Filling gaps with Glue
Adam Przepiórkowski† and Agnieszka Patejuk
†ICŚ Polish Academy of Sciences, ‡University of Warsaw

Problem
Consider the simple example of gapping (Ross 1970) in (1) and its intended neo-Davidsonian (Davidson 1967, Castañeda 1967, Parsons 1990) representation in (2), where the constants m, l, h, and b refer to Marge, Lisa, Homer, and Bart, respectively. The representation in (2) reflects the fact that (1) refers to two seeing events, not just one. The first bracketed conjunct in (2) faithfully represents the first clause of (1) (i.e., Marge saw Lisa).

(1) Marge saw Lisa and Homer – Bart.
(2) \[\exists e. \text{see}(e) \land \text{agent}(e, m) \land \text{theme}(e, l) \land \exists e. \text{see}(e) \land \text{agent}(e, h) \land \text{theme}(e, b)]

The challenge for any semantic theory of gapping is to – compositionally – obtain the second bracketed conjunct in (2) as the representation of the gapped clause in (1) (i.e., Homer – Bart), despite the missing verb.

A solution available within theories relying on empty constituents is that the gapped clause contains a covert verb with the right semantics – this could be an unpronounced copy of the overt verb in the first clause, or some generic phonetically empty verb anaphorically related to the overt verb. However, LFG eschews empty constituents unless there are good syntactic reasons to postulate them (Bresnan et al. 2016: §9.5). The aim of this paper is to provide an LFG+Glue (Dalrymple 1999, Asudeh 2022) analysis of simple gapping structures that does not assume empty constituents.

Solution
Glue is resource-sensitive, so the semantic representation see introduced by the predicate saw in (1) must be used exactly once in the glue proof. The question is how to obtain the representation in (2), where see occurs twice.

A somewhat similar problem occurs when a dependent introduces a resource that is shared between conjointed predicates, as in (3)–(4). The semantic representation of Bart, i.e., b, occurs twice in (4). The standard solution in such cases is that the representation of the shared dependent, b, is an argument of the representation of the coordination walked and whistled, i.e., of \(x.\) \([\exists e. \text{walk}(e) \land \text{agent}(e, x)] \land [\exists e. \text{whistle}(e) \land \text{agent}(e, x)],\) resulting in (4).

(3) Bart walked and whistled.
(4) \[\exists e. \text{walk}(e) \land \text{agent}(e, b)] \land [\exists e. \text{whistle}(e) \land \text{agent}(e, b)]

For a similar solution to work in gapping, the representation of the overt verb introduced in the first clause in (1) should be factored out and become an argument of the representation of the rest of the coordination, as schematically shown in (5):

(5) \[\lambda f. [\exists e. f(e) \land \text{agent}(e, m) \land \text{theme}(e, l)] \land [\exists e. f(e) \land \text{agent}(e, h) \land \text{theme}(e, b)]\] (see \(\equiv\) Glue)

We flesh out this idea by building on Champollion’s (2015) approach to event semantics and on Patejuk and Przepiórkowski’s (2017) LFG analysis of the syntax of gapping.

Semantics
Champollion (2015) argues that, instead of the usual existential closure at the level of the clause, the existential quantifier which binds the event should be introduced by the verb itself, as in (6).

(6) saw \(\sim\) \(\lambda f. \exists e. \text{see}(e) \land f(e)\)

Dependents are analysed as semantic modifiers, as in (7)–(8), combining with verbs as in (9)–(10).

(7) Marge_{agent} \(\sim\) \(\lambda V. \lambda f. (\exists e. \text{agent}(e, m) \land f(e))\)
(8) Lisa_{theme} \(\sim\) \(\lambda V. \lambda f. (\exists e. \text{theme}(e, l) \land f(e))\)
(9) saw Lisa \(\sim\) \(\beta\)-reduction \(\lambda f. \exists e. \text{see}(e) \land \text{theme}(e, l) \land f(e)\)
(10) Marge saw Lisa \(\sim\) \(\lambda f. \exists e. \text{see}(e) \land \text{theme}(e, l) \land \text{agent}(e, m) \land f(e)\)

The final relevant ingredient of Champollion 2015 is the event closure in (11) (replacing the usual existential closure), leading to the final representation of the sentence Marge saw Lisa in (12), equivalent to the first bracketed conjunct in (2).

