

Composition as a Source of Construction: Some Principles for Pruning Compositions into Constructions

Hsin-I Hsieh

University of Hawai'i

A composition and its related construction can undergo a bidirectional conversion from one to the other. Using the theory of Compositional Cognitive Grammar (CCG), we postulate a Semantic Structure representation as a sentence's compositional structure, from which we derive its constructional structure through principled pruning.

Key words: composition, construction, interaction, Compositional Cognitive Grammar (CCG), principles for pruning, Adele Goldberg

1. Introduction

The version of Construction Grammar proposed by Goldberg (2006) and Goldberg & Jackendoff (2004) analyzes a construction into two parts. "One of them, the verbal subevent, is determined by the verb of the sentence. The other subevent, the constructional subevent, is determined by the construction." (Goldberg & Jackendoff 2004:538) Consider (1):

- (1) Willy watered the plants flat.

The meaning of this resultative construction has two parts. The constructional subevent is (i) Willy cause [plants become flat] and the verbal subevent is (ii) Willy water plants. The two parts are not merely routinely connected. The meaning part (ii) is the means by which the meaning part (i) is obtained. (1) is an example of a Causative property resultative construction, which can be schematically represented as (2) (=G & J (14)):

- (2) Causative property resultative
Syntax: NP₁ V NP₂ AP₃
Semantics: X₁ Cause [Y₂ BECOME Z₃]
 MEANS [verbal subevent]

Thus the construction in (2) has a syntax, or form, and a semantics, or meaning, which has the constructional subevent and the verbal subevent as its two parts. Apparently, one can choose to derive the syntax from the semantics of a construction like (2), as G & J did, or derive the semantics from the syntax, as a referee has suggested to G & J. G & J reject the suggestion, maintaining that “the semantic properties are richer and more nuanced—syntax can be mapped from them, but not vice versa” (Goldberg & Jackendoff 2004:539). To anyone who views transformation from the semantics of a sentence to its syntax as involving the composition, movement, compression, and deletion of the rich semantic elements to result in its distorted and simplified syntax, the rejection by G & J seems perfectly well justified. Yet, the justification is crucially grounded on the adopted mechanism of composition. Whether it is conducted top-down from the whole to the parts as in pre-Minimalism generative grammar and related theories, or processed bottom-up from the parts to the whole as in Minimalism and related approaches, the syntactic as well as semantic configurations are compositional in nature. Since linguistics is quite happy to borrow the concept and technique of composition from mathematicians and logicians, few linguists have ever seriously considered interaction as a viable alternative method of configuration. Einstein is often quoted as having said that God does not play dice with the cosmos when he objected to the theory of quantum mechanics, which discovered unpredictable experimental results due to the interaction of physical particles. Not only quantum mechanics but also evolutionary theories have repeatedly emphasized that evolution is often unpredictable, and that mutations may unpredictably change two variants of one species into two different species. The theory of complex systems (Holland 1999) has proposed that complex systems emerge surprisingly and unpredictably. Thus scientists have been shifting their attention from studying the Newtonian physical universe as a predictable, compositional system to studying the Darwinian biological world as an unpredictable, interactional system.

Conceivably, language is both like a logical, mathematical structure, in which the law of composition holds, and like a biological system, in which the law of interaction prevails. This is why a grammatical unit, such as a morpheme, word, phrase, or sentence can be simplified or complicated and why new units may appear to replace old units. This is why in historical change grammaticalization can cause auxiliaries (Heine 1993) or co-verbs (Li 1980) to develop or evolve from genuine verbs. If language were purely mathematical and not partly biological, such development or evolution would be difficult to explain. We must allow two grammatical units to interact, to influence each other when they are composed together. Specifically, this means two phonemes, two morphemes, two words, two phrases, or two sentences can interact. Such an interaction would allow linguists to derive a rich or complex configuration from a poor or simple configuration. And if this can be done, then the idea of deriving the rich semantics from

the poor syntax in a construction like (2) seems no longer a worse but an equally attractive approach.

Specifically, the semantics in (2) has three elements that get lost in its syntax: Cause and Become from the constructional subevent and Means from the verbal subevent. Appealing to interaction which can add semantic elements that are surprising and unpredictable from the syntax, we can derive the semantics of (2), or (2b) < (i) <Willy, <water, the plants>>, (ii) <plants, <become, at>>, from the syntax of (2), or (2a) < (i) <Willy, <water, the plants>, (ii) <0, at>>. Due to interaction, (2a_{ii}) develops into (2b_{ii}), adding the unpredictable elements plant and at. As we can see, once we adopt the alternative method of interaction, we can approach the linking of the syntax of a construction to its semantics in two equally attractive ways: by the standard method of composition favored by G & J and by the alternative method of interaction. Because composition has been well developed in linguistic theories, it is of course easy to show what it is and how it works. Because interaction is rarely practiced in linguistic theories, it is harder to show what it is and how it works. So our concern in this discussion is twofold. Our first concern is to show how we can generally build up the new method of interaction in contrast to the old method of composition. Our second concern is to show that we can derive the syntax of a construction from its semantics, or equivalently, the construction of sentence from its composition. We will not attempt to derive a composition from a construction, although in principle it would only require a reversal from the process of deriving a construction from a composition. G & J's compositional approach divides the construction into two parts as the constructional subevent and the verbal subevent. Our compositional approach merges or blends these two subevents into a complex event, based on the theory of Compositional Cognitive Grammar (CCG) (Hsieh 1998, 2000, 2004, 2005). Because of the technical sophistication of CCG, we can fairly well predict what elements in the semantics can be deleted or merged with other elements by precisely stated constraints. Thus, benefiting from the extensive observations and illuminating insights by G & J on construction patterns, we offer a technically precise method for building up the composition and construction of a sentence and deriving its construction from its composition. The discussion following this introduction is divided into four parts. Section 2 is a description of four basic types of structure which provides a context for illuminating the contrast between composition and interaction. Section 3 shows how constructions can be derived from compositions. Section 4 briefly describes CCG, preparing for the pruning of compositions into constructions in §5. Section 5 describes the procedure for pruning. Section 6 is a discussion of related issues. And §7 is a conclusion.

2. Four basic types of structure

Intuitively, composition is like making a sandwich and interaction is like cooking a dish of gourmet Chinese food in the wok. The ingredients of a sandwich do not interfere with each other, and therefore making a sandwich is a mechanical, reliable, and predictable process. It is a science. The ingredients of a gourmet Chinese food interfere with each other, and therefore cooking a dish of gourmet Chinese food is an artistic, unreliable, and unpredictable procedure. It is an art. We now try to give a detailed formal account of this intuitive distinction between composition and interaction. To proceed, we begin by describing a system, its components, and its elements as three coarsely grained or graded structures.

A system may be viewed as a vertical hierarchy or a horizontal arrangement. As a vertical hierarchy, a system has higher and lower levels on which Compatible Elements can compose or interact with each other. As a horizontal arrangement, a system has spreading-out components, in which compatible elements can compose or interact with each other. Viewed in either way, a system has three grained or graded structures: the system itself, a component in it, and an element in a component. The graining or grading of a structure is based on its community size and the degree of complexity of the rules of operation in it. Therefore, a structure may be a system, a component, or an element. Each coarse structure has its own increasingly finer-grained sub-structures. Hence a system has its progressively diminished subsystems, a component has its progressively reduced subcomponents, and an element has its progressively decreased sub-elements.

Any two Compatible or Commensurate parts in a structure, which may be a system, a component, or an element, can be related in two ways: they are either independent or inter-dependent. If the two compatible parts are independent from each other, then they dis-interact, and if the two compatible parts are interdependent on each other, then they interact. They dis-interact if a rule or transformation T_d applied to the ordered pair $\langle P_a, P_b \rangle$ always yields a *unique* and *predictable* result or whole, that is, $T_d(\langle P_a, P_b \rangle) = W$. They interact if a rule or transformation T_i applied to the ordered pair $\langle P_a, P_b \rangle$ always yields a *non-unique* and *unpredictable* result or whole, that is, $T_i(\langle P_a, P_b \rangle) = W_k$, with W_k ranging over the variants $W_1, W_2, \dots, W_k, \dots$, and W_n . The dis-interaction T_d produces a result, W , which has a rigid or fixed *quality* and *quantity*, and the Interaction T_i yields a result, W_k , which has a exible or varying quality, or quantity, or both.

