film see before meal ate.E
  
  b. John [Chris chitra.patiya balanna kalin] da (nethnam)
      John Chris film see before da (if not)
      [Ali chitra.patiya balanna kalin] da kema keewae
      Ali film see before da meal ate.E

  'Did John have dinner before Chris finished the film or before Ali finished the film?'

- PolQs:

  (22) Complex NP-island:
  
  a. John [Chris da French katha karanawa kiyana kathawa] thahawuru kale?
      John Chris da French speak do COMP rumour confirm did.E
  
  b. John [Chris French katha karanawa kiyana kathawa] da thahawuru kale?
      John Chris French speak do COMP rumour da confirm did.E

  'Was it Chris, that John confirmed the rumour that t speaks French?'

(24) Adjunct island:

a. John [Chris da chitra.patiya balanna kalin] kema keewae
   John Chris da film see before meal ate.E

b. John [Chris chitra.patiya balanna kalin] da kema keewae
   John Chris film see before da meal ate.E

'Was it Chris, that John had dinner before t finished the film?'

---

**Problem 2 for the choice function view of da:**

Given the distribution of da, PolQs cannot be reduced to AltQs. Hence, the challenge of da in PolQs is not solved: no intuitive link between da and interrogativity.

---

2 We thank Tharanga Weerasooriya, p.c, for the judgments in this section.
4. PROPOSAL

- Idea:

  We keep the general two-legged strategy (25) in the literature and modify it as in (26):
  For the upper leg, instead of using a choice function –selecting at a distance–, we will use Kratzerian focus values \[[.]^h\] which will combine…
  … not only with the operator \textbf{Force}_Q (for WhQs and for AltQs)
  … but also with the \textbf{squiggle operator} \sim (for AltQs and for PolQs).

\begin{align*}
\text{(25)} & & \text{CP} \\
& & \text{QP} \\
& & \text{YP} \\
& & \text{XP}_F \\
\end{align*}

\begin{align*}
\text{(26)} & & \text{CP} \\
& & \text{QP} \\
& & \text{YP} \\
& & \text{XP}_F \\
\end{align*}

- Three ingredients:

  ① Kratzerian focus framework  
  ② Discourse structure and F-marking in PolQ/AltQs (Roberts96, Biezma09, Meertens et al. 2019)  
  ③ Our proposed lexical entries

- **Ingredient ①:** Kratzerian focus framework
  - Each expression has an ordinary semantic value \[[.]\] and a focus semantic value \[[.]^h\].
  - The Focus feature F is indexed and its index is interpreted via assignment h.
  - Basic lexical entries for English and Functional Application rule:3

\begin{align*}
(27) & & a. \text{[John]} & = \text{john} \\
& & b. \text{[John]}^h & = \text{john} \\
(28) & & a. \text{[John}_{\text{F1}}\text{]} & = \text{john} \\
& & b. \text{[[John}_{\text{F1}}\text{]}^h & = h(1) \\
(29) & & a. \text{[[leave]} & = \lambda x.\lambda w.\text{LEAVE}_w(x) \\
& & b. \text{[[leave]}^h & = \lambda x.\lambda w.\text{LEAVE}_w(x) \\
\ & & & \Downarrow \\
(30) & & a. \text{[[John}_{\text{F1 left}}\text{]} & = \lambda w.\text{LEAVE}_w(j) \\
& & b. \text{[[John}_{\text{F1 left}}\text{]}^h & = \lambda w.\text{LEAVE}_w(h(1)) \\
\end{align*}

