

Towards a Syntactic Structural Analysis and an Augmented Transition Explanation: A Comparative Study of the Globally Ambiguous Sentences and Garden Path Sentences

Ping-Fang Yu

School of Liberal Arts, Ludong University, Yantai, Shandong Province, China

Email: yupingfang68@126.com

Jia-Li Du

School of Foreign Languages/School of Literature, Ludong University, Yantai, Shandong Province

/Communication University of China, Beijing, China

Email: dujiali68@126.com

Abstract—Globally ambiguous sentences can be understood in more than one way and any decoding involved can be understandable. Garden Path (GP) sentences possess a special structural feature with a semantic trigger, which calls for a much heavier cognition burden since only one decoding is reasonable. The present paper explores the data structures of globally ambiguous sentences and GP ones. The augmented transition networks (ATNs) are used to deepen the analyses and comparison of globally ambiguous sentences and GP ones. We come to the conclusion that GP sentences, syntactically and cognitively different from globally ambiguous ones, have more complex structures.

Index Terms -- natural language processing, data structure, augmented transition networks, garden path sentences, ambiguous sentences

I. INTRODUCTION

Globally ambiguous phenomenon is different from partially ambiguous one to which the Garden Path (GP) phenomenon belongs. Any decoding of globally ambiguous sentences can be accepted by readers while the decoding of GP sentences is involved in the processing breakdown.

The GP sentences occurring at the different levels of human languages have attracted the attention of language researchers, who study the GP sentences from the perspective of grammar [1, 2], semantics [3, 4], pragmatics [5, 6], psychology, information processing [7, 8, 9, 10], and so on. The present paper, on the basis of cognitive linguistics and with the help of the technology of information processing, argues that the key point in decoding the GP sentences is that there occurs a cognitive enlightenment point when the decoding process is stopped, and a kind of structure rearrangement is formed to make the decoding process continue to work smoothly. This kind of semantic tracing back can be explored by way of constructing an appropriate data structure. A data structure is a set of different relations among the various

elements in a system, which can roughly be categorized as the set structure, the corresponding linear structure, the multi-matching parallel schema and one-to-many tree diagram. The four afore-cited data structures approximately correspond to the explanation of the pre-grammatical word set, the linear collocation, the many-to-many ambiguity and the semantic tracing back. We'll then explore the different forms of data structures when decoding the GP sentences and probe into the varieties of linguistic structures, with GP sentences included. Besides, a thorough investigation is conducted to explore the grammatical and cognitive difference between the GP sentences and the globally ambiguous ones, aiming to show the structural features of the GP sentences.

II. THE DATA STRUCTURE-BASED ANALYSES OF THE NON-GARDEN-PATH SENTENCES: A COMPARATIVE PERSPECTIVE

Regarding the data structures of the non-GP sentences, there mainly involve these structures, viz. the pre-grammatical word set structure, the grammatical linear structure and the ambiguous schema one.

The pre-grammatical structure means that the words in a set are randomly piled together, without grammatical requirements or semantic compatibility, which is a kind of "word salad" from the psychologists studying how human beings produce languages. Figure 1 shows how the word set structure is like.

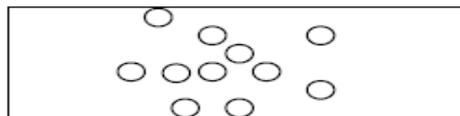


Figure 1. The pre-grammatical word set structure

The next three sentences are the typical examples showing the pre-grammatical word set structure.

Sentence 1: * The new singers the song.

Sentence 2: *The old women the boat.

Sentence 3: *The building window the sun.

This kind of word sets is not created together according to the conventionally accepted grammar rules of the target language, so the whole linear construction is ungrammatical, showing a feature of being completely unrelated both grammatically, semantically or pragmatically. For example, “[The new singers] NP + [the song] NP” in Sentence 1, “[The old women] NP + [the boat] NP” in Sentence 2, and “[The building window] NP + [the sun] NP” in Sentence 3 all have the grammatically disconnected linguistic units piled together, none of them showing a shade of semantic compatibility and grammatical acceptance and making no senses at all.