To illustrate the difference between a dis-interaction and an interaction, we use the integers once as elements in the system of the ordinary addition and again as elements in the system of an imaginary ‘funny’ addition. In the ordinary addition, the integers have no quality such as color, smell, or shape, but have only a quantity expressed by their numerical values 0, 1, 2, 3, ..., n. In the ‘funny’ addition, the integers have not only

a quantity expressed by their numerical values, but they also have a quality expressed by their three different colors. Some of the numbers are in white, some in red, and some in pink. Therefore, an integer in the ordinary addition has a fixed quality as being colorless, and a fixed quantity as 0, 1, 2, 3, ..., or n . This means, for example, in the ordinary addition, the colorless numbers 2 and 3 always add up to a colorless 5, which has the fixed quality of colorlessness, and the fixed quantity of 5. But now we add up one colorful integer m and another colorful integer n and obtain the result $m + n = k$ in the funny addition. The sum k may be in one of the three possible states: (i) (a state in which) the quality varies, (ii) (a state in which) the quantity varies, and (iii) (a state in which) both the quality and the quantity vary. Thus, adding up a red 2 and a white 3 may produce three possible results: (i) red 2 + white 3 = red 5, white 5, or pink 5 (here the quality varies); (ii) red 2 + white 3 = red 5, red 6, or red 7 (here the quantity varies); (iii) red 2 + white 3 = red 5, white 6, or pink 7 (here both the quality and the quantity vary). The result in equation (i) has a varying quality, the result in equation (ii) has a varying quantity, and the result in equation (iii) has both a varying quality and a varying quantity.

The varying qualities or quantities produced in an interaction can be drawn either from the structure itself or from a source outside of the structure. If they are obtained from the structure itself, they are internal or intrinsic qualities and quantities, and if they are acquired from a source outside of the structure, then they are external or extrinsic qualities and quantities. The examples for the funny addition in equation (i), (ii), and (iii) show varying intrinsic qualities and quantities, since the colors and numbers are all extracted from the original structure, a set of 'funny' integers. To see varying extrinsic qualities or quantities, consider (iv) red 2 + white 3 = pink 5, blue $\frac{1}{2}$, orange $\frac{3}{4}$. To the structure of 'funny' addition of integers, the colors blue and orange are extrinsic colors or qualities, and the rational numbers $\frac{1}{2}$ and $\frac{3}{4}$, being rational numbers and not integers, are extrinsic numbers or quantities. Therefore, equation (iv) illustrates varying extrinsic qualities and quantities. Based on whether a varying quality or quantity is intrinsic or extrinsic, we can make a distinction between a Developing System and an Evolving System. A developing system is a structure, in which interaction produces varying intrinsic qualities and quantities, and an evolving system is a structure, in which interaction creates varying extrinsic qualities and quantities. The extrinsic qualities and quantities arise or emerge surprisingly from an evolutionary system, and we call them Emergent Properties, borrowing a concept from the Complexity Theory.

An emergent property generated in a structure—a system, a component, or an element—can Spread or Perturb Up from a finer structure to a coarser structure, that is, from an element to a component and on to a system, or it can Spread or Perturb Down from a coarser structure to a finer structure, that is, from a system to a component and on to an element. This two-way spreading of emergent properties allows the complex

system to evolve by inducing a change bottom-up from its differently graded sub-parts to itself or by producing a change top-down from itself to its differently graded sub-parts.

Interaction can be described as a transformation achieving a certain *result*, but it can also be depicted as a transformation taking place in a certain *domain*. An interaction may take place within one structure or across two structures. An interaction between two Interactants, Int_a and Int_b , occurring within one structure is an Internal Interaction. An interaction between two Interactants, Int_a and Int_b , occurring across two structures is an External Interaction. If structure s and structure s' are affected extensively by external interaction between them, then they interact externally as two separate structures but internally as two compatible parts in a larger structure containing them both. Thus, internal and external interactions are comparative notions relative to a hosting structure.

If more than two interactants are involved in an interaction, we divided them into pairs. If n interactants are in a interaction, then they can be divided into $nCr = n! / r! (n - r)!$ pairs. If $n=3$, and $r=2$, then $nCr=3$, and if $n=4$, and $r=2$, then $nCr=6$, and so on. Hence, we can use the notion of interaction for two or more interactants.

We use internal interaction (among parts of a structure) and external interaction (among separate structures) as two criteria for differentiating structures of the same rank, whether they are all systems, or components, or elements. We use two binary-valued distinctive features to represent these two criteria: $+/-$ internal (interaction) and $+/-$ external (interaction). Thus, we obtain four types of structure: Type I $<-internal, -external>$, type II $<+internal, -external>$, Type III $<-internal, +external>$, and Type IV $<+internal, +external>$. Type I is a Static Structure, since its compatible parts do not interact among themselves and it does not interact with a separate structure, making it stay unchanged. Type II is a Dynamic Structure, since, although it does not interact with a separate structure, its compatible parts interact with themselves, making it continuously change on its own. Type III is an Affected Structure, since, even though its compatible parts fail to interact among themselves, it interacts with a separate structure or with its environment. Type IV is an Adaptive Structure, because its compatible parts interact among themselves and it interacts with a separate structure or with its environment. Since a structure can be a system, a component, or an element, we can classify systems, components, and elements by a taxonomy formed by these four types of structure.

Among these four types of structures, each lower-numbered type is either less dynamic or less adaptive than its next higher-numbered type, and on account of this increasing degree of dynamism or adaptivity, these four types form an ascending hierarchy of 'vitality' from Type I to Type II to Type III and to Type IV.

We have established a taxonomy of four types of structure, because we want to use this taxonomy to describe the grammar of a language. A grammar is a set of rules which apply in three separate Communication Domains: the Isolation Domain, the Narrative

Domain, and the Discourse Domain. The isolation domain is a set of sentences viewed as isolated and unrelated objects. No speaker or listener of the sentence is assumed. The narrative domain is a set of sentences, each of which is considered as combined with and related to another sentence in a narrative text. A speaker and a listener in their fixed roles are assumed. The discourse domain is a set of sentences, each of which is regarded as combined with and related to another sentence in a discourse fragment. Two participants randomly interchanging their roles as a speaker and a listener are assumed. A grammar thus acts on a set of sentences, each of which has three separate *domain-determined* structures, corresponding to their three communication domains.

Based on how the parts of a sentence interact with each other and on how the sentence itself interacts with another sentence, we can distinguish four *interaction-determined* structures of a sentence, using as a model the taxonomy of four types of structure, which we have just proposed. A sentence has a Compositional Structure (corresponding to Type I structure), which does not interact with a separate structure and whose parts do not interact with each other; a Constructional Structure (corresponding to Type II structure), which does not interact with a separate structure but whose parts interact with each other; a Narrative Structure (corresponding to Type III structure), that interacts with a separate structure, which is its context in a narrative text, but whose parts do not interact with each other; and a Conversational Structure (corresponding to Type IV structure), that interacts with a separate structure, which is its context in a discourse fragment, and whose parts interact with each other. Thus a sentence, *s*, has four interaction-determined structures: (i) the Compositional Structure of *s*, or *Comp(s)*, (ii) the Constructional Structure of *s*, or *Construc(s)*, (iii) the Narrative Structure of *s*, or *Narrative(s)*, and (iv) the Conversational Structure of *s*, or *Conver(s)*. Each of these four structures of a sentence *s* has a syntax or form, *f*, representing its semantics or meaning, *m*, hence a sentence *s* has four form structures representing four meaning structures. The form structures of *s* are *f-Comp(s)*, *f-Construc(s)*, *f-Narrative(s)*, and *f-Conver(s)*. The meaning structures of *s* are *m-Comp(s)*, *m-Construc(s)*, *m-Narrative(s)*, and *m-Conver(s)*. In the standard linguistic practice, a sentence has a syntactic structure, a semantic structure, and a pragmatic structure. The syntactic structure is a form structure, the semantic structure is a meaning structure, and the pragmatic structure is a meaning deduced from a narrative or discourse context, even though in practice its form is not well described. In our new approach, the *f-Comp(s)*, *f-Construc(s)*, and *f-Narrative(s)* divide the standard syntactic structure, the *m-Comp(s)*, *m-Construc(s)*, and *m-Narrative(s)* split up the semantic structure, and the *f-Conver(s)* and *m-Conver(s)* break up the form and meaning of the pragmatic structure.

Since the four structures of a sentence are compatible phases of a sentence, they may interact among themselves. In other words, for the sentence *s*, *Comp(s)*, *Construc(s)*,

Narrative(s), and Conver(s) can all interact with each other. If the interaction generates intrinsic qualities and quantities, then the sentence *develops* and if the interaction generates extrinsic qualities and quantities, which are emergent properties, then the sentence *evolves*.