(31)  
\begin{align*}
(31) & & a. \text{[[who]} & = \text{#} & (i.e., undefined) \\
& & b. \text{[[who]}^h & = h(1) \\
(32) & & a. \text{[[who}_{\text{F1}}\text{]} & = \# & (i.e., undefined) \\
& & b. \text{[[who}_{\text{F1}}\text{]}^h & = h(1) \\
(33) & & a. \text{[[leave]} & = \lambda x.\lambda w.\text{LEAVE}_w(x) \\
& & b. \text{[[leave]}^h & = \lambda x.\lambda w.\text{LEAVE}_w(x) \\
\ & & & \Downarrow \\
(34) & & a. \text{[[who}_{\text{left}}\text{]} & = \# \\
& & b. \text{[[who}_{\text{left}}\text{]}^h & = \lambda w.\text{LEAVE}_w(h(1)) \\
\end{align*}

\begin{align*}
\text{Functional Application:} \\
\text{[[F A]} & = \text{[[F]}(\text{[[A]})) \\
\text{[[F A]}^h & = \text{[[F]}^h(\text{[[A]}^h)) \\
\end{align*}

\[3\] More precisely, \[[John_{\text{F1}}]}^h = h(1) \text{ if } 1 \in \text{Dom}(1) \text{ and } \text{[John}_{\text{F1}}\text{]}^h = \text{john} \text{ otherwise (see Beck 2006:fn6). Function h always starts up empty (Beck 2006:14) and grows as operators introduce new mappings (e.g., [[IP]}^h_{\text{PolQ}} in the text below). This will be relevant later for PolQs.}
Adding the \sim \sim-\text{operator} and the \text{Force}_Q \text{ operator:}

\[(36)\]
\begin{align*}
&\text{a. } \text{[[IP } \sim \text{ C ]] } = \{p: \exists x \ [p=\text{[[IP]]}^{hx}]\} \\
&\text{b. } \text{[[IP } \sim \text{ C ]]^h = \text{[[IP]]}^h}
\end{align*}

\[(37)\]
\begin{align*}
&\text{a. } \text{[[Force}_Q\text{, IP]] } = \{p: \exists x \ [p=\text{[[IP]]}^{hx}]\} \quad \text{[To be modified later]} \\
&\text{b. } \text{[[Force}_Q\text{, IP]]^h = \text{[[Force}_Q\text{, IP]]}
\end{align*}

\[(38)\] Q: Who left?
A: [John_F1 left ] \sim C

\[(39)\] \text{[CP Force}_Q\text{, [IP who_F1 left ]]

\[(40)\] \text{[[John_F1 left] } \sim \text{ C ]] = \text{defined only if } \text{[[C] } \subseteq \{p: \exists x [p=\text{[[John_F1 left]]}^{hx/1}]\}; \\
\text{[[C] } \subseteq \{p: \exists x [p=\text{[[John_F1 left]]}^{hx/1}(1)]\}; \\
\text{[[C} ] \subseteq \{p: \exists x [p=\text{[[John_F1 left]]}^{hx/1}(x)]\}; \\
\text{[[C} ] \subseteq \{\text{John, JOHN}, \text{ bill, } \text{Chris, ... } \}

\[(41)\] \text{[[Force}_Q\text{, [who_F1 left]]] = \{p: \exists x [p=\text{[[who_F1 left]]}^{hx/1}]\}; \\
\text{[[C] } \subseteq \{p: \exists x [p=\text{[[who_F1 left]]}^{hx/1}(1)]\}; \\
\text{[[C} ] \subseteq \{p: \exists x [p=\text{[[who_F1 left]]}^{hx/1}(x)]\}; \\
\text{[[C} ] \subseteq \{\text{John, JOHN}, \text{ bill, } \text{Chris, ... } \}

Building on and modifying this focus framework, we propose the following division of labor for Sinhala:

\[(42)\]
\begin{enumerate}
\item The focus feature \textbf{F} is expressed \textit{prosodically} by focal accent (or it is carried inherently by wh-word). It is modelled via the \textbf{Roothian} \text{[[.]]}^f.
\item The focus index \textit{i} is carried by the Q-particle. It is modelled via \textbf{Kratzerian} \text{[[.]]}^b
\end{enumerate}

\[(43)\]
\begin{align*}
&\text{a. } \text{[[Chitra_F] ] = chitra} \\
&\text{b. } \text{[[Chitra_F]]^f = } \{x: x \in D_c\}
\end{align*}

