Embodied Construction Grammar in Simulation-Based Language Understanding

Benjamin K. Bergen* Nancy Chang†

Abstract

We present Embodied Construction Grammar, a formalism for linguistic analysis designed specifically for integration into a simulation-based model of language understanding. As in other construction grammars, linguistic constructions serve to map between phonological forms and conceptual representations. In the model we describe, however, conceptual representations are also constrained to be grounded in the body's perceptual and motor systems, and more precisely to parameterize mental simulations using those systems. Understanding an utterance thus involves at least two distinct processes: *analysis* to determine

*University of Hawaii at Manoa, Dept. of Linguistics, Moore 569, 1890 East-West Rd., Honolulu, HI 96822; `bergen@hawaii.edu`

†University of California at Berkeley and International Computer Science Institute, 1947 Center Street, Suite 600, Berkeley, CA 94704; `nchang@icsi.berkeley.edu`

which constructions the utterance instantiates, and *simulation* according to the parameters specified by those constructions. In this chapter, we outline a construction formalism that is both representationally adequate for these purposes and specified precisely enough for use in a computational architecture.

1 Overview

This chapter introduces a construction grammar formalism that is designed specifically for integration into an embodied model of language understanding. We take as starting point for Embodied Construction Grammar many of the insights of mainstream Construction Grammar (Goldberg 1995; Fillmore 1988; Kay and Fillmore 1999; Lakoff 1987) and Cognitive Grammar (Langacker 1991). Foremost among these is the observation that linguistic knowledge at all levels, from morphemes to multi-word idioms, can be characterized as **constructions**, or pairings of form and meaning. Along with other construction grammarians, we assume that language users exploit constructions at these various levels to discern from a particular utterance a corresponding collection of interrelated conceptual structures.

We diverge from other construction grammar research in our concern with precisely how constructional knowledge facilitates conceptually deep language understanding.[1] Understanding an utterance in this broader sense involves not only determining the speaker's intended meaning but also inferring enough information to react appropriately, whether with language (e.g., by answering a question) or some other kind of action (e.g., by complying with an order or request). These processes involve subtle interactions with variable general knowledge and the current situational and discourse context; static associations between phonological and conceptual knowledge will not suffice. Our model addresses the need for a dynamic inferential semantics by viewing the conceptual understanding of an utterance as the internal activation of **embodied schemas** – cognitive structures generalized over recurrent perceptual and motor experiences – along with the mental **simulation** of these representations in context to produce a rich set of inferences.

An overview of the structures and processes in our model of language understanding is shown in Figure 1. The main source of linguistic knowledge is a large repository of constructions that express generalizations linking the domains of **form** (typically, phonological schemas) and **meaning** (conceptual schemas). We also distinguish two interacting processes (shown as wide arrows) that draw on these schematic structures to interpret an utterance appearing in a particular communicative context:

- The **analysis** process determines which constructions the utterance in-

[1] Although we focus here on processes involved in language comprehension, we assume that many of the mechanisms we discuss will also be necessary for meaningful language production.

stantiates. The main product of analysis is the **semantic specification** (or **semspec**), which specifies the conceptual schemas evoked by the constructions involved and how they are related.

- The **simulation** process takes the semspec as input and exploits representations underlying action and perception to simulate (or enact) the specified events, actions, objects, relations, and states. The inferences resulting from simulation shape subsequent processing and provide the basis for the language user's response.

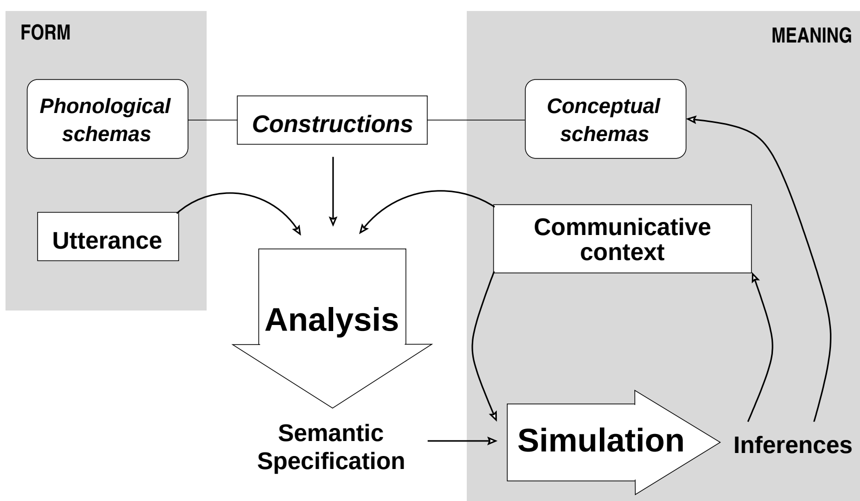

Figure 1: Overview of the simulation-based language understanding model, consisting of two primary processes: analysis and simulation. Constructions play a central role in this framework as the bridge between phonological and conceptual knowledge.

The embedding of construction grammar in a simulation-based language understanding framework has significant representational consequences. Constructions in ECG need specify only enough information to launch a simulation using more general sensorimotor and cognitive structures. This division of labor reflects a fundamental distinction between conventionalized, schematic meanings that are directly associated with linguistic constructions, and indirect, open-ended inferences that result from detailed simulation. In effect, constructions provide a limited means by which the discrete tools of symbolic language can approximate the multidimensional, continuous world of action and perception.

An adequate construction grammar formalism for our model must therefore provide a coherent interface between the disparate structures and processes needed in analysis and simulation; it must also be defined precisely enough to support a computational implementation. The remainder of this section provides an introductory tour of the ECG formalism – in particular, our representations of embodied schemas (Section 1.1) and constructions (Section 1.2) – using a simplified possible analysis of the phrase *into Rome*, as in *We drove into Rome on Tuesday*. We illustrate the formalism in greater detail with an extended analysis in Section 2, and address issues related to the overarching simulation-based framework in Section 3.

1.1 Embodied schemas

What does *into* mean, and how can we represent it? We take the central meaning of *into* to involve a dynamic spatial relation in which one entity moves from the exterior to the interior of another (as informally depicted in Figure 1.1). In the cognitive linguistics literature, such perceptually grounded concepts have been defined in terms of **image schemas** – schematic idealizations that capture recurrent patterns of sensorimotor experience (Johnson 1987; Lakoff and Johnson 1980). The relation captured by *into* can be seen as combining several image schemas, including the following:

- The Trajector-Landmark schema (Langacker 1987) captures an asymmetric spatial relationship involving a trajector, whose orientation, location, or motion is defined relative to a landmark.

- The Source-Path-Goal (or simply SPG) schema (Johnson 1987) structures our understanding of directed motion, in which a trajector moves (via some means) along a path from a source to a goal.

- The Container schema (Johnson 1987) structures our knowledge of enclosed (or partially enclosed) regions. It consists of a boundary separating the interior of the container from its exterior, and can also include a portal through which entities may pass.

Each image schema specifies structured relationships among a set of participants, often called **roles** (schema names and roles are shown in sans serif typeface above); roles can be instantiated by particular values (or **fillers**). Bottles, houses, and cities, for example, differ in many salient respects, but at a structural level they can all be interpreted as instances of the Container schema;

the other schemas likewise provide a level of structural abstraction over different situations. Roles within and across schemas may share their fillers, resulting in more complex composite structures like that associated with *into*. In our example phrase *into Rome*, the city of Rome serves as the landmark with respect to which a general locative event takes place; the destination of the motion; and the container within which the moving entity is ultimately located.

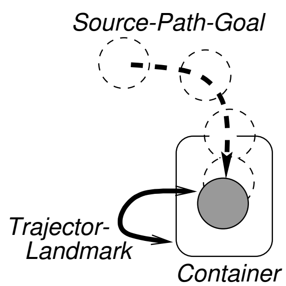

Figure 2: An iconic representation of some of the schemas involved in the meaning of *into*, including Container, Trajector-Landmark, and Source-Path-Goal.

Image schemas are part of a long tradition in linguistic analysis of schematic structures associated, at least implicitly, with richer underlying structures; these include Fillmore's (1982) semantic *frames* (script-like structures relating sets of interdefined participants and props); Talmy's (1988) *force-dynamic* schemas (capturing interactions involving the application or exertion of force); and Langacker's (1987) *semantic schemas* (the basic unit for meaning representation in Cognitive Grammar). It appears to be this schematic level, and not the more detailed sensorimotor level, that is encoded crosslinguistically in grammatical systems (Talmy 2000). In ECG, we refer to such schematic structures as **embodied schemas** (or **schemas**). The simplest embodied schemas can, like their predecessors, be depicted as a list of roles, as shown in Figure 1.1. These roles allow external structures (including other schemas as well as constructions) to refer to the schema's key variable features, providing a convenient degree of abstraction for stating diverse linguistic generalizations. More importantly for our purposes, schema roles are also intended to serve as **parameters** to more detailed underyling structures that can drive active simulations; Section 3.2 describes how a broad range of embodied meanings can be simulated using a dynamic representation called **executing schemas** (Bailey 1997; Narayanan 1997).[2]

More complex embodied schemas like Into involve the interaction of multiple schemas and their roles. Figure 1.1 draws on several additional representational devices to formalize our earlier prose description:

- The **subcase of** x tag asserts that the schema being defined is a specific case of a more general schema x; all of x's roles are accessible and its constraints apply. In the example, Into is marked as a subcase of the asymmetric relation between two entities captured by the Trajector-Landmark schema.

[2]Though we focus here on meaning, schematic representations in the form domain can also be viewed as schemas and represented using the same formalism, as we will show in the next section.

```
schema Trajector-Landmark
  roles
    trajector
    landmark
```

```
schema Container
  roles
    interior
    exterior
    portal
    boundary
```

```
schema SPG
  roles
    trajector
    source
    path
    goal
    means
```

Figure 3: ECG formalism for schemas involved in the meaning of *into*. Keywords of the notation are shown in **bold**. The initial header line names the embodied **schema** being defined, followed by an indented **roles** block listing the schema role names.

```
schema Into
  subcase of Trajector-Landmark
  evokes
    SPG as s
  roles
    trajector : Entity
    landmark : Container
  constraints
    trajector ⟷ s.trajector
    s.source ⟷ landmark.exterior
    s.goal ⟷ landmark.interior
```

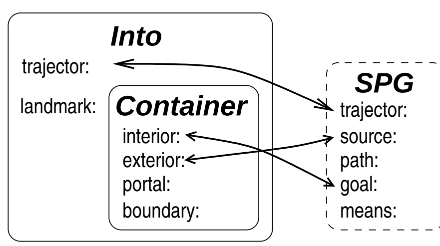

Figure 4: The Into schema, defined using the ECG formalism (left) and informally depicted as a set of linked schemas (right). Into is defined as a **subcase of** Trajector-Landmark that **evokes** an instance of the SPG schema (shown with a dashed boundary at right). Type constraints on roles require their fillers to be instances of the specified schemas, and identification bindings (⟷) indicate which roles have common fillers.

- The **evokes** block allows the schema to be defined against the background of other schemas; each line x **as** y gives the evoked schema x a local name (or **alias**) y for internal reference.[3] Here, an instance of the SPG schema is evoked and labeled as s.