Unlike the random assemblies of the words in the word set structure, the combination of the words governed by the grammatical linear structure is grammatically accepted, and there does exist a one-to-one matching pattern between the words and the conventional subject-predicate structure. The following is the figure concerning the structure of the non-GP sentences of this kind.

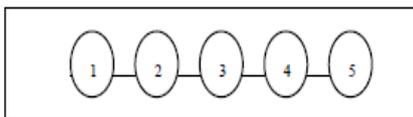


Figure 2. The grammatical linear structure of non-GP Sentences

In the process of syntactic analyses, the elements in the linear structure must be arranged in a sequentially unique way, which means that one and the same element must be syntactically accepted; thus the order of the sentence decoding of the above-mentioned kind has to be processed one by one in order of “1-2-3-4-5”. The followings are the sentences of this kind of data structures.

Sentence 4: The new singers record the song.

Sentence 5: The old women sail the boat.

Sentence 6: The building window reflects the sun.

“[The new singers] NP + [record]V + [the song]NP” in Sentence 4, “[The old women] NP + [sail]V + [the boat] NP in Sentence 5, and “[The building window] NP + [reflects]V + [the sun]NP” in Sentence 6 are all functioning properly in the Subject-Verb-Object constructions of the sentences. All the words and the phrases in the sentences have their unique syntactic functions and show a striking feature of a strict combination sequence. This kind of the sentence structure is conventionally regarded as the basic sentence pattern in human languages, which shows no sign of GP phenomenon in our language processing.

Contrary to the grammatical linear structure, the elements in an ambiguous schema structure show a many-to-many grammatical relation, which means that whatever semantic-grammatical corresponding relations we have chosen, a plausible meaning decoding is available. The data structures of the above-mentioned sentences are as follows.

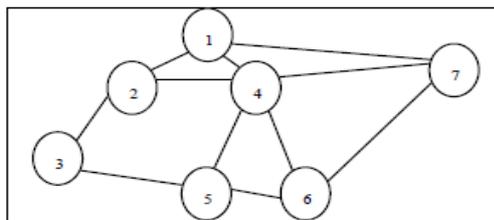


Figure 3. The data structure of ambiguous schema

The cognitive nodes in Figure 3 can be construed in some different ways, which means that no matter what directions the node word moves to, some plausible yet successful textual decoding is available, and consequently various meaning explanations occur. Let’s take an example of Sentence 7.

Sentence 7: Failing student looked hard.

There are a number of explanations when we try to decode Sentence 7, and the main reasons of the multiple understandings lie in the polysemous meanings contained in the two key lexemes “falling” and “hard”. Grammatically speaking, the word “falling” can be categorized either as an adjective or a gerund; and the word “hard” also has a double grammatical categorization both as an adjective and an adverb. Therefore, the two words with dynamic word classes have potentially provided four different explanations, which are demonstrated separately by way of Sentence 8 to Sentence 11.

Sentence 8: Failing (adj) student looked hard (adj).

Sentence 9: Failing (adj) student looked hard (adv).

Sentence 10: Failing (Grd) student looked hard (adj).

Sentence 11: Failing (Grd) student looked hard (adv).

Sentences 8-11 are to be grammatically analyzed in the following ways.

- G = {Vn, Vt, S, P}
- Vn = {S, NP, VP, Adj, Grd, V, N, Adv}
- Vt = {failing, student, looked, hard}
- S = S
- P:
- S → NP VP (a)
- NP → Adj N (b)
- NP → Grd N (c)
- VP → V Adj (d)
- VP → V Adv (e)
- Adj → {failing, hard} (f)
- Grd → {failing} (g)
- N → {student} (h)
- V → {looked} (i)
- Adv → {hard} (j)

As shown in the preceding rules, the NP (b) in Line 7 corresponds to the grammatical status set by the word “falling” in Example 8 and Example 9. According to the conventionally accepted grammatical rules, a noun and a verb form a noun phrase; therefore the word “falling” with a word class of adjective can be combined with “student”; thus a noun phrase comes into being. And when the noun phrase co-occurs with the word “hard”, which can be tagged either as an adjective or an adverb, two possible decoding procedures, viz. “f-h-b-i-f-d-a” in Sentence 8 and “f-h-b-i-j-e-a” in Sentence 9 are both grammatically and semantically plausible. The two

different decoding procedures result from the selection of the word classes of “hard”. To paraphrase the two sentences, Sentence 8 means that “the students having been failed is suffering a lot” and Sentence 9 can be paraphrased as “the student having been failed seems to be working hard”. Table I in the following part shows the potential multi-combinations of falling (adj) and “hard (adv)”.