\[(44)\]
\begin{align*}
&\text{a. } \text{[[who_F]] = } # \quad \text{[To be modified later]} \\
&\text{b. } \text{[[who_F]]^f = } \{x: x \in D_c\}
\end{align*}

\[(45)\]
\begin{align*}
&\text{a. } \text{[[XP da_F]] = } \text{[[XP]} \\
&\text{b. } \text{[[XP da_F]]^h = } \lambda w: h(1) \in \text{[[XP]]}^\prime. h(1)
\end{align*}

Take-home message for Sinhala:
The focus index \textit{i} carried by the Q-particle will be targeted by \sim \sim-\text{operator} and/or by \text{Force}_Q depending on the question type.

\[4\] Beck (2006) defines the \sim \sim-\text{operator} as an unselective binder, as in (i). Here we have made it select index \textit{i}, for simplicity. Nothing hinges on this choice. Furthermore, Beck (2006) assumes that \sim \sim - \text{resets} \text{[[.]]}^p, as in (ii). We follow Romero (2015) in departing from this assumption.

(i) \text{[[IP } \sim \text{ C ]] is defined only if } \text{[[C] } \subseteq \{\text{[[IP]]}^h: h^r \in H \& h^r \text{ is total}\}; \text{if defined, then } \text{[[IP } \sim \text{ C ]] = } \text{[[IP]]}

(ii) \text{[[IP } \sim \text{ C ]]^h = } \text{[[IP]]}
INGREDIENT ②: Roberts’ (1996) discourse framework

- The structure of a discourse includes a hierarchically ordered set of implicit or explicit moves (questions and answers): (46)
- Following moves must be congruent with the preceding Question-under-Discussion (QUD): Q/A pairs like (38) and Q…Q sequences like (47)-(48).
- Congruence is secured by inserting the ~-operator in the corresponding LFs (simplified here from Roberts 1996): (49)-(50)

(46) 1. "Who[John,bill] left when[morning,afternoon]?")
   a. 'Who left in the morning?'
      i. 'Did John leave in the morning?'
      ii. 'Did Bill leave in the morning?'
   b. 'Who left in the afternoon?'
      i. 'Did John leave in the afternoon?'
      ii. 'Did Bill leave in the afternoon?'

(47) a. Who left? ✓ Did JOHN leave?
   b. Who left? # Did John LEAVE?

(48) a. Who left? ✓ Did JOHN or BILL leave?
   b. Who left? # Did John LEAVE or STAY?

(49) PolQ:
   a. [ ForceQ [IP JOHN F1 leave]~1C ]
   b. QUD/[C] ⊆ \{p: ∃x[p=[[JOHN F1 leave]]hv/1] = \{λw.LEAVE_w(john), λw.LEAVE_w(bill), λw.LEAVE_w(chris), ...\}

(50) AltQ:
   a. [ ForceQ [ [IP JOHN F1 leave]~1C or [IP Bill F1 leave]~1C ] ]
   b. QUD/[C] ⊆ \{p: ∃x[p=[[JOHN F1 leave]]hv/1] = \{λw.LEAVE_w(john), λw.LEAVE_w(bill), λw.LEAVE_w(chris), ...\}
   c. QUD/[C] ⊆ \{p: ∃x[p=[[BILL F1 leave]]hv/1] = \{λw.LEAVE_w(john), λw.LEAVE_w(bill), λw.LEAVE_w(chris), ...\}

Take-home message for Sinhala:
In PolQs and AltQs, the ~ operator will target the focus index i carried by the Q-particle.