- **Type** constraints (indicated with a colon, as $x : y$) restrict role x to be filled by an instance of schema y. The fillers of the Into schema's trajector and landmark roles are required to be instances of the Entity (not shown) and Container schemas, respectively.[4,5]

- Slot-chain notation is used to refer to a role y of a structure x as $x.y$; thus landmark.exterior refers to the exterior role of the Into schema's landmark role (itself a Container instance).

- **Identification** constraints (indicated with a double-headed arrow, as $x \longleftrightarrow y$) cause fillers to be shared between x and y. The **constraints** block **identifies** (or **binds**) the schema's inherited trajector role with the evoked SPG instance's trajector. The other identifications assert that the trajector's path takes it from the interior to the exterior of the container.

---

[3]The evokes relation has some antecedents (though not previously formalized) in the literature: In combination with the **self** notation to be described, it can be used to raise some structure to prominence against a larger background set of structures, effectively formalizing the notion of *profiling* used in frame semantics (Fillmore 1982) and Cognitive Grammar (Langacker 1991).

[4]Though no type constraints are shown in the other schemas, more complete definitions could require the relevant roles to be categorized as, for example, entities or locations.

[5]Determining whether a given entity can satisfy a type constraint may require active *construal* that depends on world knowledge and the current situational context, discussed further in Section 3.3.2.

(Note that the same evoked schemas with a different set of bindings would be needed to express the meaning of *out of*.)

Other notational devices not illustrated by this example include:

- **Filler** constraints (expressed using a single-headed arrow, as $x \longleftarrow y$) indicate that the role x is filled by the element y (a constant value).

- The keyword **self** refers to the structure being defined. This self-reference capability allows constraints to be asserted at the level of the entire structure.

Overall, the ECG schema formalism provides precise but flexible means of expressing schematic meanings, ranging from individual schemas to structured scenarios in which multiple schemas interact. The notational devices also allow us to assert that various relations hold among schemas (subcase, evokes) and their roles (identification, filler). Some of these bear a resemblance to notions familiar from object-oriented programming languages and constraint-based grammars (Shieber 1986; Pollard and Sag 1994); these include features, inheritance, typing, and unification/coindexation. But, as suggested by some of our terminological choices,[6] the formal tools used for representing schemas must be viewed in light of their main function in the present

[6]The subcase relation, for example, does not presume strict monotonic inheritance, and is thus more appropriate for capturing radial category structure (Lakoff 1987). Similarly, the **evokes** notation encompasses a more general semantic relation than either inheritance or containment; this underspecification allows needed flexibility for building semantic specifications.

context: providing means for external structures to set simulation parameters. These external structures include not just schemas but also, more importantly, constructions represented using similar mechanisms, as we describe in the next section.

1.2 A first look at constructions

Constructional approaches to grammar take the basic unit of linguistic knowledge to consist of form-meaning pairings, called **constructions**. This characterization crosscuts many traditional linguistic divisions, applying equally well to constructions of varying sizes (from morphological inflections to intonational contours) and levels of concreteness (from lexical items and idiomatic expressions to clausal units and argument structure patterns). In this section, we analyze our example *into Rome* as involving several such form-meaning mappings – including lexical constructions for *into* and *Rome* and a phrasal construction licensing their combination – and show how to represent them in the ECG construction formalism.

We begin with the simpler lexical constructions. The construction corresponding to *into* presumably links the Into schema described in Section 1.1 with some appropriate form representation. Although potential forms are not as open-ended as potential meanings, they nevertheless include such diverse elements as acoustic schemas, articulatory gestures, orthographic form(s), and stress or tone patterns. To ease exposition, we will rely here on a reduced notion of form including only phonological information, represented (as noted earlier) using the ECG schema formalism previously applied only to the meaning domain. Figure 1.2 shows the two form schemas used to define constructions in this chapter: a highly abstract Schematic-Form schema of which all other form schemas are subcases; and a Word schema with one role phon intended to contain specific phonological strings. (We assume that all words in spoken languages have this role.)

```
schema Schematic-Form
```

```
schema Word
 subcase of Schematic-Form
 roles
  phon
```

Figure 5: The Schematic-Form schema is the most general form schema; its (simplified) subcase Word schema has a phon role for specifying phonological strings.

```
construction SPATIAL-RELATION
 form : Schematic-Form
 meaning : Trajector-Landmark
```

```
construction INTO-CXN
 subcase of SPATIAL-RELATION
 form : Word
  phon ⟵ /Intu^W/
 meaning : Into
```

Figure 6: The SPATIAL-RELATION pairs a Schematic-Form as its form pole with a Trajector-Landmark as its meaning pole; its subcase INTO-CXN further restricts these types. In particular, its form pole is constrained to be a Word whose phon role is filled with the specified phonological string.

Figure 1.2 shows how the relevant form-meaning associations for *into* are expressed in the ECG construction formalism. We define two constructions: a general SPATIAL-RELATION construction, and a more specific INTO-CXN construction for our example. The notation is similar in many respects to that in the schema formalism, with initial header lines naming the **construction**s being defined (shown in SMALL CAPS, both in the figure and in text), and a **subcase** tag in INTO-CXN relating the two constructions. In fact, the construction formalism includes all the representational devices introduced for schemas. But to fulfill their basic function, constructions also include two indented blocks, labeled **form** and **meaning**, which stand for their two linked domains, or **poles**. These poles list the elements and constraints (if any) within each domain, but they should also be considered special components of the construction that can be referred to and constrained, roughly analogous to schema roles. As shown in the figure, SPATIAL-RELATION's type constraints restrict its form pole to be an instance of Schematic-Form and its meaning to be an instance of Trajector-Landmark (from Figure 1.1). This constructional category is thus general enough to include a variety of spatial relations expressions that denote Trajector-Landmark relationships, including not just single words (like *into* and *over*) but also multiword expressions (like *out of* and *to the left of*). These type constraints apply to all subcases of the construction; INTO-CXN imposes even stricter requirements, linking an instance of Word (a subcase of Schematic-Form) with an instance of Into (a subcase of

Trajector-Landmark). The form block also includes a filler constraint on its phon role, specifying /Intu^{W}/ as the particular phonological string associated with the construction,

The other lexical construction in our example is similarly represented using a pair of related constructions, one a subcase of the other. The constructions shown in Figure 1.2 are intended to capture the basic intuition that the ROME construction is a specific **referring expression** (REF-EXPR) that picks out a known place in the world. Referring expressions will be discussed in more detail in Section 2.1. For now we need only stipulate that REF-EXPR's meaning pole, an instance of the Referent schema, includes a resolved-referent role whose filler is the entity picked out by the expression. In our example, ROME-CXN is defined as a subcase of the general construction that, besides specifying an appropriate phonological string, binds this role to the (conceptual schema) Rome, a known entity in the understander's ontology.[7]

The final construction used in our example phrase illustrates how constructions may exhibit constituent structure. The phrase *into Rome* exemplifies a pattern in which a spatial relation with a particular landmark is associated with two expressions: a SPATIAL-RELATION and a REF-EXPR, in that

[7]This direct binding of the resolved-referent effectively captures the commonsense generalization that proper nouns (by default) pick out specific known entities. Other kinds of referring expressions typically require a dynamic *reference resolution* process, parameterized by the Referent schema, to determine the relevant entity; see Section 2.1.

construction Ref-Expr
form : Schematic-Form
meaning : Referent

construction Rome-Cxn
subcase of Ref-Expr
form : Word
phon $\longleftarrow$ /ro^{w}m/
meaning
resolved-referent $\longleftrightarrow$ Rome

Figure 7: The Ref-Expr construction underlying all referring expressions pairs a schematic form with a Referent schema. Its subcase Rome-Cxn identifies the resolved-referent role of its meaning pole with the known place specified by the Rome schema, and pairs this with the appropriate phonological string.

order. Despite the relatively abstract nature of these elements, this pattern can be expressed using the same representational mechanisms as the more concrete constructions we have already seen, with one addition. As shown in Figure 1.2, we introduce a **constructional** block listing two constituent elements, sr and lm, which are typed as instances of the Spatial-Relation and Ref-Expr constructions, respectively.[8] (Instances of constructions are also called **constructs**.) These constituents, and their form and meaning poles, may be referenced and constrained just like other accessible elements. In the formalism, a subscripted f (for form) or m (for meaning) on a construct's name refers to the appropriate pole. Moreover, since the **self** notation refers to the construction being defined, **self**$_f$ and **self**$_m$ can be used to refer to the form and meaning poles, respectively, of the construction in which they appear. We can thus assert relations that must hold among constituents, or between a

[8]Note that this view of constituency extends the traditional, purely syntactic notion to include form-meaning pairings.

construction and its constituents.

construction SPATIAL-PHRASE
constructional
sr : SPATIAL-RELATION
lm : REF-EXPR
form : Schematic-Form
sr_f `before` lm_f
meaning : Trajector-Landmark
sr_m.landmark $\longleftrightarrow$ lm_m
$\textbf{self}_m$ $\longleftrightarrow$ sr_m

Figure 8: The SPATIAL-PHRASE construction has two constituents specified in the **constructional** block. The form and meaning poles of these constituents are subject to both a word order constraint (in the form block) and an identification constraint (in the meaning block). The meaning of the overall construction is also bound to the meaning of its sr constituent.

The form and meaning blocks of the SPATIAL-PHRASE construction impose several such relational constraints. The single form constraint expresses the word order requirement mentioned earlier: the form pole of rel must precede that of lm, though not necessarily immediately (since modifiers, for example, might intervene). We notate this constraint with the interval relation `before`, one of many possible binary relations between intervals set out in Allen's (1984) Interval Algebra. (Immediate precedence is expressed using the `meets` relation.) The meaning block similarly relates the two constituents: the landmark role of the sr constituent's meaning pole (an instance of the Trajector-Landmark schema) is identified with the lm constituent's meaning pole. The other constraint uses the $\textbf{self}_m$ notation to identify the overall construction's meaning pole (also an instance of the Trajector-Landmark schema) with that of its sr constituent. In other words, the meaning of the entire construction is essentially the same spatial relation specified by its sr constituent, but with the particular landmark specified by its lm constituent.

For the SPATIAL-RELATION construction to license our example phrase *into Rome*, instances of the lexical INTO and ROME constructions must satisfy all the relevant type, form, and meaning constraints on the sr and lm constituents. Note that the particular constructs involved may impose constraints not directly specified by SPATIAL-PHRASE. In this case, the Into schema constrains its landmark – identified by the first meaning constraint with the Rome schema – to be an instance of a Container. Assuming, as suggested earlier (though not formally depicted), that cities and other geographical regions may serve at least abstractly as instances of the Container schema, the binding succeeds, resulting in a set of interrelated semantic structures resembling that depicted in Figure 1.1 with the Rome schema serving as the landmark container.

Our brief introduction to Embodied Construction Grammar has highlighted the formal representations of both schemas and constructions. Embodied schemas capture generalizations over experience in the domains of form or meaning; we represent them as role description structures that can parameterize simulations. Schemas may be subcases of more general schemas, or evoke and constrain instances of other schemas; their roles may be required to have fillers of specific types, or they may be identified with other roles or filled by particular values. Constructions are in some sense a special bipolar schematic structure that captures generalizations over form-meaning pairs; they thus employ a similar range of representational mechanisms. Constructions may also have internal constructional constituents upon which they may assert relational constraints. In the next section, we illustrate the interaction of these conceptual and linguistic representations in greater detail, deferring until the third section larger issues involved in the processes of constructional analysis and simulative inference.