(11) \([\text{closure}]\) \(\sim\) \(\lambda e. \text{true}(e)\)
(12) Marge saw Lisa (after closure) \(\sim\) \(\beta\)-reduction \(\lambda f. [\exists e. \text{see}(e) \land \text{theme}(e, l) \land \text{agent}(e, m) \land \text{true}(e)]\)

For this approach to work in the analysis of gapping, semantic representations of verbs such as (6) must be split:

(13) saw \(\sim\) \(\lambda V. \lambda f. (\exists e. \text{see}(e) \land f(e))\)
(14) \(\lambda f. \exists e. f(e)\)

We assume lexical verbs introduce two meaning constructors (MCs) corresponding to meaning representations in (13)–(14), while the gapped clause constructionally introduces a constructor corresponding to (14). In the case of the running example (1), when this MC combines with the agent Homer and the theme Bart, the representation of the gapped clause is (15):

(15) Homer – Bart (initial) \(\sim\) \(\lambda f. \exists e. \text{theme}(e, b) \land \text{agent}(e, h) \land f(e)\)

Similarly, (16) is the initial representation of the first clause, utilizing only the MC corresponding to (14) (without (13)).

(16) Marge saw Lisa (initial) \(\sim\) \(\lambda f. \exists e. \text{theme}(e, l) \land \text{agent}(e, m) \land f(e)\)

Assuming the standard approach to the semantics of coordination (Partee and Rooth 1983, Dalrymple et al. 2019: §16.7), conjoining (16) and (15) results in (17):

(17) Marge saw Lisa and Homer – Bart (initial) \(\sim\)
\(\lambda f. [\exists e. \text{theme}(e, l) \land \text{agent}(e, m) \land f(e)] \land [\exists e. \text{theme}(e, b) \land \text{agent}(e, h) \land f(e)]\)

Applying the representation of the verb in (13) to the representation of coordination in (17) results in (18):

(18) Marge saw Lisa and Homer – Bart (before closure) \(\sim\) \(\beta\)-reduction \(\beta\)-reduction \(\lambda f. [\exists e. \text{theme}(e, l) \land \text{agent}(e, m) \land \text{see}(e) \land f(e)] \land [\exists e. \text{theme}(e, b) \land \text{agent}(e, h) \land \text{see}(e) \land f(e)]\)

This derivation is slightly more complex than the schematic derivation in (5), because in (18) the representation of the verb is the functor rather than the argument. Applying (18) to the closure representation in (11) results in (19), which is equivalent to the desired representation (2) of the running example (1).

(19) Marge saw Lisa and Homer – Bart (after closure) \(\sim\) \(\beta\)-reduction \(\beta\)-reduction \(\lambda f. [\exists e. \text{theme}(e, l) \land \text{agent}(e, m) \land \text{see}(e) \land \text{true}(e)] \land [\exists e. \text{theme}(e, b) \land \text{agent}(e, h) \land \text{see}(e) \land \text{true}(e)]\)
Syntax–Semantics Interface  For the syntax–semantics interface, this proposal relies on the analysis of gapping in Patejuk and Przepiıkowski 2017: (21) is the partial f-structure of the gapped clause in (1), while (20) is the partial f-structure of the first clause where dependents that are not shared are put in local (rather than in the main f-structure e).