This circumvents problems 1 and 2 of the choice function view:
1. In AltQs, one occurrence of də per disjunct, since each disjunct must check discourse congruence via its ~-operator
2. In PolQs, də does not serve ForceQ but just ~-operator, so no relation between XP-də and interrogativity or answer choices.
**INGREDIENT**: Our proposed lexical entries

(51) The operator $\text{Force}_{Q,i}$ for WhQs and AltQs:
   a. $\langle \text{Force}_{Q,i_1 \ldots j_n}, \text{IP} \rangle = \lambda p: p \in \text{[[IP]]} \text{ if } \text{[[IP]]} \text{ is defined. } \exists x_1, \ldots, y_n \text{ [p=\text{[[IP]]}^{h_{x_1 \ldots y_n}}]}$
   b. $\langle \text{Force}_{Q,i_1 \ldots j_n}, \text{IP} \rangle^h = \text{[[Force}_{Q,i_1 \ldots j_n} \text{IP}]]$

(52) The operator $\text{Force}_{Q}$ for PolQs:
   a. $\langle \text{Force}_{Q}, \text{IP} \rangle = \lambda p: p = \text{[[IP]]}^h$
   b. $\langle \text{Force}_{Q}, \text{IP} \rangle^h = \text{[[Force}_{Q,i_1 \ldots j_n} \text{IP}]]$

(53) The squiggle operator $\sim$:
   a. $\langle \text{IP} \sim \text{C} \rangle$ is defined only if $\langle \text{C} \rangle \subseteq \{p: \exists x \text{ [p=\text{[[IP]]}^{h_{x}]}]}\};$
      \text{if defined, then } \langle \text{IP} \sim \text{C} \rangle = \langle \text{IP} \rangle$
   b. $\langle \text{IP} \sim \text{C} \rangle^h = \langle \text{IP} \rangle^h$

(54) Disjunction $\text{or}$:
   a. $\langle \text{IP}_1 \text{ or } \text{IP}_2 \rangle = \{ \langle \text{IP}_1 \rangle, \langle \text{IP}_2 \rangle \}$ (Alonso-Ovalle 2006, a.o.)
   b. $\langle \text{IP}_1 \text{ or } \text{IP}_2 \rangle^h = \langle \text{IP}_1 \rangle^h \cup \langle \text{IP}_2 \rangle^h$

**Take-home message for Sinhala:**
In WhQs and AltQs, $\text{Force}_{Q}$ will target the focus index $i$ carried by the Q-particle.

---

---

---

---

5 $\text{Force}_{Q,i}$ for WhQs and AltQs and $\text{Force}_{Q}$ for PolQs are not unified into a single lexical entry at this point (though note that the blue parts can be easily unified). We leave this for future work.
Sample derivation of WhQ with island in Sinhala:

(55) a. \([\text{DP}]\) = \# \\
b. \([\text{DP}]^\text{f}\) = \{ \lambda w. uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{chitra, french})] \\
\quad \lambda w. uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{guna, french})], \\
\quad \lambda w. uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{ali, french})], \ldots \}

(56) a. \([\text{DP} \text{ da}_1]\) = \# \\
b. \([\text{DP} \text{ da}_1]^\text{h}\) = \{ \lambda w. h(1) \in [\text{DP}]^\text{f}. h(1) \\
\quad = \lambda w. h(1) \in \{ \lambda w. uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{chitra, fr})] \cdot h(1) \\
\quad \lambda w. uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{guna, fr})], \\
\quad \lambda w. uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{ali, fr})], \ldots \}

(57) a. \([\text{FocP}]\) = \# \\
b. \([\text{FocP}]^\text{h}\) = \lambda w. h(1) \in [\text{DP}]^\text{f}. \text{CONFIRM}_w(j, h(1)) \\
\quad = \lambda w. h(1) \in \{ \lambda w. uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{chitra, fr})] \cdot \text{CONFIRM}_w(j, h(1)) \\
\quad \lambda w. uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{guna, fr})], \\
\quad \lambda w. uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{ali, fr})], \ldots \}