2 A detailed analysis

This section shows our construction formalism at work in a more complex example. We present a collection of constructions that together license an analysis of the utterance in (1):

(1) Mary tossed me a drink.

Our analysis follows that of Goldberg (1995) in presuming that the ditransitive argument structure (in this example, the active ditransitive argument structure) imposes an interpretation in which one entity takes some action that causes another entity to receive something. Thus, although the verb *toss* appears with a variety of argument structures, its appearance in the example sentence is allowed only if its meaning pole can be understood as contributing to a transfer event of this kind.

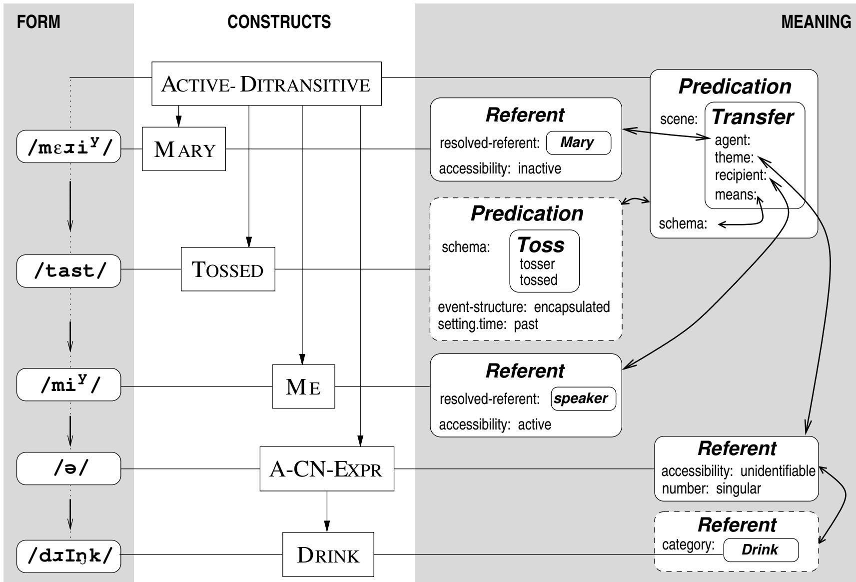

Figure 9: A depiction of a constructional analysis of *Mary tossed me a drink*. Constructs involved are shown in the center, linking elements and constraints in the domains of form and meaning; schemas are shown as rounded rectangles. (Some details not shown; see text.)

Figure 2 is a simplified depiction of the analysis we develop in this section. The form and meaning domains linked by constructional knowledge are shown as gray rectangles on either side of the figure. Form elements — including phonological schemas (shown simply as phonological strings in rounded rectangles) and word order relations (shown as arrows on a schematic time line) — appear in the form domain. Meaning elements — including schemas (shown as rounded rectangles) and bindings among their roles (shown as double-headed arrows) — appear in the meaning domain. The six rectangles lying between these domains correspond to the six constructs involved in the analysis. Each construct is labeled according to the construction it instantiates and is linked to other elements in the analysis in various ways. Horizontal lines link each construct with its form and meaning poles, while vertical arrows between the boxes express constructional constituency. For example, the box for the MARY construct has a (form) link to the phonological form /mɛɹi^{y}/ (residing in the form domain) and a (meaning) link to Referent schema (residing in the meaning domain), which resolves to a Mary schema; in this analysis it is also a constructional constituent of the ACTIVE-DITRANSITIVE construct.

The constructions and schemas shown in the diagram (as well as several others not shown) are defined in this section using the ECG formalism. As will become clear, many of the details of the analysis — such as the specific constructions and schemas involved, as well as the subcase relations among them — are subject to considerable debate. Our current purpose, however, is not to offer the most general or elegant definition of any particular construction, but rather to demonstrate how the ECG formalism can express the choices we have made. The analysis also highlights the interaction between lexical and clausal semantics, suppressing details of how the formalism could represent sub-lexical constructions and more significant interactions with the discourse context; alternative analyses are mentioned where relevant.

We broadly divide the constructions to be defined in this section into those that allow the speaker to *refer* and those that allow the speaker to *predicate*. This division reflects the differing communicative functions of reference (typically associated with entities) and predication (typically associated with events). Following Croft (1990, 1991, 2001), we take reference and predication to be primary propositional acts that motivate many traditional grammatical categories and relations; they also have natural interpretations in our framework as the main schemas structuring the simulation (Section 3.1). We organize our analysis accordingly: the referring expressions in our example — *Mary*, *me*, and *a drink* — are defined in Section 2.1, followed by expressions involved in predication — both the main verb *tossed* and the ditransitive argument structure construction — in Section 2.2.

2.1 Referring expressions

The act of making **reference** (to some **referent** or set of referents) is a central function of linguistic communication. Speakers use language to evoke or direct attention to specific entities and events. A wide range of constructions is used for this function, including pronouns (*he*, *it*), proper names (*Harry*, *Paris*), and complex phrases with articles, modifiers, and complements (e.g., *a red ball*, *Harry's favorite picture of Paris*). But while the forms used in these constructions are highly variable, they all rely on the notion of reference as a core part of their meaning. The REF-EXPR (referring expression) construction defined in Section 1.2 and repeated here, is thus relatively schematic, linking a Schematic-Form with a Referent (Figure 2.1).

The roles of the Referent schema correspond to information that a refer-

schema Referent
roles
category
restrictions
attributions
number
accessibility
resolved-referent

construction REF-EXPR
form : Schematic-Form
meaning : Referent

Figure 10: The Referent schema, the meaning pole of all referring expressions (REF-EXPR, repeated from Figure 1.2), contains information related to an active reference resolution process, including the number and accessibility of the intended referent.

ring expression may convey about a referent. These include its ontological category (e.g., human, ball, picture); restrictions and attributions that apply to various open-class characteristics of the referent (e.g., size or color); the number of the referent (e.g. singular or plural), and its default level of accessibility (Lambrecht 1994) in the current discourse context (active, accessible, inactive, unidentifiable, etc.).[9,10]; Specific subcases of REF-EXPR may place further constraints on these roles, which are used in a separate reference resolution procedure that finds the most likely referent in context (for example, a particular known individual or event); this actual referent, when determined, is the filler of the resolved-referent role. Some referring expressions, such as

[9]Though not shown, the context model includes speaker and hearer roles, discourse context (referents and predications in previous utterances), situational context (entities and events in the actual or simulated environment), and shared conceptual context (schema instances known to both speaker and hearer). We use a simplified version of Lambrecht's (1994) terminology for referential identifiablity and accessibility, though other discourse frameworks could be substituted.

[10]Other roles of this schema that may be relevant for particular languages include gender and animacy; they are not relevant to the current example and thus are not discussed here.

proper nouns (like *Rome*) and local deictic pronouns (like *I* and *me*) assert a direct binding on the resolved-referent role.

Our example includes three different referring expressions: *Mary*, *Me*, and *a drink*. We will analyze these as involving three constructions that are all subcases of the REF-EXPR construction — MARY, ME, and A-CN-EXPR — as well as COMMON-NOUN and its subcase DRINK-CXN. Some constraints in the constructions we show could be expressed instead in these more general constructions corresponding to proper nouns, pronouns, and determined phrases. To simplify the analysis, we have opted for more specific constructions that make fewer commitments with respect to subcase relations. Note, however, that the two approaches can be viewed as informationally equivalent with respect to the utterance under consideration.

We begin with the MARY and ME constructions (Figure 2.1). Both of these are specified as subcases of REF-EXPR, and have form and meaning poles that are structurally similar to the ROME construction from Section 1.2. Each form pole is an instance of the Word schema with the appropriate phonological string, and each meaning pole constrains the resolved-referent role and specifies the referent's level of accessibility. The differences in meaning pole constraints reflect the differing functions of proper nouns and pronouns: proper nouns like *Mary* refer to known ontological entities (here, the Mary schema is intended to correspond to an individual conventionally named "Mary") and thus can be used with no prior mention; they need only a minimal inactive level of accessibility. In contrast, pronouns like *me* and *you* identify referents for which the interlocutors have active representations in the current discourse; in this case, the Me construction makes deictic reference to the speaker role in the current context (notated here as current-space.speaker; see Section 4 for discussion of how this role relates to work in *mental spaces*).

```
construction Mary
 subcase of Ref-Expr
 form : Word
  phon ⟵ /mɛɹi^y/
 meaning
  resolved-referent ⟷ Mary
  accessibility ⟵ inactive
```

```
construction Me
 subcase of Ref-Expr
 constructional
  case ⟵ object
 form : Word
  phon ⟵ /mi^y/
 meaning
  resolved-referent ⟷ current-space.speaker
  accessibility ⟵ active
```

Figure 11: The Mary and Me constructions, both subcases of Ref-Expr, bind the Referent schema's resolved-referent role to the Mary schema and the current speaker, respectively, and set different default levels of accessibility. The Me construction also constrains its case constructional feature.

The Me construction also differs from the Mary construction in having a **constructional** block, whose single case role is assigned the value object. In the Spatial-Phrase construction, this block was used only to list constructional constituents. Here, however, we illustrate its more general function of specifying any elements or constraints applicable to the construction as a whole – that is, information residing in neither the form nor meaning domain alone. The case role (also termed a constructional **feature**) distinguishes the ME construction from the constructions for *I* (subject case) and *my* (possessive case) (as discussed further in Section 2.2.3). Note that in a more complete analysis of English, the case feature would be defined in a general PRONOUN construction; for other languages with wider use of case, this feature might be defined in the more abstract REF-EXPR construction.

The final referring expression in our example, the phrase *a drink*, has more internal structure than the other ones we have considered. In traditional analyses, each word in the phrase — the article *a* and the common noun *drink* — corresponds to a constituent of the overall expression. But we elect here to treat the article as semantically and formally inseparable from the referring expression — that is, as tied to the context in which it precedes some category-denoting expression (traditionally called a *common noun*) and refers to an individual of the specified category. We formalize this analysis in Figure 2.1 with three constructions: a COMMON-NOUN construction, its subcase DRINK-CXN construction, and the A-CN-EXPR construction (or *a*-common noun expression, to contrast with a similar *the*-common noun expression, not shown). As usual, other alternatives are possible, but this analysis captures the constraints present in our example while demonstrating the flexibility of the ECG formalism as used for referring expressions.

construction COMMON-NOUN
form : Schematic-Form
meaning
evokes Referent **as** ref
self$_m$ $\longleftrightarrow$ ref.category

construction DRINK-CXN
subcase of COMMON-NOUN
form : Word
phon $\longleftarrow$ /dɹɪŋk/
meaning : Drink

construction A-CN-EXPR
subcase of REF-EXPR
constructional
com-noun **:** COMMON-NOUN
form
a-form $\longleftarrow$ /ə/
a-form `before` com-noun$_f$
meaning
self$_m$ $\longleftrightarrow$ com-noun$_m$.ref
accessibility $\longleftarrow$ unidentifiable
number $\longleftarrow$ singular

Figure 12: Constructions underlying *a drink*: COMMON-NOUN and its subcase DRINK-CXN supply a referent's category by bindings its meaning pole (for DRINK-CXN, the Drink schema) to its evoked Referent schema's category slot. The A-CN-EXPR construction has one constructional constituent, typed as a COMMON-NOUN, which it constrains to follow the form element it introduces (/a/). Its meaning pole, Referent schema, is identified with the evoked Referent of its constituent and further constrained.