TABLE I.
THE POTENTIAL MULTI-COMBINATION OF “FAILING” (ADJ) AND “HARD”(ADV)

1. Failing student looked hard			
2. Adj student looked hard		(f)	
3. Adj N looked hard		(h)	
4. NP looked hard		(b)	
5. NP V hard		(i)	
6.	NP V Adj (f)	6'	NP V Adv (j)
7.	NP VP (d)	7'	NP VP (e)
8.	S (a)	8'	S (a)

The NP(c) in Line 8 corresponds to the grammatical status set by the word “falling” in Sentence 10 and Sentence 11. It’s universally acknowledged that a gerund can be combined with a noun, forming a noun phrase. Therefore the word “falling”, tagged as a gerund, can be collocated with the word “hard”, which is tagged both as an adjective and an adverb. Then we’ll have two sentences with different decoding procedures. To be more specific, the decoding procedure of Sentence 10 is “g-h-c-i-f-d-a”, which is paraphrased as “it’s no easy job to fail the student”. And the decoding procedure of Sentence 11 is “g-h-c-i-j-e-a”, which means that “someone is working hard with the aim of failing the student”.

TABLE II.
THE POTENTIAL COMBINATION OF “FAILING” (GRD) AND “HARD”(ADV)

1. Failing student looked hard			
2. Grd student looked hard		(g)	
3. Grd N looked hard		(h)	
4. NP looked hard		(c)	
5. NP V hard		(i)	
6.	NP V Adj (f)	6'	NP V Adv (j)
7.	NP VP (d)	7'	NP VP (e)
8.	S (a)	8'	S (a)

According to the above-mentioned analyses, we know that grammatically the word “falling” shows a double word classes just as Line 11(f) and Line 12(g) have shown to us. Therefore, the generation of the NP, governed by the rules of Line 7(b) and Line 8(c), is ambiguous. Similarly, the word “hard” also has a double word classes shown in Line 11(f) and Line 15(j), which makes the generation of the VP work according to either the rule of Line 9 (d) or that of Line 10 (e). The dual ambiguities of the two words in word classes bring in four different ways of semantic decoding when they are collocated, which is shown in Table III.

TABLE III.
THE POTENTIAL MULTIPLE MATCHES OF “FAILING” AND “HARD” OWING TO THEIR POSSIBLE WORD CLASSES.

1. Failing student looked hard							
2. Adj student looked hard (f)				2' Grd student looked hard (g)			
3. Adj N looked hard (h)				3' Grd N looked hard (h)			
4. NP looked hard (b)				4' NP looked hard (c)			
5. NP V hard (i)				5' NP V hard (i)			
6.	NP V Adj (f)	6'	NP V Adv (j)	6	NP V Adj (f)	6'	NP V Adv (j)
7.	NP VP (d)	7'	NP VP (e)	7	NP VP (d)	7'	NP VP (e)
8.	S1 (a)	8'	S2 (a)	8	S3 (a)	8'	S4 (a)

It’s clearly shown by the preceding Table III that the multiplicity of the lexical-grammatical status of the two collocates entails the non-uniqueness of the sentences generated according to accepted grammatical rules. This kind of ambiguity is to be shown with the help of a many-to-many schema structure, which is demonstrated in Figure 4.