(58) \([\text{Force}_{0,1} \text{FocP}]\) \\
\quad = \lambda p. p = [\text{FocP}] \text{ if } [\text{FocP}] \text{ is defined. } \exists x [p = [\text{FocP}]^\text{h}_{x/1}] \\
\quad = \lambda p. \exists x [p = [\text{FocP}]^\text{h}_{x/1}] \\
\quad = \lambda p. \exists x [p = \lambda w. h^x(1) \in \{ \lambda w. uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{chitra, fr})] \cdot \text{CONF}_w(j, h^x(1)) \\
\quad \lambda w. uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{guna, fr})], \\
\quad \lambda w. uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{ali, fr})], \ldots \} \\
\quad = \lambda p. \exists x [p = \lambda w. x : \{ \lambda w. uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{chitra, fr})] \cdot \text{CONF}_w(j, x) \\
\quad \lambda w. uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{guna, fr})], \\
\quad \lambda w. uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{ali, fr})], \ldots \} \\
\quad = \{ \lambda w. \text{CONFIRM}_w(j, uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{chitra, french})]), \\
\quad \lambda w. \text{CONFIRM}_w(j, uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{guna, french})]), \\
\quad \lambda w. \text{CONFIRM}_w(j, uq [\text{RUMOR}_w(q) \land q = \lambda w'.\text{SPK}_w'(\text{ali, french})]), \ldots \}
Sample derivation of an AltQ with island in Sinhala:

(60) First FocP:
   a. \[ [\text{FocP}]^a = \lambda w. \text{CONFIRM}_w(j, t_q \text{RUMOR}_w(q) \land q = \lambda w'. \text{SPK}_w(\text{chitra, fr})] \]
   b. \[ [\text{FocP}]^h = \lambda w: h(1) \in [\text{DP}]^f. \text{CONFIRM}_w(j, h(1)) \]
      \[ = \lambda w: h(1) \in \{ \lambda w. t_q \text{RUMOR}_w(q) \land q = \lambda w'. \text{SPK}_w(\text{chitra, fr}) \}. \text{CONFIRM}_w(j, h(1)) \]
      \[ \lambda w. t_q \text{RUMOR}_w(q) \land q = \lambda w'. \text{SPK}_w(\text{guna, fr}), \]
      \[ \lambda w. t_q \text{RUMOR}_w(q) \land q = \lambda w'. \text{SPK}_w(\text{ali, fr}), \ldots \} \]

(61) Second FocP:
   a. \[ [\text{FocP}]^a = \lambda w. \text{CONFIRM}_w(j, t_q \text{RUMOR}_w(q) \land q = \lambda w'. \text{SPK}_w(\text{guna, fr})] \]
   b. \[ [\text{FocP}]^h = \lambda w: h(1) \in [\text{DP}]^f. \text{CONFIRM}_w(j, h(1)) \]
      \[ = \lambda w: h(1) \in \{ \lambda w. t_q \text{RUMOR}_w(q) \land q = \lambda w'. \text{SPK}_w(\text{chitra, fr}) \}. \text{CONFIRM}_w(j, h(1)) \]
      \[ \lambda w. t_q \text{RUMOR}_w(q) \land q = \lambda w'. \text{SPK}_w(\text{guna, fr}), \]
      \[ \lambda w. t_q \text{RUMOR}_w(q) \land q = \lambda w'. \text{SPK}_w(\text{ali, fr}), \ldots \} \]