The overall intuition captured by the analysis is that common nouns provide categorical information about a referent, and expressions involving common nouns place further restrictions on the reference resolution process. The COMMON-NOUN construction thus evokes a Referent, whose category role is identified with the entire construction's meaning pole. Its subcase DRINK-CXN specializes both its form pole (with a particular phonological string) and its meaning pole (typed as a Drink). In sum, these two constructions assert that the common noun *drink* has as its meaning pole the Drink schema, which is the category of the Referent schema it evokes by virtue of being a common noun (as depicted in Figure 2). The A-CN-EXPR construction combines the Referent evoked by its com-noun constituent — which, as an instance of COMMON-NOUN, supplies categorical information — with its own Referent meaning pole. The form block introduces an internal form element a-form and constrains it to appear before the com-noun constituent. The meaning block imposes additional constraints on the overall Referent, corresponding to the traditional functions of the indefinite singular determiner *a*: the accessibility is set as unidentifiable, which among other effects may introduce a new referent into the discourse context; and its number is set as singular.

Our treatment of reference, though preliminary, nevertheless suffices for the simple lexical and phrasal referring expressions in our example. Further research is necessary to account for the full range of referential phenomena, including modifiers, complements, and relative clauses. But we believe that even these complex referring expressions can be approached using the basic strategy of evoking and constraining a Referent schema that serves as input for reference resolution.

2.2 Predicating expressions

The act of **predication** can be considered the relational counterpart to reference. Speakers make attributions and assert relations as holding of particular entities; and they locate, or ground, these relations (in time and space) with respect to the current speech context. Central cases of constructions used to predicate include Goldberg's (1995) basic argument structure constructions and other clausal or multiclausal constructions. But many other kinds of construction — including the traditional notion of a *verb* as designating a relation between entities, as well as both morphological constructions and larger verb complexes that express tense, aspect, and modality — provide information relevant to making predications.

```
schema Predication
  roles
    scene
    schema
    event-structure
    setting
```

```
construction Pred-Expr
  form : Schematic-Form
  meaning : Predication
```

Figure 13: The Predication schema and Pred-Expr construction are the analogs in the domain of predication to the Referent schema and Ref-Expr construction. The Predication schema captures major aspects of predicating, including the overall scene and the primary schema involved.

Figure 2.2 shows an ECG schema that organizes predicative content, the Predication schema. As usual, the roles given here are not intended to be exhaustive, but they suffice for describing a wide range of predications, including the one in our example, in precise enough terms to simulate. The schematic Pred-Expr (predicating expression) construction is analogous to the Ref-Expr construction in covering a wide range of expressions that predicate; it pairs a Schematic-Form instance with a Predication instance. (Other predicative constructions, like the verbal constructions to be considered later, may simply evoke a Predication instance in their meaning poles.)

The first two roles of Predication together specify the main conceptual content and participant structure being asserted, in terms of both the overall scene (typically set by clausal constructions) and a main schema involved (typically set by verbal constructions). In general, the underlying semantics associated with these two roles must be understood as part of one coherent event. The scene role can be filled by a relatively limited set of schemas that describe basic patterns of interaction among a set of participants. These correspond roughly to what Goldberg (1995) refers to as “humanly relevant scenes”, as well as to the basic scenes associated with children’s cross-linguistically earliest grammatical markings (Slobin 1985); examples include Force-Application (one participant exerting force on another), Self-Motion (a self-propelled motion by a single participant), Caused-Motion (one participant causing the motion of another), or, as in our example sentence, Transfer (a participant transfers an entity to a second participant). These overall scenes generalize over the particular concrete actions involved — whether, for example, the participant in an instance of Self-Motion sustains the motion by walking, hopping, or pushing through a crowd; the concrete schemas are bound instead to the schema role. As we shall see, the relation between scene and schema is at the crux of the analysis process, since many factors influence their interaction. Their separation in the Predication schema provides some useful representational flexibility: individual constructions may specify as much or as little as needed about these roles and how they are related.

The remaining roles of the Predication schema supply additional information about how the event is to be understood. The event-structure role constrains the shape of the event asserted in the predication or the particular stage it profiles; cross-linguistically, markers of linguistic *aspect* typically affect this role. The event may also be located in a particular setting in time or space; tense markings, for example, generally affect a substructure time of the setting role.

We analyze our example sentence as involving two main constructions that interact to define the overall predication: the verbal TOSSED construction and the clausal ACTIVE-DITRANSITIVE construction. These constructions exemplify the pattern mentioned above: the verbal construction binds a particular action schema (the Toss schema) to the schema role, while the clausal construction binds a Transfer schema to the scene role.[11] In the analysis we will develop, these separately contributed schemas are directly related in the final predication: the tossing action is understood as the *means* by which a transfer is effected.[12] We examine first the schemas needed to represent the meanings involved in our example sentence (Section 2.2.1) and then use these to define

[11]Both constructions can be viewed as combining two other constructions: the finite verb TOSSED could result from a morphological construction combining the verbal stem *toss* with an *-ed* marker; and the information in the ACTIVE-DITRANSITIVE construction could be separately specified in a DITRANSITIVE argument structure construction and an ACTIVE clausal construction, which could also impose constraints on the predication's information structure (not included in the current analysis). These more compositional analyses are consistent with the approach adopted here and can be expressed in the ECG formalism.

[12]Other possible relations mentioned by Goldberg (1995) include subtype, result, precondition, and manner.

the relevant verbal (Section 2.2.2) and clausal (Section 2.2.3) constructions.

2.2.1 Representing scenes

In this section we consider some schemas needed to represent the meanings predicated by our example sentence, *Mary tossed me a drink*. We interpret the sentence as asserting that at some point before speech time, the referent of *Mary* applied a tossing action to the referent of *a drink*, which as a result is received by the referent of *me* (the speaker in the current context). Prototypically, the action of tossing is a low-energy hand action that causes an entity to move through the air; since it intrinsically causes motion, we will define it relative to the general Caused-Motion schema. Our example has the further implication that the referent of *a drink* is received by the speaker. That is, it depicts an overall scene of Transfer, in which one entity acts to cause another to receive a third entity, irrespective of the particular action involved.

We follow Goldberg (1995) in attributing this Transfer semantics to the ditransitive clausal pattern, or argument structure construction, where the subject encodes the causer of transfer, the first postverbal object encodes the recipient of transfer, and the second postverbal object the transferred entity. We base this analysis on evidence such as that in (2):

(2) a. Mary spun/broomed me a drink. *(transfer)*

b. ? Mary tossed the floor a drink. *(?transfer)*

c. Mary tossed a drink to the floor. *(caused-motion)*

Sentence (2a) shows that ditransitive syntax can impose an intended transfer reading even on verbs not prototyically associated with transfer, including transitive verbs like *spin* as well as novel denominal verbs like *broom*. This transfer sense is distinct from the semantics associated with caused-motion clausal syntax, as demonstrated by the differing acceptability of the sentences in (2b) and (2c). The referent of the first object in a ditransitive sentence must serve as a recipient — that is, it must be categorized or construed as something that can receive the transferred object. Thus (2b) has an acceptable reading only under a (metaphorical, anthropomorphized) construal of *the floor* as a possible receiver and possessor of objects. This requirement does not apply to the caused-motion argument structure in (2c), which implies only that the agent causes motion of the entity along some path, without any entailment of receiving.[13]

These intuitions can be made concrete using the representational tools of ECG to define the two relevant scenes, Caused-Motion and Transfer (Figure 2.2.1), each defined in terms of several other schemas (Figure 2.2.1). The two scenes are structurally parallel: each involves a forceful action on the part

[13]See Goldberg (1995) for further motivation of details of the analysis, such as the choice of the action of receiving rather than a state of possession as the result of the transfer action.

```
schema Caused-Motion
 evokes
  Force-Application as fa
  SPG as s
  Cause-Effect as ce
 roles
  agent ⟷ fa.energy-source
  theme ⟷ fa.energy-sink ⟷ s.trajector
  path ⟷ s
  means ⟷ fa.means
 constraints
  ce.cause ⟷ fa
  ce.effect ⟷ s
```

```
schema Transfer
 evokes
  Force-Application as fa
  Receive as rec
  Cause-Effect as ce
 roles
  agent ⟷ fa.energy-source
  theme ⟷ rec.received
  recipient ⟷ rec.receiver
  means ⟷ fa.means
 constraints
  ce.cause ⟷ fa
  ce.effect ⟷ rec
```

Figure 14: The structurally similar Caused-Motion (in which an agent acts on a theme via some means such that it moves along a path) and Transfer (in which an agent acts on a theme via some means such that it is received by a recipient) capture scenes relevant to the example.

of an agent entity, which causes some effect on a theme entity. The forceful action is captured by the Force-Application schema, which involves an energy-source that exerts force on an energy-sink via some means, possibly through an instrument; the type and amount of force may also be specified.[14] The causal structure is captured by the simple Cause-Effect schema, which lists only a cause and a resulting effect. Each of the schemas in Figure 2.2.1 evokes both the Force-Application and Cause-Effect schemas and asserts constraints that identify the agent in each scene with the energy-source of the forceful action, the overall means of the scene with the means of the forceful action, and the forceful action itself with the Cause-Effect's cause.

```
schema Force-Application
 roles
   energy-source
   instrument
   energy-sink
   force-type
   force-amount
   means
```

```
schema Cause-Effect
 roles
   cause
   effect
```

```
schema Receive
 roles
   receiver
   received
```

Figure 15: Embodied schemas contributing to the example sentence: Force-Application captures scenarios in which an energy-source exerts force on an energy-sink; Cause-Effect captures causal relations; and Receive schema has roles for a receiver and a received entity.

Where the two scenes differ is in their effects — that is, in the particular schemas bound to the effect role of their evoked Cause-Effect schemas. In

[14]This schema can be seen as one of many types of force-dynamic interaction described by Talmy (1988).

the Caused-Motion scene, the result of the forceful action is the motion of the theme entity along a path; this is captured by an evoked SPG schema (defined earlier), whose trajector is bound to the theme. (Note that the formalism allows multiple identifications to be expressed at once, in either the roles or constraints block.) In the Transfer scene, the effect is bound not to an SPG but rather to an evoked Receive schema, with the receiver and the received bound to the Transfer scene's recipient and theme roles, respectively.

Both scenes we have defined are abstract in that the particular action (or means) involved is not specified; indirectly, however, they both require some action that is construable as applying force, and that the agent role's filler must be capable of performing. The concrete actions are typically supplied by specific verbs. These indirect constraints thus play a key role in determining how verbs interact with clausal constructions evoking these scenes, as we will show for the particular verb *tossed* in the remainder of this section.