The composition of s, or Comp(s), and the construction of s, or Construc(s), can be converted from one to the other in either direction. We use a simple example to show how this bidirectional mapping can be performed.

3. From compositions to constructions

Consider (3) and (4):

- (3) a. Zhāngsān hē zuì le jiǔ.
Zhangsan drink drunk perf. wine
'Zhangsan got drunk by drinking wine.'
- b. (i) Zhāngsān hē jiǔ, (ii) Zhāngsān zuì.
'(i) Zhangsan drinks wine, (ii) Zhangsan is drunk.'
(We disregard the perfective aspect marker *le*.)
- c. <Zhāngsān, <<hē, zuì>, jiǔ>>
'Zhangsan drink-drunk wine.'
- (4) a. Zhāngsān hē guāng le jiǔ.
Zhangsan drink empty perf. wine
'Zhangsan drank up the wine.'
- b. (i) Zhāngsān hē jiǔ, (ii) jiǔ guāng.
'(i) Zhangsan drinks wine, (ii) wine is totally consumed.'
- c. <Zhāngsān, <<hē, guāng>, jiǔ>>
'Zhangsan drink-empty wine.'

(3a) and (4a) are the surface sentences, whose postulated compositional structures are (3b) and (4b), and whose postulated constructional structure are (3c) and (4c), respectively. We can see that if we employ a simple-minded composition to build (3a) and (4a) by combining the two lexical items in the resultative compounds *hē zuì* 'drink-drunk' and *hē guāng* 'drink-empty', we would have a semantic problem. We cannot explain why the same resultative compound pattern should have yielded two different meanings. In (3a), the one who gets drunk is the subject Zhangsan rather than the object *jiǔ* 'wine', but in (4a) the thing that gets totally consumed is the object *jiǔ* 'wine' rather than the subject Zhangsan. If composition is really the mechanism employed, the two compounds, *hē zuì* 'drink-drunk' and *hē guāng* 'drink-empty' should have both involved the same

person Zhangsan or the same thing *jiǔ* ‘wine’. They should both have identified either Zhangsan as the drunken person or the consumed thing or *jiǔ* as the drunken person or the consumed thing. But in fact, they have not. We apparently need interaction for our analysis. Invoking interaction, we combine *hē* ‘drink’ and *jiǔ* ‘wine’ to obtain, in addition to ‘Zhangsan drinks wine’, ‘the drinker gets drunk’, which is an unpredictable meaning element. We also combine *hē* ‘drink’ and *guāng* ‘empty’ to obtain, in addition to ‘Zhangsan drinks wine’, ‘the wine gets totally consumed’, which is also an unpredictable meaning element. Of course, one might object by saying that, without interaction and with composition, the real world knowledge ensures that we get the supposedly surprising meaning element in both (3a) and (4a). Yes, but the mathematician does not assume or allow real world knowledge for their notion of composition. To allow the real world knowledge to enter into a composition, we must still use the tool of interaction. In effect, interaction allows existing or emergent real world knowledge to come in from the back door to join the composition to achieve the full meaning.

This is how we can use interaction to analyze (3a) and (4a). But we can use a more sophisticated composition to analyze (3a) and (4a), and from the syntax of the compositional structure we can derive the syntax of the constructional structure. During the derivation, the meaning of the construction is preserved or inherited from the meaning of the composition and so we do not have a problem with meaning. This means that, for an *s*, we can simplify or *prune* the analysis tree of its compositional structure, or *Comp(s)*, to derive the analysis tree of its constructional structure, or *Construc(s)*. The compositional structures of (3a) and (4a) turn out to be (3b) and (4b). (3b) has two subparts, and so has (4b). Part (i) of (3b) says that Zhangsan drinks wine and part (ii) says that Zhangsan gets drunk. Part (i) of (4b) says that Zhangsan drinks wine and part (ii) says that the wine gets totally consumed. The semantic relation between the action in part (i) and the result in part (ii) is only implied, but it could be explicitly indicated by the formula *Event=<action, result>*.

We can see that (3b) and (4b) give sufficiently detailed information. Especially, (3b) indicates that Zhangsan is the one who gets drunk and (4b) indicates that the wine is the thing that gets totally consumed. With the meaning difference taken care of, we transform the compositional structures (3b) and (4b) into the constructional structures (3c) and (4c), yielding the surface structures (3a) and (4a).

We have shown that, for any *s*, we can in principle build the compositional structure for *s*, or *Comp(s)*, illustrated by (3b) and (4b), build the constructional structure for *s*, or *Construc(s)*, exemplified by (3c) and (4c), and then transform the one into the other. From *Construct(s)*, we derive the surface form of *s*, exemplified by (3a) and (4a). We now discuss in detail the process of transformation from *Comp(s)* to *Construc(s)*.

To prepare for our discussion, we give a brief description of the Compositional

Cognitive Grammar (CCG). Readers interested in the technical details of CCG can consult the Appendix for additional information.

4. Compositional Cognitive Grammar

The theory of Compositional Cognitive Grammar (CCG) was proposed by Hsieh (2000, 2004) and explored by Hsieh and associates at the University of Hawai'i (Wang 1998, Chang 1998, Gammon 1999, Hayden 1997, Cheng 2004, Parinyavottichai 2006, Chen 2004). CCG postulates five components or levels of a grammar, which form a Descending Hierarchy (DH) of representational complexity from meaning to form and a reverse Ascending Hierarchy (AH) of representational complexity from form to meaning. Within DH, the top and first level is the Image Structure (IS), whose elements are Image Structure representations (ISrr). The second level is the Semantic Structure (SS), whose elements are Semantic Structure representations (SSrr). The third level is the Thematic Structure (TS), whose elements are Thematic Structure representations (TSrr). The fourth level is the Functional Structure (FS), whose elements are Functional Structure representations (FSrr). The fifth and bottom level is the Constituent Structure (CS), whose elements are Constituent Structure representations (CSrr). We assume that the meaning of a sentence is richer than what its form can express. Therefore, the ISr(k) of a sentence k has a more complex binary structure tree, or bst, than does its CSr(k), hence it is 'richer' and 'deeper' than CSr(k). Conversely, CSr(k) has a less complex bst than does ISr(k), and hence it is 'simpler' or 'shallower' than ISr(k).

IS is the meaning extreme, and CS is the form extreme. Between these two extremes we find SS, which blends meaning and form to serve as bridge connecting IS to CS in both directions. Via SS, the chain of transformations applies bi-directionally, connecting meaning to form in the top-to-bottom direction and form to meaning in the bottom-to-top direction. The transformations may apply downward and take the form $IS \rightarrow SS \rightarrow TS \rightarrow FS \rightarrow CS$, or upward and take the form $CS \rightarrow FS \rightarrow TS \rightarrow SS \rightarrow IS$. On each of the five levels, for each sentence, k, a procedure of composition creates its representation as bst(k). A transformation maps the bst(k) on one level to the bst(k) on another level. Therefore, for any k, we have the DH formed by ISr(k), SSr(k), TSr(k), FSr(k), and CSr(k), and have the AH as its reverse. A chain of transformations apply to this DH to convert ISr(k) ultimately to CSr(k), and a reversed chain of transformations apply to the AH to convert CSr(k) ultimately to ISr(k). The puzzle of how to derive a deep and rich meaning in the ISr(k) from the shallow and simple form in CSr(k) can be solved. We assume that in each step of the upward transformation from CSr(k) to ISr(k), pragmatic information, conversational context, and real world knowledge consecutively supply needed but missing meaning elements.

