1 Ellipsis

(1) Dan likes Golf, and George does, too.

- syntactic rule: $\|[_{VP}\text{ does}]\| = \lambda x \lambda s.P(s,x)$ (P some free property variable
- identify parallel elements: Dan / George
- mark *primary occurrence* of parallel element in semantic representation of antecedent clause

$$\lambda s.\text{LIKE'}(s, \underline{\text{D'}}, \text{GOLF'})$$

- compositional interpretation
  - $\lambda s.\text{LIKE'}(s, \underline{\text{D'}}, \text{GOLF'})$
  - $\lambda s.P(s, \text{G'})$
- Parallelism constraint

$$\begin{aligned} \lambda s.\text{LIKE'}(s, \underline{\text{D'}}, \text{GOLF'}) &= \phi(\text{D'}) \\ \lambda s.P(s, \text{G'}) &= \phi(\text{B'}) \end{aligned}$$

- solve equation:

$$\begin{aligned} \phi &\mapsto \lambda x \lambda s.\text{LIKE'}(s, \underline{\text{D'}}, \text{GOLF'}) \\ \phi &\mapsto \lambda x \lambda s.\text{LIKE'}(s, x, \text{GOLF'}) \end{aligned}$$

- in the correct solution, the primary occurrence must not occur

$$\phi \mapsto \lambda x \lambda s.\text{LIKE'}(s, x, \text{GOLF'})$$

- compute pragmatic interpretation of ellipsis clause

$$\phi(\text{B'}) = \lambda s.\text{LIKE'}(s, \text{B'}, \text{GOLF'})$$

- ellipsis resolution:

$$P = \text{LIKE'}(\text{GOLF'})$$

(2) Dan likes his wife, and George does, too.

- $\|\text{his}_i\| = \lambda P.\text{OF'}(x_i, P)$
- compositional interpretation:
  - Dan likes his_1 wife. $\rightsquigarrow \lambda s.\text{LIKE'}(s, \underline{\text{D'}}, \text{OF'}(x_1, \text{WIFE'}))$
  - George does (too). $\rightsquigarrow \lambda s.P(s, \text{G'})$
- contextual constraint: $x_1 = \text{D'}$

- replace all contextually constraint variables by constants:

$$\text{Dan likes his wife.} \rightsquigarrow \lambda s.\textsc{like'}(s, \underline{\textsc{d'}}, \textsc{of'}(\textsc{d'}, \textsc{wife'}))$$

- Parallelism:

$$\begin{aligned} \lambda s.\textsc{like'}(s, \underline{\textsc{d'}}, \textsc{of'}(\textsc{d'}, \textsc{wife'})) &= \phi(\textsc{d'}) \\ \lambda s.P(s, \textsc{g'}) &= \phi(\textsc{g'}) \end{aligned}$$

- solve equation:

$$\begin{aligned} \phi_1 &\mapsto \lambda x \lambda s.\textsc{like'}(s, x, \textsc{of'}(\textsc{d'}, \textsc{wife'})) \\ \phi_2 &\mapsto \lambda x \lambda s.\textsc{like'}(s, x, \textsc{of'}(x, \textsc{wife'})) \end{aligned}$$

- interpretations of elliptical clause

$$\begin{aligned} \phi_1(\textsc{g'}) &= \lambda s.\textsc{like'}(s, \textsc{g'}, \textsc{of'}(\textsc{d'}, \textsc{wife'})) \\ \phi_2(\textsc{g'}) &= \lambda s.\textsc{like'}(s, \textsc{g'}, \textsc{of'}(\textsc{g'}, \textsc{wife'})) \end{aligned}$$

- ellipsis resolution:

$$P_1 = \textsc{like'}(\textsc{of'}(\textsc{d'}, \textsc{wife'})) \quad (1)$$
$$P_2 = \textsc{like'}(\textsc{of'}(\textsc{g'}, \textsc{wife'})) \quad (2)$$

(3) John's mother adores him, and Bill's mother does, too.

- antecedent clause: $\lambda s.\textsc{adore'}(s, \underline{\textsc{of'}(\underline{\textsc{j'}}, \textsc{mother'})}, \textsc{j'})$
- ellipsis clause: $\lambda s.P(s, \textsc{of'}(\textsc{b'}, \textsc{mother'}))$
- Parallelism constraints:

$$\begin{aligned} \lambda s.\textsc{adore'}(s, \underline{\textsc{of'}(\underline{\textsc{j'}}, \textsc{mother'})}, \textsc{j'}) &= \phi(\textsc{j'}, \textsc{of'}(\textsc{j'}, \textsc{mother'})) \\ \lambda s.P(s, \textsc{of'}(\textsc{b'}, \textsc{mother'})) &= \phi(\textsc{b'}, \textsc{of'}(\textsc{b'}, \textsc{mother'})) \end{aligned}$$

- solutions:

$$\begin{aligned} \phi_1 &\mapsto \lambda y \lambda x \lambda x.\textsc{adore'}(s, y, \textsc{j'}) \\ \phi_2 &\mapsto \lambda y \lambda x \lambda x.\textsc{adore'}(s, y, x) \end{aligned}$$

- interpretation of elliptical clause

$$\begin{aligned} \phi_1(\textsc{b'}, \textsc{of'}(\textsc{b'}, \textsc{mother'})) &= \lambda s.\textsc{adore'}(s, \textsc{of'}(\textsc{b'}, \textsc{mother'}), \textsc{j'}) \\ \phi_1(\textsc{b'}, \textsc{of'}(\textsc{b'}, \textsc{mother'})) &= \lambda s.\textsc{adore'}(s, \textsc{of'}(\textsc{b'}, \textsc{mother'}), \textsc{b'}) \end{aligned}$$

- ellipsis resolution:

$$\begin{aligned} P_1 &= \text{ADORE'}(\text{J'}) \\ P_2 &= \text{ADORE'}(\text{B'}) \end{aligned}$$

(4) An U follows each X, and a Y does, too.

- Compositional interpretation:

$$\lambda s.\forall x(X(s,x) \to \exists y(U(s,y) \wedge \text{FOLLOW'}(s,x,y)))$$

$$\lambda s.\exists y(Y(s,y) \wedge P(s,y))$$

- Parallelism:

$$\begin{aligned} \lambda s.\forall x(X(s,x) \to \exists y(U(s,y) \wedge \text{FOLLOW'}(s,x,y))) &= \phi(\lambda Q \lambda s.\exists y(U(s,y))) \\ \lambda s.\exists y(Y(s,y) \wedge P(s,y)) &= \phi(\lambda Q \lambda s.\exists y(Y(s,y))) \end{aligned}$$

- Solution:

$$\phi = \lambda T \lambda s.\forall x(X(s,x) \to T(\lambda z \lambda s.\text{FOLLOW'}(s,x,z)))$$

- elliptical clause:

$$\phi(\lambda Q \lambda s.\exists y(Y(s,y))) = \lambda s.\forall x(X(s,x) \to \exists y(Y(s,y) \wedge \text{FOLLOW'}(s,x,y)))$$

- ellipsis resolution:

$$P = \phi$$

ACD

(5) John read every book_1 that Paul did.

Prohibition of non-vacuous binding: Every λx_i that corresponds to a movement operation must bind at least one variable.