2.2.2 TOSSED as a VERB

We first consider how the action of tossing can be represented using embodied schemas before defining the construction for the verb *tossed*. As noted earlier, the Toss schema needed for our example is semantically compatible with either of the scenes we have described, but it is intrinsically associated with caused motion and thus defined here against the backdrop of the Caused-Motion schema (Figure 2.2.2). Specifically, Toss evokes both a

Caused-Motion schema and a Fly schema (not shown); it identifies itself with the means role of the evoked Caused-Motion, as expressed by the first line in the constraints block. The remaining constraints straightforwardly identify the Toss's two roles, a tosser and a tossed object, with appropriate roles in the evoked schemas; restrict the degree of force used in the causal action to low; and bind the means of the associated resulting motion to the evoked Fly action. In sum, the action of tossing is a (somewhat) forceful action on an entity that causes it to fly. (As usual, this schema should be viewed as summarizing the motor parameters for a more detailed representation of the tossing action schema, to be discussed in Section 3.2.1.)

```
schema Toss
 evokes
  Caused-Motion as cm
  Fly as f
 roles
  tosser ⟷ cm.agent
  tossed ⟷ cm.theme ⟷ f.flyer
 constraints
  cm.means ⟷ self
  cm.fa.force-amount ⟵ low
  cm.path.means ⟷ f
```

Figure 16: The Toss schema is identified with the means of its evoked Caused-Motion. It also constrains the associated Force-Application to be a low-force action that results in a flying motion.

We now turn to the verb *tossed*, which is linked to the Toss schema described in the last section, but also carries aspect and tense information that applies to the larger predication associated with the overall sentence. Loosely following Langacker (1991), we define the VERB construction as a word that evokes a Predication instance, such that its subcases (including the TOSSED construction) may assert further constraints (both constructions are shown in Figure 2.2.2). Specifically, the TOSSED construction associates the phonological form /tast/ with a meaning pole typed as an instance of the Toss schema. This entire meaning pole is bound to pred.schema, indicating that it serves as the main schema of its evoked Predication. The remaining constraints affect Predicaton roles related to aspect and tense. First, as discussed further in Section 3.2.1, the English simple past tense can be modeled using executing schemas that suppress, or **encapsulate**, details of their internal structure during simulation; the Predication's event-structure is thus set as encapsulated. Second, the constraint setting the pred.setting.time as past indicates that the time during which the relational predication holds, corresponding to Reichenbach's (1947) Event Time, must be prior to the (contextually specified) Speech Time.

2.2.3 The ACTIVE-DITRANSITIVE construction

The only remaining construction to define is the argument structure construction spanning the entire utterance, the ACTIVE-DITRANSITIVE construction. As suggested earlier, we analyze this construction (Figure 2.2.3), as well as other ditransitive constructions like PASSIVE-DITRANSITIVE and IMPERATIVE-DITRANSITIVE, as a subcase of the PRED-EXPR construction

construction VERB
form : Word
meaning
evokes Predication **as** pred

construction TOSSED
subcase of VERB
form
phon **:** /tast/
meaning : Toss
self$_m$ $\longleftrightarrow$ pred.schema
pred.event-structure $\longleftarrow$ encapsulated
pred.setting.time $\longleftarrow$ past

Figure 17: The VERB construction evokes a Predication schema. Its subcase TOSSED construction identifies its meaning pole, typed as a Toss schema, with the evoked Predication schema's main schema role and asserts aspect and tense constraints.

whose associated predication is based on a scene of Transfer. The close relation between this clausal construction and the Transfer scene is reflected by its four constituents, which are deliberately given aliases parallel to those of the Transfer schema's roles.

Constructional constraints enforce case restrictions on pronouns filling the agent, theme, and recipient constituents (discussed in Section 2.1), accounting for the judgments in (3):[15]

[15]Our use of a formal case attribute does not preclude the possibility that case patterns may be motivated by semantic regularities (Janda 1991). The current analysis is intended to demonstrate how constraints on such a constructional feature could be imposed; a more detailed analysis would involve defining constructions that capture the form and meaning regularities related to case marking.

construction ACTIVE-DITRANSITIVE
subcase of PRED-EXPR
constructional
agent : REF-EXPR
action : VERB
recipient : REF-EXPR
theme : REF-EXPR
recipient.case $\longleftarrow$ object
agent.case $\longleftarrow$ subject
theme.case $\longleftarrow$ object
form
agent_f `before` action_f
action_f `meets` recipient_f
recipient_f `meets` theme_f
meaning
evokes Transfer **as** tr
$\textbf{self}_m$.scene $\longleftrightarrow$ tr
tr.agent $\longleftrightarrow$ agent_m
tr.theme $\longleftrightarrow$ theme_m
tr.recipient $\longleftrightarrow$ recipient_m
tr.means $\longleftrightarrow$ action_m
$\textbf{self}_m$ $\longleftrightarrow$ action_m.pred

Figure 18: The ACTIVE-DITRANSITIVE construction has four constituents, including three referring expressions with specified case values. Besides imposing order constraints, the construction binds its meaning pole (a Predication), with its verbal constituent's evoked predication; its evoked Transfer schema with its scene role; and the meaning poles of its constituents with roles of the Transfer schema.

(3) a. * Mary tossed I/my a drink.

b. * Me/my tossed Mary a drink.

The three order constraints reflect intuitions suggested by the examples in (4):

(4) a. Mary tossed me a drink.

b. Mary happily tossed me a drink.

c. * Mary tossed happily me a drink.

d. * Mary tossed me happily a drink.

e. Mary tossed me a drink happily.

That is, the agent must precede the action (though not necessarily immediately), and no intervening material is allowed between the action and recipient constituents, nor between the recipient and theme constituents.

The meaning constraints are more complicated. The entire meaning pole is a Predication, as specified by the PRED-EXPR construction, but it also evokes an instance of the Transfer schema. This schema is bound to **self**$_m$.scene — that is, the scene role of the overall construction's meaning pole, which is itself an instance of Predication — and its roles are in turn bound to the meaning poles of the various constituents. A final complication is dealt with by the last meaning constraint, which identifies the entire meaning pole with the Predication evoked by the verbal action constituent. (This binding corresponds to the double-headed arrow linking the two Predication schemas in Figure 2.) This constraint allows the overall predication to incorporate any relevant constraints expressed by the verb.

We can now examine the interaction of verbal and clausal semantics in our example, in which the Active-Ditransitive construction's action constituent is filled by the verb *tossed*. The verbal and clausal constructions both assert constraints on the overall predication: TOSSED supplies aspect and tense information and the main schema involved (Toss), while Active-Ditransitive specifies the scene (Transfer) and binds its roles. Crucially, the Toss schema provided by the verb is required to serve as a means of transfer (since it is bound to the Transfer schema's means role). This binding succeeds, since both Toss and the Transfer schema's means role are bound to the means of a Force-Application schema (see Figure 2.2.1 and Figure 2.2.2). As a result, the forceful action involved in a transfer event is identified with the forceful action involved in a tossing action, which in turn causes the agent of transfer to be bound to the tosser. Similar propagation of bindings also leads the tossed object to be identified with the theme of the transfer event, although we have not shown the relevant internal structure of the Receive schema.[16]

As just shown, the formalism permits the expression (and enforcement) of bidirectional constraints between verbal and clausal semantics — in this case, for example, a restriction on ditransitive construction to verbs that entail some force-dynamic transfer (Langacker 1991). Failure to fulfill such restrictions can result in reduced acceptability and grammaticality of particular combinations of clausal constructions with particular verbs or referring expressions:

[16] A fuller definition of the Receive schema would evoke an SPG as (part of) the effect of the Transfer schema's evoked Force-Application. Since the forceful actions of the Toss and Transfer schemas are identified, their respective effects are as well, resulting in a binding between their tossed and theme roles.

(5) * Mary slept me a drink. *(Her sleeping gave the speaker a drink.)*

In an attempted analysis of (5) as an instance of the ACTIVE-DITRANSITIVE construction, the construction filling the action constituent would be that corresponding to *slept*. The lack of the requisite force-dynamic semantics in the schema associated with sleeping accounts for the sentence's questionable acceptability. Section 3.3.1 discusses related phenomena arising during analysis that likewise depend on semantic compatibility.

We have now completed our extended tour through the constructions licensing one analysis of *Mary tossed me a drink.* As should be clear from the disclaimers along the way, some details have been simplified and complications avoided for ease of exposition. But while the resulting analysis may not capture all the linguistic insights we would like, we believe that issues related to the content of the construction are separable from our primary goal of demonstrating how a broad variety of constructional facts can be expressed in the Embodied Construction Grammar formalism. The next section situates the formalism in the broader context of language understanding, using the constructions and schemas we have defined to illustrate the analysis and simulation processes.

3 ECG in language understanding

Now that we have shown how constructions and schemas can be defined in the ECG formalism, we shift our attention to the dynamic processes that use the formalism for language understanding. Section 3.1 shows how the analysis process finds relevant constructions and produces a semantic specification, and Section 3.2 then shows how the simulation can use such a semspec, along with its associated embodied structures, to draw inferences that constitute part of the understanding of the utterance. In Section 3.3, we consider issues that arise in attempting to account for wider linguistic generalizations and sketch how they might be handled in our framework.

3.1 Constructional analysis

Constructional analysis is a complex undertaking that draws on diverse kinds of information to produce a semantic specification. In particular, since constructions carry both phonological and conceptual content, a construction **analyzer** — essentially, a parser for form-meaning constructions — must respect both kinds of constraint. Analysis consists of two interleaved procedures: the search for candidate constructions that may account for an utterance in context; and the unification of the structures evoked by those constructions in a coherent semspec. Bryant (2003) provides technical details of an implemented ECG analyzer along these lines; here we illustrate both procedures in the vastly simplified situation in which the known constructions consist *only* of the constructions defined in Section 2. The search space is thus extremely limited, and the unification constraints in the example are relatively straightforward.

A typical analysis begins with the phonological forms in an utterance triggering one or more constructions in which they are used. Given our reduced search space, this happens unambiguously in our example: the lexical constructions underlying the words *Mary*, *tossed*, *me*, and *drink* (ignoring the possible verb stem construction with the same form) each trigger exactly one construction; since no additional form constraints remain to be satisfied, the various schemas evoked by the constructions are added to the semspec. The word *a* similarly cues the A-CN-EXPR construction (since the phonological form corresponding to *a* is part of its form pole). The cued construction has an additional com-noun constituent to fill; fortunately, the relevant form and meaning constraints are easily satisfied by the previously cued DRINK construct. The ACTIVE-DITRANSITIVE is triggered by the presence of the other analyzed constructs in the observed order; its constraints are then checked in context. As mentioned in Section 2.2.3, it is this step — in particular, ensuring that the construction's semantic requirements are compatible with those of its verbal constituent — that poses the main potential complication. In our example, however, the schemas as defined are enough to license the bindings in question, and the utterance is successfully analyzed.