(62) a. \[ [\text{FocP} \sim_1 \text{C}]^a \text{ is defined only if} \]
      \[ [\text{C}] \subseteq \{ p : \exists x [p = [\text{IP}]^{\text{h}x}] \}, \text{i.e.,} \]
      \[ [\text{C}] \subseteq \{ \lambda w. \text{CONFIRM}_w(j, t_q \text{RUMOR}_w(q) \land q = \lambda w'. \text{SPK}_w(\text{chitra, fr})], \}
      \[ \lambda w. \text{CONFIRM}_w(j, t_q \text{RUMOR}_w(q) \land q = \lambda w'. \text{SPK}_w(\text{guna, fr})], \}
      \[ \lambda w. \text{CONFIRM}_w(j, t_q \text{RUMOR}_w(q) \land q = \lambda w'. \text{SPK}_w(\text{ali, fr})], \ldots \} \]
   b. \[ [\text{FocP} \sim_1 \text{C}]^h \]
      \[ = \lambda w: h(1) \in [\text{DP}]^f. \text{CONFIRM}_w(j, h(1)) \]
      \[ = \lambda w: h(1) \in \{ \lambda w. t_q \text{RUMOR}_w(q) \land q = \lambda w'. \text{SPK}_w(\text{chitra, fr}) \}. \text{CONFIRM}_w(j, h(1)) \]
      \[ \lambda w. t_q \text{RUMOR}_w(q) \land q = \lambda w'. \text{SPK}_w(\text{guna, fr}), \]
      \[ \lambda w. t_q \text{RUMOR}_w(q) \land q = \lambda w'. \text{SPK}_w(\text{ali, fr}), \ldots \} \]

(63) a. \[ [\text{FocP or FocP}] = \{ [\text{FocP}], [\text{FocP}] \} \]
   b. \[ [\text{FocP or FocP}]^h = [\text{FocP}]^h \cup [\text{FocP}]^h \]
      \[ = [\lambda w: h(1) \in [\text{DP}]^f. \text{CONFIRM}_w(j, h(1))]. \cup \]
      \[ [\lambda w: h(1) \in [\text{DP}]^f. \text{CONFIRM}_w(j, h(1)) \]
      \[ = \lambda w: h(1) \in [\text{DP}]^f. \text{CONFIRM}_w(j, h(1)) \]
      \[ = (61.b) = (62.b) \]
(65) \[ [[\text{Force}_{0,1} \ [\text{FocP or FocP}]]] \]
\[ = \lambda p: \text{pe}[[[\text{FocP or FocP}]] \text{ if } [[\text{FocP or FocP}]] \text{ is defined. } \exists x \ [p=\text{[[FocP or FocP]]}^{h^v/1}] \]
\[ = \lambda p: p \in \{\lambda w. \text{CONFIRM}_{w}(j, 1q \ [\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w'}(\text{chitra, fr})]), \}
\[ \lambda w. \text{CONFIRM}_{w}(j, 1q \ [\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w'}(\text{guna, fr})])\}
\[ \exists x \ [p = \lambda w: h^{x/1}(1) \in [[\text{DP}]]^f. \text{CONFIRM}_{w}(j, h^{x/1}(1))\]
\[ = \lambda p: p \in \{\lambda w. \text{CONFIRM}_{w}(j, 1q \ [\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w'}(\text{chitra, fr})]), \}
\[ \lambda w. \text{CONFIRM}_{w}(j, 1q \ [\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w'}(\text{guna, fr})])\}
\[ \exists x \ [p = \lambda w: x \in [[\text{DP}]]^f. \text{CONFIRM}_{w}(j, x)\]
\[ = \lambda p: p \in \{\lambda w. \text{CONFIRM}_{w}(j, 1q \ [\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w'}(\text{chitra, fr})]), \}
\[ \lambda w. \text{CONFIRM}_{w}(j, 1q \ [\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w'}(\text{guna, fr})])\}
\[ p \in \{\lambda w. \text{CONFIRM}_{w}(j, 1q \ [\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w'}(\text{chitra, fr})]), \}
\[ \lambda w. \text{CONFIRM}_{w}(j, 1q \ [\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w'}(\text{guna, fr})])\],
\[ \lambda w. \text{CONFIRM}_{w}(j, 1q \ [\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w'}(\text{ali, fr})]), \ldots\} \]
Sample derivation of a PolQ with island in Sinhala:

\[
\begin{align*}
\text{(66)} & \quad \text{[ForceQ] } \text{IP } \sim_{1} C \\
\text{(67)} & \quad \text{[ForceQ] } \text{IP } \sim_{1} C
\end{align*}
\]