In the SS, each SSr(k) is both a semantic Event and a syntactic Action. An Action, or AC, is either simple or complex. We compose a complex AC, or c-AC, from a finite number of simple ACs, or s-ACs. An s-AC is a particularized or particular ACF, or PACF. An ACF is a basic general sentence pattern cast in a most generalized and abstract mode. There are a total of 28 ACFs, serving as the ‘building blocks’ for an SSr(k). An ACF or more precisely a PACF indicates a ‘proposition’ or a ‘semantic sentence’. As Figure 1 shows, the subject of the semantic sentence is the Initiator (I), the verb is the Act (A), and the object is the Receiver (R). We avoid using terms such as agent, patient, and theme, since these are reserved for the TS (Thematic Structure). The A and R combine into Complex Act, or A’. The I and A’ combine into Action, or AC. Figure 1 provides an illustration. There the tree is a bst, below which line (i) shows a general ACF, line (ii) shows ACF <1>, and line (iii) shows PACF <1>, where the Full Verb (FV) is spelled out as F-buy, for ‘Full Verb: buy’. The general ACF has one entity e_i representing its I, another entity e_j representing its R, and an Abstract Verb (AV) representing its A. Line (ii) makes the AV a Full Verb (FV), and the e_i and e_j the indexed variables x_i and y_j . Later, in the CSr, the x_i and y_j will be converted into suitably placed NP_i and NP_j, such as John_i and a car_j, by a language-specific procedure of ‘instantiation’. Thus, this PACF <1> is the deep source of a shallow sentence like (4) John_i buys/bought a car_j, when tense is added. The idea of indexed variables has originated with McCawley (1971) and Huang (1992). In ACF <1> in line (ii), the AV is a Full Verb (FV), but elsewhere in other ACFs, it may be a Half Verb (HV) or a Grammatical Verb (GV). These three kinds of AV are realized differently in the CSr. Full verbs will be realized as genuine verbs or adjectives, Half Verbs as prepositions, conjunctions, adverbs, auxiliaries, aspects, tenses, etc., and Grammatical Verbs as demonstratives, determiners, grammatical particles, affixes, etc.

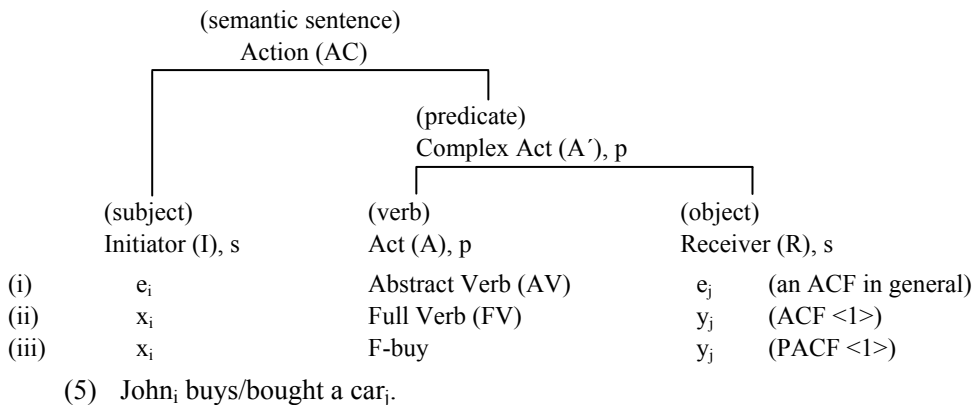
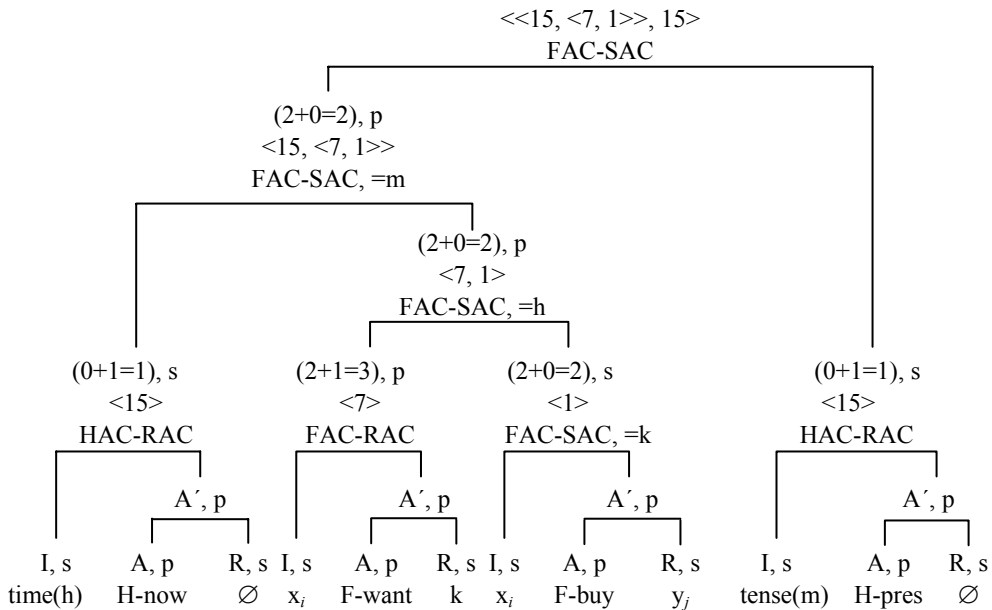


Figure 1: ACF and PACF

In other types of ACFs, the e_i or e_j can be represented as a simple constant h or k , signaling an embedded clause placed outside of the ACF. It can also be represented as a complex constant $f(h)$ or $f(k)$, indicating a feature or property, such as the tense, aspect, time, location, manner, or speed of the AV in another ACF, which is identified as h or k and placed outside of the original ACF. The e_i or e_j can also be represented by the empty symbol, \emptyset , to indicate un-represented I or R. Figure 2 illustrates simple constants, complex constants, and empty symbols.



(6) Now John wants to buy a car.

Figure 2: SSr(6). Simple constant k indicates an embedded clause, complex constants $\text{time}(h)$, $\text{tense}(m)$ usher in time adverb *now* and present tense marker $-s$, and empty symbol \emptyset indicates that in both instances of $\langle 15 \rangle$ R is empty. Also x_i and y_j anticipate John_i and a car_j .

The bst in Figure 2 has four PACFs or s-ACFs: $\langle 15 \rangle$, $\langle 7 \rangle$, $\langle 1 \rangle$, and $\langle 15 \rangle$. The k in the R position of $\langle 7 \rangle$ indicates that the PACF $\langle 1 \rangle$, marked ' $=k$ ', is an embedded clause. The pair $\langle 7, 1 \rangle$ means x_i wants to do something, and that something is to buy y_j . The $\text{time}(h)$ in the I position of $\langle 15 \rangle$ means the time of occurrence for the event $\langle 7, 1 \rangle$, marked as ' $=h$ ', is now. The $\text{tense}(m)$ in the I position of $\langle 15 \rangle$ means the tense for the event $\langle 15, \langle 7, 1 \rangle \rangle$, marked as ' $=m$ ', is the present tense. The symbols 0 in the R position in both instances of $\langle 15 \rangle$ means that the position is empty. Finally, the indexed

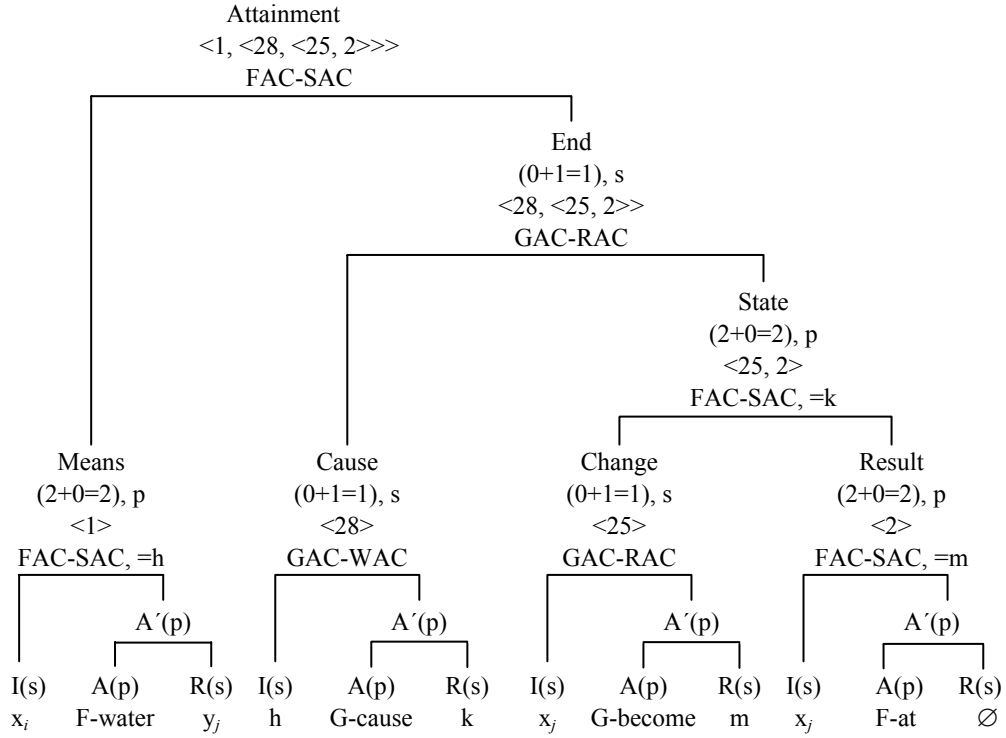
variables x_i and y_j anticipates John_i and a car_j . This is a complex AC, or c-AC, whose Verb-noun Compound Type is FAC-SAC, and which is composed from four simple ACs, or s-ACs, identified as the PACFS $\langle 15 \rangle$, $\langle 7 \rangle$, $\langle 1 \rangle$, and $\langle 15 \rangle$. Finally, the indexed variables x_i and y_j will be instantiated by $\text{NP}_i = \text{John}_i$ and $\text{NP}_j = \text{a car}_j$ in the CSr(6). SSr(6) will be converted into CSr(6), following a set of rules of conversion. These rules turn the AVs into their concrete shapes, move subtrees in the bst, delete or move and merge lexical elements, and, for any k , instantiate x_k as NP_k . (see Hsieh 1998 for these rules).