We mention in passing some issues that arise when constructional analysis is not restricted to our carefully orchestrated example sentence. The search for candidate constructions grows much harder with larger sets of constructions and their attendant potential ambiguities. The number of constraints to be satisfied — and ways in which to satisfy them — may also make it difficult to choose among competing analyses. Approaches to these essentially computational problems vary in cognitive plausibility, but a few properties are worth noting as both cognitively and computationally attractive. As in our example, analysis should proceed in both bottom-up and top-down fashion, with surface features of the utterance providing bottom-up cues to the constructions involved, and cued constructions potentially supplying top-down constraints on their constituents. An equally important principle (not explicit in our example constructions) is that processing should reflect the graded nature of human categorization and language processing. That is, constructions and their constraints should be regarded not as deterministic, but as fitting a given utterance and context to some quantifiable degree; whether several competing analyses fit the utterance equally well, or whether no analysis fits an utterance very well, the result of processing is the *best-fitting* set of constructions.[17]

[17]Both probabilistic and connectionist models have some of the desired properties; either approach is theoretically compatible with the ECG formalism, where constructions and their constraints could be associated with probabilities or connection weights. See Narayanan and Jurafsky (1998) for a probabilistic model of human sentence processing that combines psycholinguistic data involving the frequencies of various kinds of lexical, syntactic and semantic information. The resulting model matches human data in the processing of garden path sentences and other locally ambiguous constructions.

The semantic specification resulting from the unification process described above is shown in Figure 3.1. Predications and referents are shown in separate sections; in a coherent semspec, all schemas are eventually bound to some predication or referent structure. The depicted schemas and bindings illustrate the main ways in which the constructions instantiated in a successful analysis contribute to the semspec:

- Constructions may include schemas (and the bindings they specify) directly in their meaning poles, or they may evoke them. The three referents and single predication shown can each be traced to one or more constructions, and each schema effects various bindings and type constraints on its subparts and roles.

- Constructions may effect bindings on the roles of their schemas and constituents. Most of the bindings shown in the figure come from the ACTIVE-DITRANSITIVE construction and its interaction with its constituents. Note also that the figure shows a single predication, the result of unifying the predications in the TOSSED and the ACTIVE-DITRANSITIVE constructions; the Drink category has likewise been unified into the appropriate referent schema.

- Constructions may set parameters of their schemas to specific values; these values have fixed interpretations with respect to the simulation. The TOSSED construction, for example, sets its associated predication's setting.time to be past (shorthand for locating the entire event previous to speech time) and its event-structure to be encapsulated (shorthand for running the simulation with most details suppressed, to be discussed in the next section).

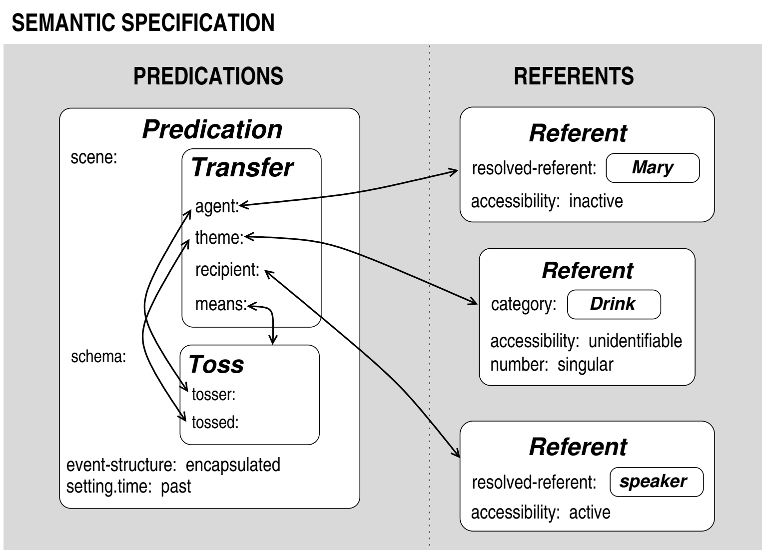

Figure 19: Semantic specification showing predications and referents produced by the analysis of *Mary tossed me a drink*. The overall predication has a Transfer schema as its scene, and a Toss schema (which is also the means of transfer) as its schema. The Transfer schema's agent is bound to the Mary schema, its recipient to the speaker, and its theme to an unidentifiable, singular referent of category Drink.

The figure does not show other schemas evoked by several of the schemas, including the instances of Force-Application in both the Transfer and Toss actions that are unified during analysis. It also does not show how the semspec interacts with discourse context and the reference resolution process. Nevertheless, the semspec contains enough information for an appropriate simulation to be executed, based primarily on the Toss schema and the embodied motor schema it parameterizes. In Section 3.2 we describe how such dynamic knowledge is represented and simulated to produce the inferences associated with our example.

3.2 Simulative inference

We have claimed that constructional analysis is merely a crucial first step toward determining the meaning of an utterance, and that deeper understanding results from the simulation of grounded sensorimotor structures parameterized by the semspec. This section first describes active representations needed for the tossing action of our example (Section 3.2.1), and then discusses how these representations can be simulated to produce fine-grained inferences (Section 3.2.2).

3.2.1 An execution schema for tossing

Executing schemas, or **x-schemas**, are dynamic representations motivated in part by motor and perceptual systems (Bailey 1997; Narayanan 1997), on the assumption that the same underlying representations used for executing and perceiving an action are brought to bear in understanding language about that action. The x-schema formalism is an extension of Petri nets (Murata 1989) that can model sequential, concurrent, and asynchronous events; it also has natural ways of capturing features useful for describing actions, including parameterization, hierarchical control, and the consumption and production of resources. Its representation also reflects a basic division into primitives that correspond roughly to stative situations and dynamic actions.

We use tossing, the central action described by our example utterance, to illustrate the x-schema computational formalism. The Toss schema evoked by the TOSSED construction parameterizes the Tossing-Execution schema, which is the explicit, grounded representation of the sensorimotor pattern used (by an implicit tosser) to perform a tossing action, shown in Figure 3.2.1. Informally, the figure captures a sequence of actions that may be performed in tossing an object (the tossed parameter), including possible preparatory actions (grasping the object and moving it into a suitable starting position) and the main tossing action of launching the object (shown in the hexagon labeled nucleus). This main event may include subsidiary actions that move the object along a suitable path before releasing the object, all with low force. A number of perceptual conditions (shown in the area labeled **percept vector**) must also hold at specific stages of the event: the tossed object must be in the hand (of the tosser) before the action takes place, and afterward it will be flying toward some target. (The target role was not shown in the Toss schema definition from Figure 2.2.2, but would be bound to its spg.goal.)

The x-schema formalism provides a graphical means of representing the actions and conditions of the dynamic event described. An x-schema consists of a set of **places** (drawn as circles) and **transitions** (drawn as hexagons) con-

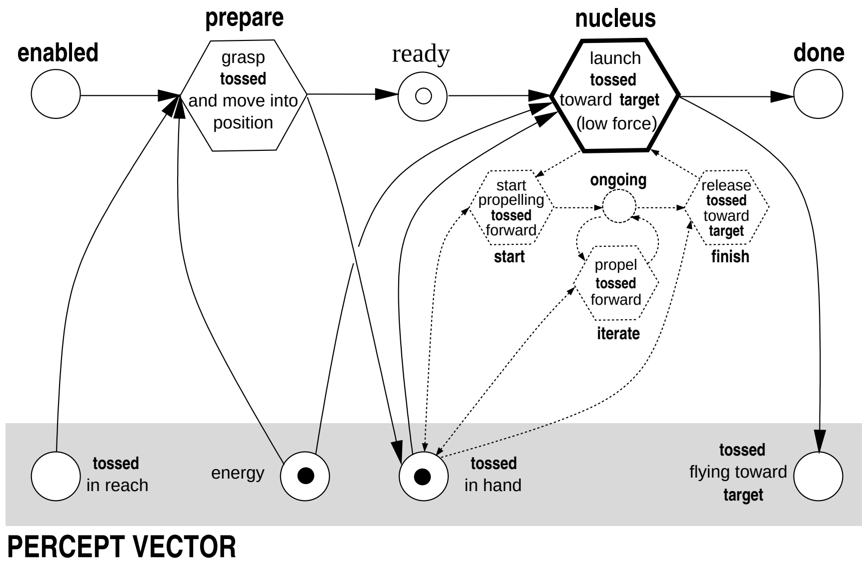

Figure 20: A simplified x-schema representing motor and perceptual knowledge of the tossing action, defined relative to the tosser. (Not all arcs are shown.)

nected by **arcs** (drawn as arrows). Places typically represent perceptual conditions or resources; they may be **marked** as containing one or more **tokens** (shown as black dots), which indicate that the condition is currently fulfilled or that the resource is available. In the stage depicted in the figure, for example, two places in the percept vector are marked, indicating that the object to be tossed is currently in the tosser's hand, and that the tosser currently has some energy. (The figure does not show incoming arcs from separate perceptual input mechanisms that detect whether the appropriate conditions hold.) The other places in the figure are control states for the action (e.g., enabled, ready, ongoing, done, which we discuss in Section 3.2.2). The overall state of the x-schema is defined as the distribution of tokens to places over the network; this assignment is also called a **marking** of the x-schema.

Transitions typically represent an action or some other change in conditions or resources; the ones shown here each correspond to a complex action sequence with subordinate x-schemas whose details are suppressed, or **encapsulated**, at this level of granularity. The figure shows how the tossing x-schema's main launching action could be expanded at a lower level of granularity; the subordinate schemas are drawn with dotted lines to indicate that they are encapsulated. Note that these transitions also have labels relevant to the overall control of the action (prepare, start, finish, iterate, nucleus); again, these will be discussed in Section 3.2.2. Directed arcs (depicted in the figure as arrows) connect transitions to either **input places** (i.e., places from which it has an incoming arc) or **output** places (i.e., places to which it has an outgoing arc).

X-schemas model dynamic semantics by the flow of tokens. Tokens flow through the network along **excitatory** arcs (single-headed arrows), according to the following rules: When each of a transition's (excitatory) input places has a token, the transition is **enabled** and can **fire**, consuming one token from each input place and producing one token in each output place. An x-schema **execution** corresponds to the sequence of markings that evolve as tokens flow through the net, starting from an initial marking. Given the initial marking shown in the figure, the transition labeled nucleus can fire, consuming tokens from each input place. The firing of this transition causes the execution of the subordinate sequence of actions; once these have completed, the transition's firing is complete and tokens are placed in its output places, asserting that the tossed object is now on its trajectory. The overall token movement can be interpreted as the expenditure of energy in a movement that results in the tossed object leaving the tosser's hand and flying through the air.

Most of the arcs shown in the Toss-Execution schema are excitatory; places and transitions may also be connected by **inhibitory** and **enabling** arcs. Inhibitory arcs (not shown in the figure), when marked, prevent the firing of the transitions to which they have an outgoing connection. Enabling arcs (shown as double-headed arrows) indicate a static relationship in which a transition requires but does not consume tokens in enabling places. The figure shows two of the subschemas encapsulated within the nucleus transition as having enabling links from the place indicating that the object is in the tosser's hand; this makes sense since contact with the object is maintained throughout the action of propelling the tossed object. (Again, the arcs are drawn using dotted lines to indicate their encapsulated status.)

The x-schema formalism has just the properties needed to drive simulation in our framework. X-schemas can capture fine-grained features of complex events in dynamic environments, and they can be parameterized according to different event participants. Constructions can thus access the detailed dynamic knowledge that characterizes rich embodied structures merely by specifying a limited set of parameters. Moreover, the tight coupling between action and perception allows highly context-sensitive interactions, with the same x-schema producing strikingly different executions based on only slight changes in the percept vector or in the specified parameters. In the next section we show how x-schemas can be used for fine-grained inference on the basis of an analyzed utterance.