\[
\begin{align*}
& a. \quad \text{[Chitra]} = \text{chitra} \\
& b. \quad \text{[Chitra]} = \{x : x \in D_{c}\}
\end{align*}
\]

\[
\begin{align*}
& a. \quad \text{[IP]} = \lambda w. \text{CONFIRM}_{w}(j, 1q [\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w}(\text{chitra, fr})]) \\
& b. \quad \text{[IP]}^{h} = \lambda w : h(1) \in \{z : z \in D_{c}\}. \text{CONF}_{w}(j, 1q [\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w}(h(1), \text{fr})])
\end{align*}
\]

\[
\begin{align*}
& a. \quad \text{[IP } \sim_{1} C\text{]} \quad \text{is defined only if} \\
& \quad \text{[C]} \subseteq \{p : \exists x [p = [\text{IP}^{h}(x,j)]}\}; \\
& \quad \text{[C]} \subseteq \{ p : \exists x [p = \lambda w : h^{w}(1) \in \{z : z \in D_{c}\}. \text{CONF}_{w}(j, 1q [\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w}(h^{w}(1), \text{fr})])}\}; \\
& \quad \text{[C]} \subseteq \{ p : \exists x [p = \lambda w : x \in D_{c}. \text{CONF}_{w}(j, 1q [\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w}(x, \text{fr})])}\}; \\
& \quad \text{[C]} \subseteq \{ \lambda w. \text{CONFIRM}_{w}(j, 1q[\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w}(\text{chitra, fr})]), \lambda w. \text{CONFIRM}_{w}(j, 1q[\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w}(\text{guna, fr})]), \lambda w. \text{CONFIRM}_{w}(j, 1q[\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w}(\text{ali, fr})]), \ldots\}; \\
& b. \quad \text{[IP } \sim_{1} C\text{]}^{h} = \text{[IP]}^{h}
\end{align*}
\]

\[
\begin{align*}
& \quad \text{[ForceQ IP]} = \lambda p. \quad p = \text{[IP]}^{h} \\
& \quad = \lambda p. \quad p = \lambda w. \text{CONFIRM}_{w}(j, 1q [\text{RUMOR}_{w}(q) \land q = \lambda w'. \text{SPK}_{w}(\text{chitra}^{6}, \text{fr})])
\end{align*}
\]

\[
\begin{align*}
^{6} \text{See footnote 3.}
\end{align*}
\]
5. CONCLUSIONS AND OUTLOOK

- In a prominent line of work (Hagstrom 1998, Cable 2010, Slade 2011), Q-particles like Sinhala \( də \) have been analyzed as introducing a choice function that mediates between the Roothian focus value \([.]^f\) and the Force\(Q\) operator.

- This line of work has been shown to face (at least) two problems:
  - For AltQs, there is no rationale for the multiple use of \( də \) when we are intuitively choosing only once.
  - In PolQs, \( də \) is intuitively not choosing from the focus value \([.]^f\) of its syntactic sister. Trying to reduce PolQs to partially elided AltQs to avoid this problem fails to account for the asymmetric distribution of \( də \) in the two question types.

- A new analysis has tentatively been proposed whereby the Q-particles \( də \) mediates between two focus percolation systems: Roothian focus value \([.]^f\) and Kratzerian focus value \([.]^h\). The Kratzerian focus value \([.]^h\) will serve not only the Force\(Q\) operator (in WhQs and AltQs) but also the \(\sim\)-operator (in AltQs and PolQs).

- This new analysis circumvents the two problems faced by the choice function view:
  - In AltQs, two \( də \) particles are present because we check congruence with the previous discourse via the \(\sim\)-operator twice, once per disjunct.
  - In PolQs, \( də \) does not link to Force\(Q\) (hence, no link to interrogativity or choice of answer) but just to the \(\sim\)-operator for discourse congruence.

- For the future:
  - Extension to Q-particles in so-called Q-adjunction languages like Japanese and Korean.
  - Comparison of \( də \) in questions with \( də \) with indefinites.
REFERENCES