How do we assign a verb-noun compound type to an AC, which may be an s-AC or a c-AC composed from several s-ACs? An s-AC is just a PACF derived from one of the 28 ACFs. A PACF, viewed as a Verbal Type (vt), is a Full Verb Action (FAC) if its AV is a Full Verb, a Half Verb Action (HAC) if its AV is a Half Verb, and a Grammatical Verb Action (GAC) if its AV is a Grammatical Verb. These are the verbal types (vtt). A PACF, viewed as a Noun Type (nt), is a Solitary AC (SAC) if it contains no (simple or complex) constants, a Receptive AC (RAC) if it contains one constant, a Warm AC (WAC) if it contains two constants. These are the noun types (ntt). Combining the verb type (vt) and the noun type (nt), a PAC has a verb-noun compound type (vnt), such as FAC-SAC, FAC-RAC, FAC-WAC, etc. The vnt of an s-AC is therefore determined by its own bst. The vnt of a c-AC is obtained by rules of computation, which recursively compute the vnt of a c-AC from the vnt of its composing s-ACs. First we compute for the vt and then we compute for the nt, and finally we combine the two results into the vnt (see Tables 2 & 3 in the Appendix). In the bst of an SSr, every pair of composed elements or parts have a distinction in ‘primacy (ranking or degree)’ between ‘primary’ (‘p’) and ‘secondary’ (‘s’), corresponding to the standard distinction between ‘head’ and ‘modifier’ in syntax. The p and s ranks within a PACF or an s-AC are pre-determined. Between two ACs, the p and s ranks are obtained through computation. First, we combine the vt and nt degrees or weights of primacy for one AC into a combined weight, m , and second we combine the vt and nt degrees or weights of primacy for the other AC into a combined weight, n . Then the two resulting weights m and n are compared. If m is greater than n , then m is translated as p and n is translated as s (see Tables 4 & 5). If m is equal to n , then we invoke an auxiliary criterion, such as Tai’s (1985) principle of temporal sequence, which will assign s to the preceding event and p to the following event in a temporal sequence.

5. Pruning a composition into a construction

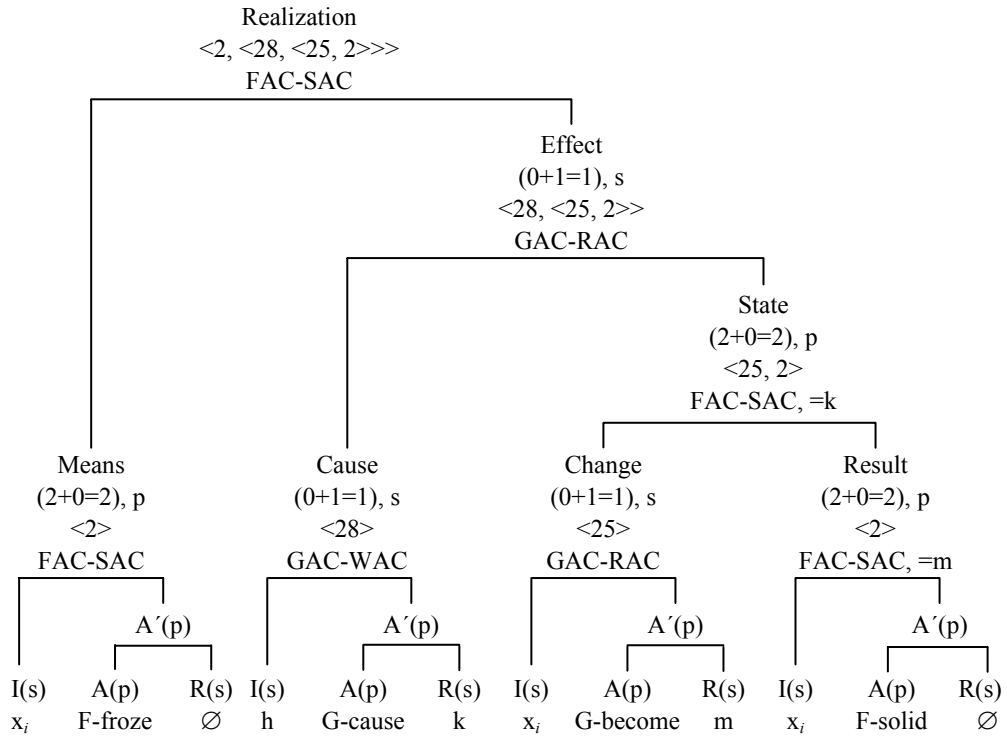
We are ready to show how we can prune a composition into a construction by using some of the sentences cited by G & J as below:

- (7) (=G & J (13)) Willy watered the plants at.
 (8) (=G & J (6a)) The pond froze solid.
 (9) Bill rolled the ball down the hill.
 (10) (=G & J (12a)) Bill spit out the window.
 (11) (=G & J (17b)) The wagon creaked down the road.



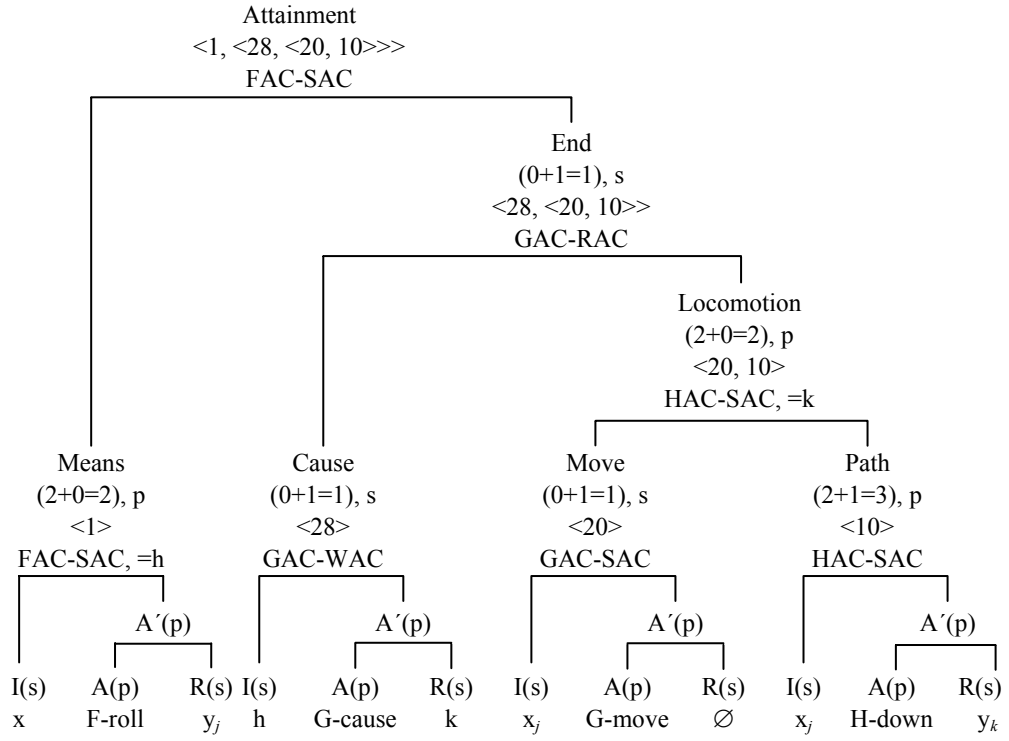
- (7) (=G & J (13)) Willy watered the plants at.