We assume that proper nouns introduce the usual MCs, e.g., \( m : \uparrow_e \) for Marge. Thematic roles are introduced constructionally; for example, in a rule assigning the structure of a dependent to the subj value of the verb, the dependent is also assumed to be an agent (perhaps as one of the options) and assigned the following MC:

\[
(24) \lambda \varepsilon. \lambda f. V(\lambda \varepsilon, agent(e,x) \land f(e)) : \uparrow_e \rightarrow \uparrow f_e \rightarrow \uparrow f_t
\]

These two constructors, \( m : \uparrow_e \) and (24), in the instantiated MC for the agent Marge in (25), where \( c^t \) refers to the relevant f-structure in (23), and analogously for the theme Lisa (as well as for the agent Homer and the theme Bart – in these cases \( \uparrow \) instantiates to \( c^t \)).

\[
(25) \lambda \varepsilon. \lambda f. V(\lambda \varepsilon, agent(e,m) \land f(e)) : \uparrow_e \rightarrow \uparrow f_e \rightarrow \uparrow f_t
\]

The constructor introduced under the last IP in (22) in this case instantiates to (26) and it combines with MCs for the agent Homer and the theme Bart (similar to that in (25), with \( c^t \)), resulting in the MC (27) corresponding to the gapped clause.

\[
(26) \lambda f. \exists e. f(e) : (c^t_e \rightarrow c^t_e) \rightarrow c^t_{f_t}
\]

\[
(27) \lambda f. \exists e. theme(e,b) \land agent(e,h) \land f(e) : (c^t_{f_t} \rightarrow c^t_{f_t}) \rightarrow c^t_{f_t}
\]

Lexical verbs introduce the following three MCs – the first one, in (28), corresponds to the closure in (11), while the next two, in (29)–(30), correspond to the representations of the semantics of verbs in (13)–(14):

\[
(28) \lambda v \rightarrow \lambda e. true(e) : \uparrow_v \rightarrow \uparrow t_e
\]

\[
(29) \lambda v. \lambda f. V(\lambda e. see(e) \land f(e)) : \uparrow_v \rightarrow \uparrow f_v \rightarrow \uparrow f_t \rightarrow \uparrow t_e
\]

\[
(30) \lambda f. \exists e. f(e) : (\%v \rightarrow \%v_t) \rightarrow c^t_{f_t}, \text{ where } \%v = (\uparrow local)
\]

Note that (30) relies on the variable \( \%v \) which can resolve to \( \uparrow \) or \( (\uparrow local) \). The latter is used in the analysis of gapping: in the running example (1) with the f-structure in (23), (30) instantiates to (31) and combines with the agent Marge and the theme Lisa in a way analogous to the gapped clause, resulting in the MC (32) corresponding to the first conjunct.

\[
(31) \lambda f. \exists e. f(e) : (c^t_{f_t} \rightarrow c^t_{f_t}) \rightarrow c^t_{f_t}
\]

\[
(32) \lambda f. \exists e. theme(e,l) \land agent(e,m) \land f(e) : (c^t_{f_t} \rightarrow c^t_{f_t}) \rightarrow c^t_{f_t}
\]

The resulting MCs (32) and (27) for the two conjuncts are arguments to the instantiation of the standard meaning of and given in (33); note that the arguments correspond to conjuncts \( c^t \) and \( c^t \), and the result corresponds to the coordination \( e \).

\[
(33) \lambda V_1. \lambda V_2. \lambda f. V_1(f) \land V_2(f) : (c^t_{f_t} \rightarrow c^t_{f_t}) \rightarrow c^t_{f_t} \rightarrow c^t_{f_t} \rightarrow c^t_{f_t} \rightarrow c^t_{f_t} \rightarrow c^t_{f_t} \rightarrow c^t_{f_t}
\]

When (33) is applied to the MCs of conjuncts in (32) and (27), the result is (34) below (cf. (17) above).

\[
(34) \lambda f. \exists e. theme(e,l) \land agent(e,m) \land f(e) \land \exists e. theme(e,b) \land agent(e,h) \land f(e) : (c^t_{f_t} \rightarrow c^t_{f_t}) \rightarrow e_t
\]