3.2.2 Simulation-based inferences

We complete the discussion of our example sentence by summarizing how the active representations just described are used during simulation. The semspec in Figure 3.1 contains all of the parameters necessary to run the simulation, including the Toss-Execution schema shown in Section 3.2.1, a Transfer schema for the overall event, and the relevant referents. We assume that the semspec referents are resolved by separate processes not described here; we simply use the terms MARY, SPEAKER, and DRINK to refer to these resolved referents. Our example semspec asserts that the specified tossing execution takes place (in its entirety) before speech time. In other words, the nucleus transition is asserted to have fired, placing a token in the done place, all before speech time.

The dynamic semantics described in the last section give x-schemas significant inferential power. The parameterization and marking state asserted by the semspec can be executed to determine subsequent or preceding markings. The asserted marking thus implies, for instance, that the object in hand place was marked at an earlier stage of execution (shown in the figure as part of Toss.ready), and that the energy place has fewer tokens after execution than it did before (not shown in the figure). Part of the inferred trace of evolving markings is shown in Figure 3.2.2, organized roughly chronologically and grouped by the different stages associated with the event-level transfer schema and the action-level tossing schema. We use the labels TRANS and TOSS to refer to the particular schema invocations associated with this semspec.

TRANS.ready		SPEAKER does not have DRINK
TRANS.nucleus		MARY exerts force via TOSS
	TOSS.enabled	DRINK in reach of MARY
	TOSS.ready	DRINK in hand of MARY
	TOSS.nucleus	MARY launches DRINK toward SPEAKER
		MARY expends energy (force-amount = low)
	TOSS.done	DRINK flying toward SPEAKER
		DRINK not in hand of MARY
TRANS.nucleus		MARY causes SPEAKER to receive DRINK
TRANS.done		SPEAKER has received DRINK

Figure 21: Some inferences resulting from simulating *Mary tossed me a drink*.

The stages singled out in the table are, not coincidentally, the same as in the bold labels in Figure 3.2.1. These labels play an important structuring role in the event: many actions can be viewed as having an underlying process semantics characterized by the identified stages. The common structure can be viewed as a generalized action controller that, for a particular action, is bound to specific percepts and (subordinate) x-schemas. This generalized action controller captures the semantics of event structure and thus provides a convenient locus for constructions to assert particular markings affecting the utterance's aspectual interpretation. The resulting inferences have been used to model a wide range of aspectual phenomena, including the interaction of inherent aspect with tense, temporal adverbials and nominal constructions (Narayanan 1997; Chang, Gildea, and Narayanan 1998). For current purposes, it is sufficient to note that certain constructions can effect specific markings of the tossing x-schema:

(6) a. Mary is about to toss me a drink. *(ready place marked)*

b. Mary is in the middle of tossing me a drink. *(ongoing place marked)*

c. Mary has tossed me a drink. *(done place marked)*

As previously mentioned, tense and aspect markers can also force an entire x-schema to be viewed as encapsulated within a single transition, much like the subordinate x-schemas in Figure 3.2.1. This operation has the effect of suppressing the details of execution as irrelevant for a particular level of simulation. In our example sentence, this encapsulated aspect is imposed by the TOSSED construction described in Section 2. As a result, while the full range of x-schematic inferences are available at appropriate levels of simulation, the default simulation evoked by our example may eschew such complex details such as how far the tosser's arm has to be cocked and at what speed a particular object flies.

3.3 Scaling up

In this section we venture outside the safe haven of our example and show how the semantic expressiveness of the ECG formalism can be exploited to model some of the remarkable flexibility demonstrated by human language users. The key observation is that the inclusion of detailed semantic information adds considerable representational power, reducing ambiguities and allowing simple accounts for usage patterns that are problematic in syntactically oriented theories. Section 3.3.1 explores the use of semantic constraints from multiple constructions to cope with ambiguous word senses, while Section 3.3.2 addresses creative language use by extending the formalism to handle metaphorical versions of the constructions we have defined.

3.3.1 Sense disambiguation

Section 2 showed how verbal and clausal constructions interact to determine the overall interpretation of an event, as well as to license (or rule out) particular semantic combinations. As mentioned in Section 2.2.3, this account provides a straightforward explanation for the differing behavior of *tossed* and *slept* with respect to the ditransitive construction, as illustrated by (7a); a similar pattern is shown in (7b) (exemplifying Goldberg's (1995) CAUSED-MOTION construction, not shown here):

(7) a. Mary tossed/*slept me a drink. *(transfer)*

b. Mary tossed/*slept the drink into the garbage. *(caused motion)*

In both examples, the acceptability of the verb *toss* hinges directly on the fact that its associated semantic schema for tossing — unlike that for sleeping — explicitly encodes an appropriate force-dynamic interaction. The examples in (7) involving *tossed* also illustrate how the same underyling verb semantics can be bound into different argument structures. Thus, in (7a) the tossing action is the means by which a transfer of the drink is effected; in (7b) the tossing action is used as part of an event of caused motion.

The same mechanisms can help select among verb senses that highlight different event features:

(8) a. Mary rolled me the ball. *(caused motion)*

b. The ball rolled down the hill. *(directed motion)*

The verb *rolled* as used in (8a) is quite similar to the use of *tossed* in our example sentence, referring to the causal, force-dynamic action taken by Mary to cause the speaker to receive an object. But (8b) draws on a distinct but intimately related sense of the verb, one that refers to the revolving motion the trajector undergoes. A simple means of representing these two senses within the ECG framework is to hypothesize two schemas associated with rolling – one evoking the Caused-Motion schema shown in Figure 2.2.1 and the other evoking a Directed-Motion schema (not shown). Each of the two senses of the verb *rolled* could identify its meaning pole with the means of the appropriate schema. The requisite sense disambiguation would depend on the semantic requirements of the argument structure construction involved. Thus, the ACTIVE-DITRANSITIVE construction's need for a sense involving force-dynamic interaction will select for the caused-motion sense. Although we have not shown the DIRECTED-MOTION construction that accounts for the use in (8b), it could be defined as requiring a verbal argument whose meaning pole binds with the means of a Directed-Motion schema. Note that the differences between the two verb senses are purely semantic: the particular schemas they evoke determine the clausal constructions in which they can participate.

We have focused so far on the interactions between verbal and clausal requirements, but in fact, semantic constraints imposed by features of entities also play a decisive role in constructional sense disambiguation:

(9) a. Mary poured me some coffee. *(pour = means of transfer)*

b. Mary poured me a drink.

(pour = means of creation, with intent to transfer)

The surface similarities between the sentences in (9) obscure their rather different interpretations. Sentence (9a) can be analyzed much as our example from Section 2, with pouring the means by which the transfer of coffee is effected. But in sentence (9b), pouring — which we assume requires a pourable liquid or mass — isn't a direct means of a transfer; in fact, no drink exists until the pouring action has happened. Rather, the pouring action is interpreted as an act of creation, and it is the resulting drink — and not its liquid contents — whose transfer is intended. In this creation variant of the ditransitive construction, the verb specifies not the means of transfer but the means of creation (a precondition for an intended transfer).

Although this situation is more complex than the other sense disambiguation cases, we can still address the inherent ambiguity of the combination of the verb *pour* with ditransitive expressions by examining the interacting constraints posed by its meaning pole and that of its accompanying nominal expressions. In particular, we can define the pouring schema definition as evoking a Creation schema relating the pouring action to a resulting bounded mass; the creation sense of *pour* would have this Creation schema as its meaning pole. The creation variant of the ditransitive construction would also involve a Creation schema, and require the potential nominal filler (*drink*) to be identified with the created object.

3.3.2 Metaphor: a case study in construal

The examples discussed in the last section demonstrate some relatively limited means of applying semantic constraints to problems that resist clean purely syntactic solutions. These mechanisms exploit static properties of the schema formalism, such as subcase relations, evokes relations, constituency and type constraints. By themselves, however, such static properties can encode only conventionalized patterns of meaning. They cannot capture unexpected or unusual patterns of usage; they cannot account for the ubiquity of creative language use, nor for the relative ease with which humans understand such usages. Lexical and phrasal constructions can occur in novel configurations that are nevertheless both meaningful and constrained. Ultimately, in a full-scale language understanding system intended to be robust to varying speakers and contexts, it would be neither possible nor desirable to pre-specify all potential uses of a semantic schema: under the right circumstances, constructs that do not explicitly satisfy a given semantic requirement may still be treated as if they do. Creative linguistic production must be mirrored by creative linguistic understanding. We use the general term **construal** to refer to a widespread set of flexible processing operations that license creative language use, including novel metaphorical and metonymic expressions (Lakoff and Johnson 1980), as well as implicit type-shifting processes that have been termed **coercion** (Michaelis, this volume). In this section we highlight metaphorical construal as a case study of how construal might be treated by a simple extension to the ECG formalism.

Metaphors are a pervasive source of creative language use, allowing speakers to structure a more abstract **target domain** in terms of a more concrete **source domain** (Lakoff and Johnson 1980). Metaphors can be characterized as conventionalized mappings spanning domains of knowledge, typically linking a perceptually and motorically embodied source domain (such as object manipulation, physical proximity, or physical force) onto a relatively more abstract target domain (such as reason, emotional connection, or social action). Some metaphorical uses might be treated simply as conventionalized linguistic units; the use of *delivered* in (10a) below exemplifies a conventionalized use of a metaphor in which the verbal communication of ideas is interpreted as the physical transfer of objects. But metaphors can also structure novel uses of constructions, as shown by the use of *tossed* in (10b). It is this second, creative use of metaphor that we consider an instance of construal and attempt to address in this section.

(10) a. Our president has just delivered the most important speech of his short career.

b. Mary tossed *The Enquirer* a juicy tidbit.

Sentence (10b) bears a surface resemblance to the example sentence analyzed in Section 2, employing several of the same constructions, including the MARY, TOSSED, and A-CN-EXPR. We assume that suitable constructions can be defined to license the remaining (sub)expressions: a *The Enquirer* referring expression whose meaning is a specific news agency; a common noun *tidbit* with two conventionalized senses referring to a small but high-quality unit of food or information, respectively; a similarly polysemous modifier *juicy* that can characterize the consistency of a unit of either information or sustenance; and a construction that licenses the combination of a modifier and a common noun. Given such constructions, could sentence (10b) be analyzed as instantiating the ACTIVE-DITRANSITIVE construction? This potential analysis yields some apparent type mismatches: the food sense of *juicy tidbit* fits the needs of the Transfer and Toss schemas better than the information sense, but the news institution *The Enquirer* cannot be a literal recipient (though not shown earlier, the Receive schema requires a physical entity as its Receiver).