Figure 3: R-SSr(7)



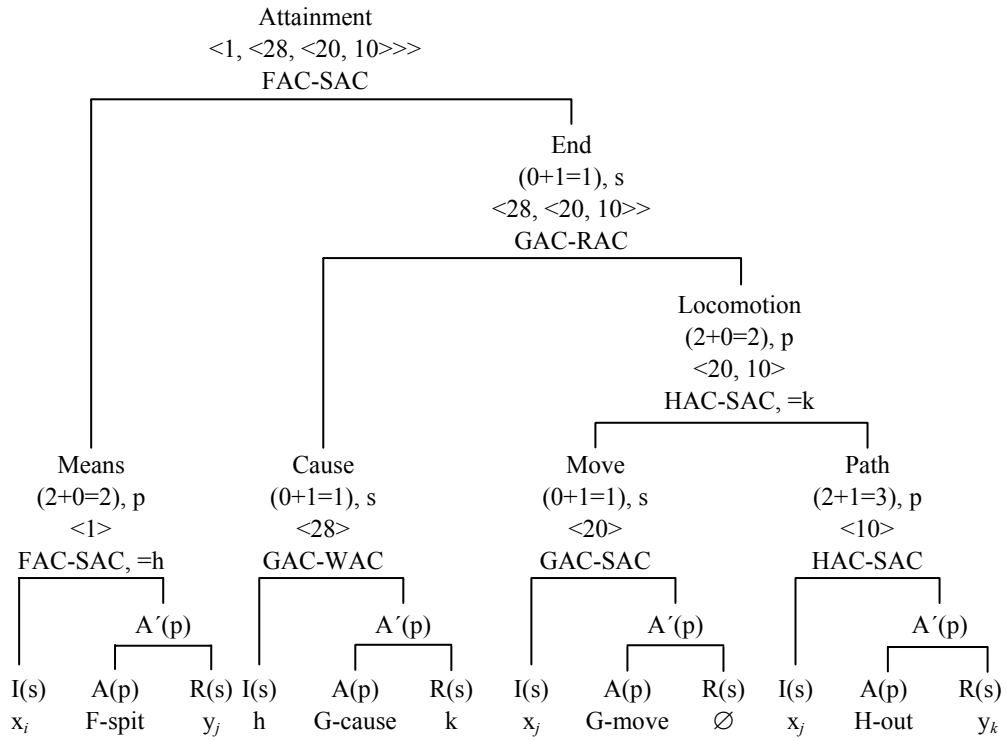
(8) (=G & J (6a)) The pond froze solid.

Figure 4: R-SSr(8)



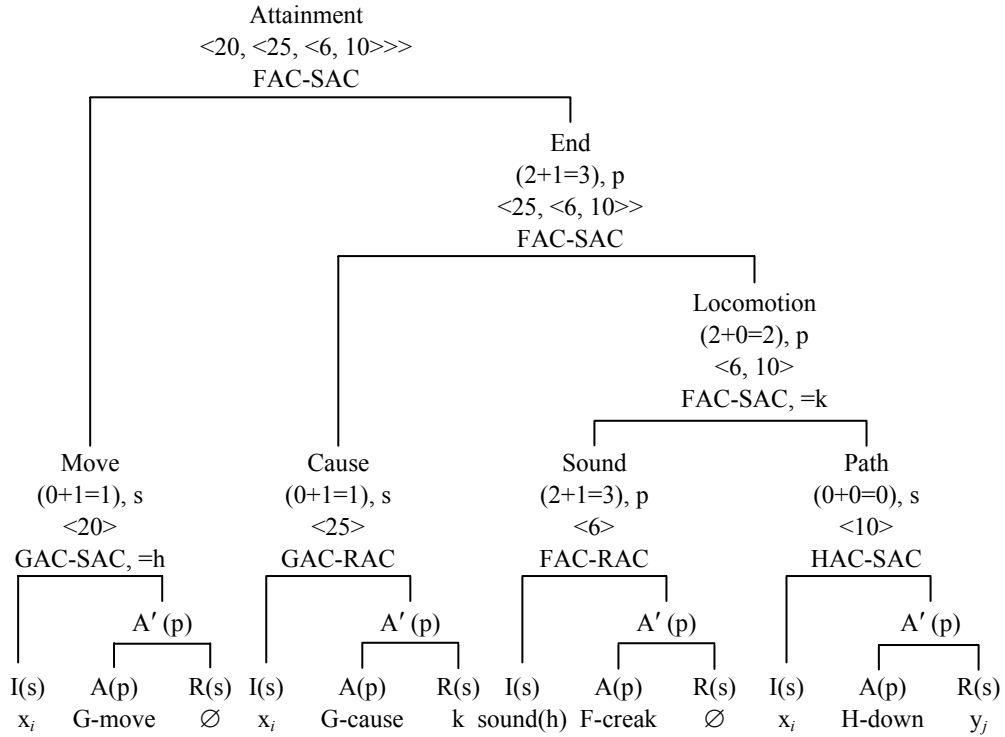
(9) Bill rolled the ball down the hill.

Figure 5: R-SSr(9)



(10) (G & J (12a)) Bill spit out the window.
 ($y_j = x_j$ = saliva is deleted or unrealized.)

Figure 6: R-SSr(10)



(11) (=G & J (17b)) The wagon creaked down the road.

Figure 7: R-SSr(11)

Let us first look at Figure 3 for Refined SSr(7), or R-SSr(7). We give each s-AC and c-AC in R-SSr(7) its semantic category. AC <1> is Means, AC <28> is Cause, AC <25> is Change, AC <2> is Result, AC <25, 2> is State, AC <28, <25, 2>> is End, and finally AC <1, <28, <25, 2>>> is Attainment. As we examine R-SSr(7), we can see that the Semantics part in (2) Causative property resultative, provided for sentence (1) by G & J is captured precisely by the bst of R-SSr(7). We see that the Means is captured by <1> Means, the Cause by <28> Cause, the Become by <25> Become, and Z3 by <2> Result. These semantic categories in SSr(7) are composed into other categories according to a set of general rules of semantic computation, so that Change and Result compose into State, Cause and State compose into End, and Means and End compose into Attainment.

In describing the conditions for pruning a composition into a construction, we want to pay particular attention to the s or p status of each s-AC or c-AC. We want to show that there is a fairly general principle for pruning a compositional tree like R-SSr(7) into a constructional tree or a construction like (7). This general Principle for Pruning is:

(PfP)(If deletion occurs), delete the elements or subtrees ranked *s*(secondary) in an *R-SSr(k)*, for any sentence *k*. In the case of *R-SSr(7)*, the deleted subtrees are <28> and <25>, both ranked *s*. The End, also ranked *s*, is not deleted. This means that being an *s* element or subtree is necessary but not sufficient for its deletion. PfP applies quite persistently. In *R-SSr(7)*, Cause and Change as *s* (secondary) events are deleted, following PfP. In *R-SSr(8)* (Figure 4), Cause and Change as *s* are deleted, according to PfP. In *R-SSr(9)* (Figure 5), Cause and Move as *s* are deleted, obeying PfP. In *R-SSr(10)* (Figure 6) Cause and Move as *s* are deleted, following PfP. Here the $y_j = x_j$ for saliva is an *s* element and is also deleted. Finally, in *R-SSr(11)* (Figure 7), Move and Cause are *s* and deleted, according to PfP.

The sentences (7) through (11) are some of the conspicuous patterns of Resultative Constructions. We have shown that we can postulate *R-SSr(7)* through *R-SSr(11)* and derive (7)-(11) from them, by deleting the secondary ACs or elements using the PfP. Therefore, we have proved that one way to account for these constructions is to assume that they are derived from their source compositions by applying PfP for the deletion of the secondary ACs or elements.

Apparently, pruning involves more principles than just the PfP. Consider (12a) and (12b):

- (12) a. Zhāngsān yòng máobǐ xiě zì.
 Zhangsan use brush write character
 'Zhangsan uses a brush to write characters.'
 b. Zhāngsān xiě máobǐ.
 Zhangsan write brush
 'Zhangsan writes characters with a brush.'

R-SSr(12a), coarsely sketched as *C-SSr(12a)* <Instrument <10> (x_i H-*yòng* y_j), Performance <1> (x_i F-*xiě* y_k)>, serves as the composition from which to derive (12b) as a related construction. In *C-SSr(12a)*, *yòng máobǐ* 'with a brush' is the Instrument event, and *xiě zì* 'write characters' is the Performance event. Instrument is *s* and Performance is *p*. Within Instrument, *máobǐ* 'brush' is *s*, and *yòng* 'with' is *p*. Within Performance, *zì* 'character' is *s*, and *xiě* 'write' is *p*. When *R-SSr(12a)* is transformed into (12b), the *p* element *yòng* 'with' in the *s* Instrument is deleted, contradicting PfP. But the *s* element, *zì* 'characters', in the *p* Performance is deleted, following PfP. Then there is a movement of the *s* element *máobǐ* 'brush' to the position vacated by the *s* element *zì* 'character', based on a movement rule that moves an *s* element from the *s* Instrument to the empty *s* position in the *p* Performance. Therefore, we can see that the pruning or reshaping of a composition into a construction can involve principles other

than the PfP. However, as we can see in this example, the PfP still remains active as a key principle. And this justifies our focusing on the PfP as a principle for pruning compositions into constructions.