The MC in (34) is input to (29), where \( \uparrow \) is instantiated to \( c \), resulting in (35) (cf. (18) above):

\[
(35) \lambda f. \exists e. theme(e,l) \land agent(e,m) \land see(e) \land f(e) \land \exists e. theme(e,b) \land agent(e,h) \land see(e) \land f(e) : (c^t_{f_t} \rightarrow c^t_{f_t}) \rightarrow c^t_{f_t}
\]

Finally, applying (35) to the closure in (28) results in the following final representation (cf. (19) and (2) above):

\[
(36) \exists e. theme(e,l) \land agent(e,m) \land see(e) \land true(e) \land \exists e. theme(e,b) \land agent(e,h) \land see(e) : c^t_{f_t}
\]

Note that the proposed analysis also covers examples where some dependents of the first conjunct are shared with the gapped conjunct, as in (37) where Marge can be the shared subject of both clauses (or Maggie can be the shared object).

\[
(37) \text{Marge sent Maggie to Lisa and Bart to Homer.}
\]

\(^1\)We assume the first-order approach to Glue (Kokkonidis 2008), with the usual basic semantic types \( e \) and \( t \), as well as \( n \) – the type of events.
Implementation The analysis presented above has been computationally verified as an XLE+Glue (Dalrymple et al. 2020) implementation. In the case of the running example (1), multiple proofs result in four different meaning representations, all equivalent to (2) – they only differ in the order of conjuncts corresponding to thematic roles, depending on the order in which meaning representations of dependents combine with the meaning representation (14) ($\lambda f. 3c. f(e)$).

Alternative Solution The above solution crucially relies on Champollion’s (2015) approach to event semantics, leading to reasonable semantics of the gapped clause even without the lexical verb (see (15) or (27)). This would not be possible if the meaning of the verb were the more standard $\lambda x. \lambda y. \lambda e. \text{see}(e) \land \text{agent}(e, x) \land \text{theme}(e, y)$ – or the event-less $\lambda x. \lambda y. \text{see}(x, y)$ – and if the representations of dependents were constants, without reference to events and thematic roles.

An attempt at a more general alternative solution is inspired by an idea implemented in XLE+Glue, namely, that meaning constructors are encoded in f-structures – as members of the glue attribute:

$\lambda p. \lambda q. \text{see} (p, q)$

Glue attribute: $\lambda x. \lambda y. \text{see}(x, y)$

If glue behaved like $\text{pred}$ (including being distributive), the syntactic analysis of gapping would produce the f-structures (40)–(41), fully analogous to (20)–(21) above, and the verb’s glue in (40) would distribute to conjuncts, as shown in (42).

If $\text{glue}$ behaved like $\text{pred}$ (including being distributive), the syntactic analysis of gapping would produce the f-structures (40)–(41), fully analogous to (20)–(21) above, and the verb’s glue in (40) would distribute to conjuncts, as shown in (42).

This solution would lead to the correct representation of the semantics of gapping. Unfortunately, there are two problems.

The potential conceptual problem is that syntax impinges on the resource sensitivity of Glue: while the MC which is the value of the verb’s glue attribute is introduced just once, in the lexical entry (38), it gets multiplied via the syntactic mechanism of f-structure distributivity. We believe that such behaviour is justified and desired, but it may be controversial.

The other problem is technical: in the current implementation of XLE, there does not seem to be a way of making glue behave like $\text{pred}$, i.e., in such a way that the $\uparrow$ metavariable used in assignments of glue values is instantiated independently in each conjunct (as in $\text{pred}$). As verified implementationally, even though glue can be declared as a distributive feature, $\uparrow$ used in glue points to the whole coordinate structure e (instead of resolving to $e^1$ in the first conjunct and $e^2$ in the second), while $\text{subj}$ gets resolved to the subsumption of the subj values of the two conjuncts (and similarly for $\text{obj}$).

References


2Due to space constraints, we gloss over the specific encoding of XLE+Glue meaning constructors here; see Dalrymple et al. 2020.

3Since glue is distributive, this analysis uses a slightly modified representation of conjunctions (see (42)). For space reasons, glue values within subj and obj are not represented in (40)–(42).