A potential solution to the analyzer's problems is to introduce metaphorical map capturing the intuitions described earlier. Figure 3.3.2 defines a Conduit metaphor that allows a target domain involving Communication to be structured in terms of a corresponding source domain of Object-Transfer; the schemas are not defined here, but their relevant roles are shown in the figure, using notation similar to that used in the schema and construction formalisms. The mappings listed in the **pairs** block assert that a speaker communicating some information to a hearer can be construed as a physical agent sending a physical recipient some object.

```
map Conduit
 roles
  source : Object-Transfer
  target : Communication
 pairs
  source.sender ↦ target.speaker
  source.recipient ↦ target.hearer
  source.object ↦ target.information
```

Figure 22: Example map definition: The Conduit metaphor links a source domain of Object-Transfer to a target domain of Communication.

We assume the analyzer has access to ontological information categorizing *The Enquirer* as an institution that can collect verbal information, making it a suitable hearer in the Communication schema. (We ignore for now the additional metonymy that could link *The Enquirer* to an associated reporter.) Access to the Conduit metaphor could help the analyzer deal with the sentence in (10b) by allowing *The Enquirer* to be construed as a suitable recipient in an Object-Transfer schema. Further analysis is affected by this mapping: If the recipient is metaphorical, then in the most likely analysis the object is metaphorical as well, leading to the selection of the information-related senses of *juicy* and *tidbit*. Similarly, both the overall event and the means by which it was asserted to have taken place must be interpreted as a verbal, rather than physical, acts of transfer.

A hallmark of metaphorical language use is that the mapping of inferences from source to target domain can involve relatively subtle simulative detail. For example, we know from Section 3.2 that *toss*, when used in a ditransitive context, implies that the launching action involves low force. Mapped to the target domain of communication, this inference becomes one of casualness on the part of the speaker. (For a technical description of how metaphorical inference can be performed and propagated to a target domain, the reader is directed to Narayanan (1997).) The inclusion of metaphor maps in the formalism, along with appropriate interfaces to the active simulation, opens the door to creative metaphorical inferences of this kind.

4 Concluding remarks

In this chapter, we have formalized and extended ideas from the construction grammar literature to accommodate the requirements of a larger simulation-based model of language understanding. Constructions in this model serve to evoke and bind embodied semantic structures, allowing language understanding to depend on both specifically linguistic knowledge and general conceptual structures. We have attempted to illustrate the representational properties of our formalism for a variety of linguistic phenomena, including straightforward issues that arise in our example analysis, as well as more complex issues surrounding sense disambiguation and metaphorical inference.

The ECG formalism diverges in several respects from other construction grammars in the literature, in large part due to its non-trivial interactions with both the analysis and simulation processes. It is also motivated and constrained by the need to develop a computational implementation of the overall model, which explains similarities it bears to object-oriented programming languages, as well as to some implementation-oriented versions of HPSG (Pollard and Sag 1994). As we have noted, the presentation in the current work has focused on the formalism itself, simplifying many details to highlight how particular analyses can be expressed within the overall framework. We thus conclude by briefly expanding on some of the issues that motivate ongoing and future research.

Our example constructions use a somewhat restricted set of formal elements. But constructions can have formal realizations that span levels of description, including syntactic, lexical, morphological, phonological, and prosodic cues (for examples, see the discussion of *there*-constructions in Lakoff (1987)). In other work, we have shown how minor extensions allow the formalism to cover a broader range of phenomena in a common notation. For example, the same set of interval relations we use to express syntactic order can be applied to enforce word-internal order of morphemes and to align prosodic contours with lexical hosts.

Our discussion has also deliberately sidestepped complications related to situational and discourse context, but work in progress is exploring how the mechanisms we have introduced can be extended to address discourse-level phenomena in general and mental spaces phenomena (Fauconnier 1985) in particular. The notion of a **space** as a domain of reference and predication fits in especially well with semantic specifications, which are described here as likewise containing referents and predications. We can thus view semspecs as being situated in some space, and these spaces can be evoked, introduced, and constrained by constructions called **space builders**. Other constructions — and their corresponding semspecs — can then be defined relative to the currently active space. For example, a space-building construction X-SAID-Y might be defined to handle reported speech:

(11) Frank said, "Mary tossed me a drink."

Such a construction would presumably introduce an embedded space for the reported speech and require the corresponding constituent to associate its semspec with that embedded space. Given such a constraint, the ME construction — defined in Section 2.1 as identifying its referent with the speaker in the *current* space — would correctly designate the speaker in the embedded space (Frank), and not the global speaker. A more general treatment of mental spaces phenomena awaits further research, but Chang et al. (2002) offer a preliminary sketch of how the formal tools of ECG can be extended to capture interactions between constructions and multiple spaces.

Another dimension of ongoing research focuses on neural (or connectionist) modeling of our computational architectures. Previous models have explicitly related the conceptual structures and mechanisms mentioned here — including image schemas (Regier 1996), x-schemas (Bailey 1997), and metaphor maps (Narayanan 1997) — to neural structures. X-schemas, for example, are defined at the computational level as representing abstractions over neural motor control and perceptual systems (Bailey 1997). At a more detailed connectionist level of representation, Shastri et al. (1999) implement x-schemas as interconnected clusters of nodes. The binding of roles to other roles and to fillers has also been subject to extensive connectionist modeling, in particular as part of the SHRUTI model (Shastri and Ajjanagadde 1993).

Although we have not emphasized this point here, the representational and inferential mechanisms used in the ECG formalism have been restricted to those that can be realized in a connectionist architecture.

As the strands of research mentioned here might suggest, the goals and methods driving both the formalism we have introduced and our broader approach to language understanding are inherently interdisciplinary. Our main goal has been to show how an embodied construction grammar formalism permits fine-grained interactions between linguistic knowledge and detailed world knowledge. The work presented here also, however, exemplifies the methodology of applying converging computational, cognitive and biological constraints to flesh out in formal detail insights from theoretical linguistics. Although many challenges remain, we are hopeful that the ideas we have explored will help to stimulate the continued integration of diverse perspectives on language understanding.

Acknowledgments

This chapter in its various incarnations has benefited from a succession of collaborators and colleagues. The underlying formalism evolved from early collaboration with Mark Paskin and more recent work with Keith Sanders, Jerome Feldman, and Robert Porzel, Johno Bryant, and Srini Narayanan. We also gratefully acknowledge the input of George Lakoff, Charles Fillmore, Josef Ruppenhofer, and other associates of the Neural Theory of Language and FrameNet research groups at UC Berkeley/ICSI. We offer special thanks and our sympathy to two anonymous reviewers of a very early manuscript. All remaining errors are ours.

References

Allen, James F. 1984. Maintaining knowledge about temporal intervals. *Communications of the ACM 26*(1), 832–843.

Bailey, David R. 1997. *When Push Comes to Shove: A Computational Model of the Role of Motor Control in the Acquisition of Action Verbs*. Ph.D. thesis, University of California at Berkeley.

Bryant, John. 2003. Constructional analysis. Master's thesis, University of California at Berkeley.

Chang, Nancy, Feldman, Jerome, Porzel, Robert, and Sanders, Keith. 2002. Scaling cognitive linguistics: Formalisms for language understanding. In *Proceedings 1st International Workshop on Scalable Natural Language Understanding*, Heidelberg, Germany.

Chang, Nancy, Gildea, Daniel, and Narayanan, Srini. 1998. A dynamic model of aspectual composition. In *Proceedings of the Twentieth Annual Meeting of the Cognitive Science Society*.

Croft, William. 1990. A conceptual framework for grammatical categories (or, a taxonomy of propositional acts). *Journal of Semantics 7*, 245–79.

Croft, William. 1991. *Syntactic categories and grammatical relations: the cognitive organization of information*. Chicago: University of Chicago Press.

Croft, William. 2001. *Radical Construction Grammar: syntactic theory in typological perspective*. Oxford: Oxford University Press.

Fauconnier, Gilles. 1985. *Mental Spaces: Aspects of Meaning Construction in Natural Language*. Cambridge, Mass. and London: MIT Press/Bradford.

Fillmore, Charles. 1982. Frame semantics. In *Linguistics in the Morning Calm*, 111–137. Hanshin Publishing Co.

Fillmore, Charles. 1988. The mechanisms of construction grammar. In *Berkeley Linguistics Society*, Volume 14, 35–55.

Goldberg, Adele E. 1995. *Constructions: A Construction Grammar Approach to Argument Structure*. University of Chicago Press.

Janda, Laura. 1991. The radial network of a grammatical category: Its genesis and dynamic structure. *Cognitive Linguistics 1*(3), 269–288.

Johnson, Mark. 1987. *The Body in the Mind*. University of Chicago Press.

Kay, Paul and Fillmore, Charles. 1999. Grammatical constructions and linguistic generalizations: the *What's X doing Y?* construction. *Language 75*(1), 1–33.

Lakoff, George. 1987. *Women, Fire, and Dangerous Things: What Categories Reveal about the Mind*. University of Chicago Press.

Lakoff, George and Johnson, Mark. 1980. *Metaphors We Live By*. University of Chicago Press.

Lambrecht, Knud. 1994. *Information Structure and Sentence Form: Topic, Focus, and the Mental Representations of Discourse Referents*. Cambridge, UK: Cambridge University Press.

Langacker, Ronald W. 1987. *Foundations of Cognitive Grammar: Theoretical Prerequisites*. Stanford University Press.

Langacker, Ronald W. 1991. *Concept, Image, and Symbol: The Cognitive Basis of Grammar*. Cognitive Linguistics Research. Berlin and New York: Mouton de Gruyter.

Murata, Tadao. 1989. Petri nets: Properties, analysis, and applications. In *Proc. IEEE-89*, Volume 77, 541–576.

Narayanan, Srini. 1997. *Knowledge-based Action Representations for Metaphor and Aspect (KARMA)*. Ph.D. thesis, University of California at Berkeley.

Narayanan, Srini and Jurafsky, Daniel. 1998. Bayesian models of human sentence processing. In *Proceedings of the Twentieth Annual Meeting of the Cognitive Science Society*, 84–90.

Pollard, Carl and Sag, Ivan A. 1994. *Head-Driven Phrase Structure Grammar*. Chicago: University of Chicago Press.

Regier, Terry. 1996. *The Human Semantic Potential*. Cambridge, Mass.: MIT Press.

Reichenbach, Hans. 1947. *Elements of Symbolic Logic*. Macmillan.

Shastri, L. and Ajjanagadde, V. 1993. From simple associations to systematic reasoning. *Behavioral and Brain Sciences 16*(3), 417–494.

Shastri, Lokendra, Grannes, Dean, Narayanan, Srini, and Feldman, Jerome. 1999. A connectionist encoding of parameterized schemas and reactive plans. In G. Kraetzschmar and G. Palm (eds.), *Hybrid Information Processing in Adaptive Autonomous Vehicles*, Lecture Notes in Computer Science, Lecture Notes in Artificial Intelligence. Berlin: Springer-Verlag.

Shieber, Stuart M. 1986. *An Introduction to Unification-Based Approaches to Grammar*. Stanford, CA: CSLI Publications.

Slobin, Dan I. 1985. Crosslinguistic evidence for the language-making capacity. In D. I. Slobin (ed.), *Theoretical Issues*, Volume 2 of *The Crosslinguistic Study of Language Acquisition*, Chapter 15, 1157–1256. Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Talmy, Leonard. 1988. Force dynamics in language and cognition. *Cognitive Science* (12), 49–100.

Talmy, Leonard 2000. *Toward a Cognitive Semantics*. Cambridge, MA: MIT Press.