6. Discussion

In this section, I respond positively and gratefully to the comments and suggestions made by an encouraging reviewer. The reviewer would like to know on which side I stand in the debate between Generative Grammar—especially in its minimalism formulation—and Construction Grammar. The reviewer also would like to see me try to simplify my SSr trees to make them more readable. Finally, the reviewer hopes that in the future I can find a way to analyze the more challenging constructions such as *drink the pub dry* in English. I now reply item by item.

Minimalism (Chomsky 2001) assumes strict composition and continues to focus on the syntax of a sentence as the form which by a homomorphism relates to the meaning of the sentence. Syntactic rules and their interaction are the primary concern in this theory, and the unexpected meaning or nuance that surprisingly emerges from the syntax of a sentence is not a major concern. However, it is precisely these unanticipated meaning elements that Construction Grammar aims to describe and explain. Goldberg's one favorite example showing that construction meaning is not an epiphenomenon of the compositional meaning of a sentence is the following pair:

- (13) a. John baked a cake for Mary.
b. John baked Mary a cake.

These two sentences are not exactly identical in meaning. (13b) implies that Mary is the beneficiary of John's baking, but (13a) simply means that John does the baking on behalf of or for the sake of Mary. There are many similar pairs involving a prep-NP adjunct as in (13a), in contrast to a ditransitive as in (13b), and this supports Goldberg's claim that (13a) is a construction, distinct from (13b), which is another construction.

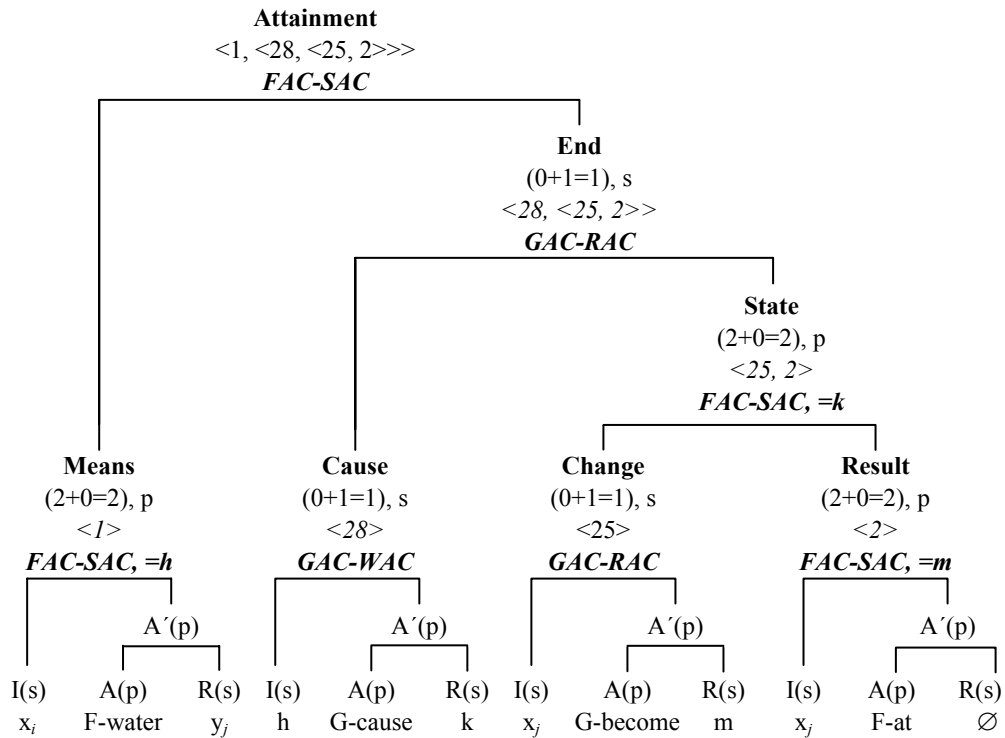
I proposed to derive a construction from its compositional source. For (13a), the compositional root is the same as (13a) itself. For (13b), I shall have to postulate something like (13b'), which a syntactic transformation would convert to (13b):

- b'. (i) John baked a cake for Mary and (ii) John intended Mary to be its beneficiary.

Having been Goldberg's student twice, once in her class at the 2005 LSA Linguistic

Institute at MIT and again in her class at the 2007 LSA Linguistic Institute at Stanford, I know she mistrusts composition and cannot agree with me. But fundamentally, I stand on her side and believe that for any sentence, there is not only a sentential meaning but also an accompanying construction meaning. Given a sentence like (1), we can follow G & J and analyze it by means of a merger of two meaning parts as in (2), or we can employ CCG to analyze it by composition of its sub-parts (7). The analytical approaches may differ, but the belief that there are constructions, as Goldberg and others see them, seems true.

The SSr trees are somewhat complicated and one way to simplify them is to keep the original tree but to write different components of the tree in different fonts, so as to give the reader an easy way to focus on any chosen component of a tree. In this simplification, Figure 3 would reappear as Figure 3':



(7') (=G & J (13)) Willy watered the plants at.

Figure 3': R-SSr(7) simplified

As Figure 3' shows, there are four components in a tree and its sub-trees, and each

component is written in a different font. Guided by these different fonts, the reader can focus on a chosen component. For example, given the result event in Figure 3', the reader can focus on its semantic category indicated by a bold face, **Result**; its p or s status by a regular font, (2+0=2), p; its ACF number, <2>, by an italic; and its compound type, ***FAC-SAC, =m***, by a combined bold face and italic. In this way, the reader can read an entire tree as well as any of its sub-trees by focusing on one of its four components. Compared to a more complex tree such as one in HPSG (Sag et al. 2003), the simplified SSr tree in CCG seems more readable. Regrettably, although the trees can be simplified, the CCG grammar, described in many concepts, terms, and symbols, is not easy to simplify, if we wish to maintain the explicitness, precision, and rigor of the grammar.

An English phrase such as *drink the pub dry* or a Chinese phrase such as *dié pò yǎn jìng* 'totally surprising, totally mystifying' 跌破眼鏡 is a genuine challenge to a Construction Grammar analysis. This is because using CCG, we would have to postulate some sub-trees to encode the unexpected meaning elements and allow transformations to delete, alter, or merge them. However, other than our postulated PfP, we know very little about the constraints on such transformations. Let us hope that as the movement of Construction Grammar makes progress, we may understand this problem better and may find a more reasonable solution.

7. Conclusion

As G & J see it, a construction has a constructional subevent and a verbal subevent, and the two subevents are connected into the whole constructon. We have shown that we can blend the two subevents into the SSr(k), for any sentence k, in CCG. With SSr(k) as a composition, we can derive CSr(k) as its construction, which yields the surface form of the sentence k. We also showed that PfP is the key principle for pruning a composition as SSr(k) into a construction as CSr(k), even though we are aware that other pruning principles are also needed for a full account of the pruning.

Appendix

Tables for classification and computation in CCG
(These tables are borrowed from Hsieh 2004, with revisions.)

Table 1: Action Frames (ACFs) Classified

	<I, <A, R>>				<I, <A, R>>				<I, <A, R>>			
SAC	(1)	x_i	FV	y_j	(10)	x_i	HV	y_j	(19)	x_i	GV	y_j
	(2)	x_i	FV	\emptyset	(11)	x_i	HV	\emptyset	(20)	x_i	GV	\emptyset
	(3)	\emptyset	FV	y_j	(12)	\emptyset	HV	y_j	(21)	\emptyset	GV	y_j
	(4)	\emptyset	FV	\emptyset	(13)	\emptyset	HV	\emptyset	(22)	\emptyset	GV	\emptyset
RAC	(5)	h	FV	y_j	(14)	h	HV	y_j	(23)	h	GV	y_j
	(6)	h	FV	\emptyset	(15)	h	HV	\emptyset	(24)	h	GV	\emptyset
	(7)	x_i	FV	k	(16)	x_i	HV	k	(25)	x_i	GV	k
	(8)	\emptyset	FV	k	(17)	\emptyset	HV	k	(26)	\emptyset	GV	k
	(27)	x_i	k	\emptyset								
WAC	(9)	h	FV	k	(18)	h	HV	k	(28)	h	GV	k
	FAC				HAC				GAC			

FV = Full Verb (verb, adjective)

HV = Half Verb (preposition, conjunction, adverb, auxiliary, aspect, tense, etc.)

GV = Grammatical Verb (demonstrative, determiner, particles, affixes, etc.)

F/H/GAC = Full/Half/Grammatical Verb AC

SAC = Solitary AC, containing no unsaturated constants

RAC = Receptive AC, containing one unsaturated constant

WAC = Warm AC, containing two unsaturated constants

Notes:

Within an ACF, we may find (i) an indexed variable v_m , written as x_i for I and y_j for R, with i and j referring to distinct participants, or (ii) a simple constant c , written as h for I and k for R, or (iii) a complex constant $f(c)$, written as $f(h)$ for I and $f(k)$ for R, or (iv) an empty symbol \emptyset , or (v) an abstract verb FV, HV, or GV. To simplify the table, an 'h' represents either h or $f(h)$, and a 'k' represents either k or $f(k)$. Filled with a particular AV (Abstract Verb), an ACF becomes a PACF, which is an s-AC (simple AC). The c in an s-AC indicates an embedded AC externally placed, and the $f(c)$ refers to a feature (such as 'time', 'place', 'manner', 'agreement') of a co-composing, externally placed AC, identified as c and marked by '=c'. The external AC marked by '=c' co-composes with an s-AC containing a constant c or $f(c)$ to 'saturate' it. Each PACF or s-AC has a verbal type based on its FV, HV, or GV and also a nominal type based on its 0, 1, or 2 'unsaturated' constants. The three verbal types, FAC, HAC, and GAC, combine with the three nominal types, SAC, RAC, and WAC, to form nine possible composite types for an ACF or s-AC. ACF (27) combines with an FAC-SAC to form a topic-comment construction.

Table 2: Computation for v-types

	FAC	HAC	GAC
FAC	(i) FAC	(ii) FAC	(iii) (a) FAC, if GAC is an SAC or RAC (b) GAC, if GAC is a WAC
HAC		(iv) HAC	(v) (a) HAC, if GAC is an SAC or RAC (b) GAC, if GAC is a WAC
GAC			(vi) GAC

(Column vt composes with row vt into cell vt.)

Table 3: Computation for n-types

	SAC	RAC	WAC
SAC	(i) SAC	(ii) SAC	(iii) RAC
RAC		(iv) RAC	(v) RAC
WAC			(vi) WAC

(Column nt composes with row nt into cell nt.)

Table 4: Computing the Primacy Degrees in v-types

	FAC	HAC	GAC
FAC	(i) <2,2>	(ii) <2,0>	(iii) <2,0>
HAC	(iv) <0,2>	(v) <2,2>	(vi) <2,0>
GAC	(vii) <0,2>	(viii) <0,2>	(ix) <2,2>

(Column vt and row vt are assigned weights <p, q> in the cell.)

Table 5: Computing the Primacy Degrees in n-types

	SAC	RAC	WAC
SAC	(i) <1,1>	(ii) <0,1>	(iii) <0,1>
RAC	(iv) <1,0>	(v) <1,1>	(vi) <0,1>
WAC	(vii) <1,0>	(viii) <1,0>	(ix) <1,1>

(Column nt and row nt are assigned weights <r, s> in the cell.)

($m=p+r$, $n=q+s$ are the combined weights for two compared ACs. If $m>n$, then m is p and n is s , and if $m=n$, other criteria are invoked to assign p and s .)

References

- Chang, Yu-Ling. 1998. Beyond constituent structures: semantic structure representations. *Proceedings of the 9th North American Conference on Chinese Linguistics (NACCL-9)*, Vol. 1: *Syntax, Semantics and Discourse*, ed. by Hua Lin, 35-46. Los Angeles: GSIL.
- Chen, Jia-Yin. 2004. Teaching languages effectively in a multilingual classroom with Global Grammar. Paper presented at the 8th University of Hawai'i at Manoa College of Languages, Linguistics, and Literature Conference. Honolulu, Hawai'i, USA.
- Cheng, Ken. 2004. Global Grammar for teaching Mandarin syntax in a multilingual adult classroom. Paper presented at the 8th University of Hawai'i at Manoa College of Languages, Linguistics, and Literature Conference. Honolulu, Hawai'i, USA.
- Chomsky, Noam. 2001. Derivation by phase. *Ken Hale: A Life in Language*, ed. by Michael Kenstowicz, 1-52. Cambridge: MIT Press.
- Goldberg, Adele E. 2006. *Constructions at Work: The Nature of Generalization in Language*. Oxford & New York: Oxford University Press.
- Goldberg, Adele E., and Ray Jackendoff. 2004. The English resultative as a family of constructions. *Language* 80.3:532-568.
- Gammon, Roderic. 1999. Ontological roots of syntax. *Proceedings of the 10th North American Conference on Chinese Linguistics (NACCL-10)*, Vol. 1: *Phonetics and Phonology, Formal Syntax*, ed. by Chaofen Sun, 134-150. Los Angeles: GSIL.
- Hayden, Jeffrey J. 1997. *Fangshu, fangpi, fangfengzheng: a gestalt and its images*. Paper presented at the 9th North American Conference on Chinese Linguistics (NACCL-9). Victoria: University of Victoria.
- Heine, Bernd. 1993. *Auxiliaries: Cognitive Forces and Grammaticalization*. New York: Oxford University Press.
- Holland, John H. 1999. *Emergence: From Chaos to Order*. New York: Basic Books.
- Hsieh, Hsin-I. 1998. Compositional cognitive grammar: an outline. *Studies in Chinese Linguistics* 2:1-24. (In Chinese)
- Hsieh, Hsin-I. 2000. The subject chain in a compositional cognitive grammar. *Language and Linguistics* 1.1:61-88.
- Hsieh, Hsin-I. 2004. Toward a global cognitive grammar. *Proceedings of the 9th International Symposium on Chinese Languages and Linguistics (IsCLL-9)*, ed. by Shuanfan Huang, 173-191. Taipei: National Taiwan University.
- Hsieh, Hsin-I. 2005. Toward a Global Grammar of Chinese. *POLA Forever: Festschrift in Honor of Professor William S-Y. Wang on His 70th Birthday*, ed. by Dah-an Ho & Ovid J. L. Tzeng, 1-17. Taipei: Institute of Linguistics, Academia Sinica.
- Huang, C.-T. James. 1992. Complex predicates in control. *Control and Grammar*, ed.

- by Richard K. Larson et al., 109-147. Dordrecht & Boston: Kluwer.
- Li, Ying-che. 1980. The historical development of the coverb and coverbial phrase in Chinese. *Journal of Chinese Linguistics* 8.2:273-293.
- McCawley, James D. 1971. Where do noun phrases come from? *Semantics: An Interdisciplinary Reader in Philosophy, Linguistics and Psychology*, ed. by Danny D. Steinberg & Leon A. Jakobovits, 217-231. Cambridge: Cambridge University Press.
- Parinyavottichai, Chanyaporn. 2006. A pedagogical application of Global Grammar to locative and directional phrases in Chinese and Thai. Paper presented at the 10th Annual Student Conference of the College of Language, Linguistics, and Literature. Honolulu: University of Hawai'i.
- Sag, Ivan A., Thomas Wasow, and Emily M. Bender. 2003. *Syntactic Theory: A Formal Introduction* (2nd edition). Stanford: CSLI.
- Tai, James H-Y. 1985. Temporal sequence and Chinese word order. *Iconicity in Syntax*, ed. by John Haiman, 49-72. Amsterdam & Philadelphia: John Benjamins.
- Wang, Haidan. 1998. Prototype, exemplars, and images. *Proceedings of the 9th North American Conference on Chinese Linguistics (NACCL-9)*, Vol. 2: *Phonology and Other Papers*, ed. by Hua Lin, 251-267. Los Angeles: GSIL.

[Received 16 December 2006; revised 20 September 2007; accepted 1 November 2007]

Department of East Asian Languages and Literatures
University of Hawai'i
Honolulu, HI 96822
USA
hhsieh@hawaii.edu

組合是建構的根源

謝信一

夏威夷大學

一個句子的組合體與建構體可以雙向轉換。我們根據組合認知語法理論，把組合體擬定為一個語義描繪，經過有規則的修剪手續，把它簡化成一個建構體。

關鍵詞：組合，建構，互動，組合認知語法理論，修剪規則