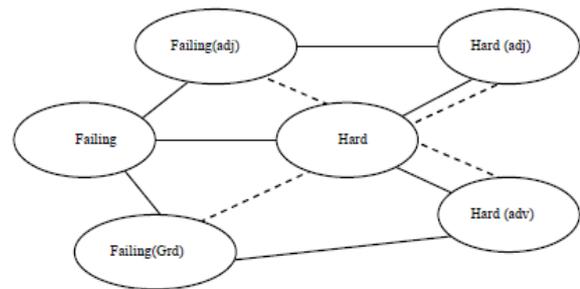


Figure 4. The data structure of the sentence “Failing student looked hard”

III. ANALYSES OF THE DATA STRUCTURE OF GARDEN-PATH SENTENCES

The non-GP sentences governed by either the word set structure, the grammatical linear structure or the ambiguous schema structure have a defining characteristic of being mono-directional. Unlike the non-Garden-Path ones, the GP sentences show a distinguishing and recurring feature of being forced to turn back to the original starting point to find a new way out in the decoding process, which turns out to be a tree diagram regarding its data structure. The tree diagrams of the GP sentences mean that there does exist a one-to-many corresponding relation among the elements in the structure under discussion. When one of the relations is by default interpreted as a cognitive prototype, it will automatically become the root of the whole decoding process; then extend along the seemingly plausible decoding route until a sudden semantic breakdown calls for a prompt return to the starting point. Afterwards, it will move along a next node, until all the elements in the structure go smoothly in the decoding process. The structure of sentences of this kind is shown in Figure 5.

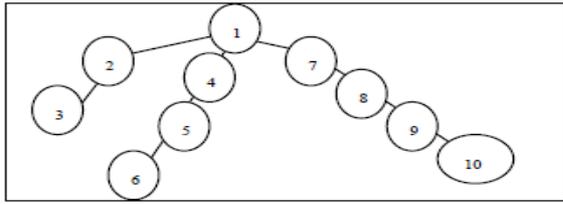


Figure 5. The tree diagram of the Garden-Path sentences

Suppose the root element in the above-mentioned figure corresponds to three different ways of semantic cognition, and the appropriate way of the decoding of the target sentence is No.10; then we'll have the possible order of decoding as "1-2-3-3-2-1-4-5-6-6-5-4-1-7-8-9-10". It can be easily found that a kind of semantic returning is necessarily entailed in the decoding of the GP sentence. The simultaneous appearances of the two decoding routes viz. "3-2-1" and "6-5-4-1" demonstrate that minimally two tracing-backs are included in the process of sentence understanding, which demonstrates that the double negation turns out to be an indispensable part of the sentence decoding and it can be statistically verifiable. We'll have another example.

Sentence 12: The new record the song.

Since the word "record" can be tagged either as a noun or a verb, the proper understanding of sentence 12 can be achieved by way of the context-free grammar, which is demonstrated in the following part.

G={Vn, Vt, S, P}
 Vn={S, NP, VP, Det, Adj, V, N}
 Vt={the, new, record, song}
 S=S
 P:

- S → NP VP (a)
- NP → Det Adj (b)
- NP → Det N (c)
- NP → Det Adj N (d)
- VP → V NP (e)
- Det → {the} (f)
- Adj → {new} (g)
- N → {record, song} (h)
- V → {record} (i)

Processing Procedure (Bottom-Up)

- 1: The new record the song
- 2: Det new record the song (f)
- 3: Det Adj record the song (g)
- 4: Det Adj N the song (h)
- 5: NP the song (d)
- 6: NP Det song (b)
- 7: NP Det N (c)
- 8: NP NP (e)
- 9: Det Adj record the song (g)
- 10: NP record the song (a)
- 11: NP V the song (i)
- 12: NP V Det song (f)
- 13: NP V Det N (d)
- 14: NP V NP (c)
- 15: NP VP (e)
- 16: S (a)
- SUCCESS

Judged by the afore-cited theoretical framework of context-free grammar, it's clear that the decoding of Sentence 12 involves a kind of temporary halting and the sequent on-time returning because of a sudden semantic breakdown. According to the tree diagram of Figure 6, the correct sequence of the decoding process is as follows: 1-2(f)-3(g)-4(h)-5(d)-6(f)-7(h)-8(c)-8(c)-7(h)-6(f)-5(d)-4(h)-3 (g)-9(g)-10(b)-11(i)-12(f)-13(h)-14(c)-15(e)-16(a). Owing to the fact that Rules 3(g) - 9(g) are concerned with the decisions that whether the node word "record" should be tagged as a noun or as a verb, the word "record" thus becomes the source of the semantic tracing-back, and brings in two explanations of the target

sentences. One decoding procedure is "3(g)-4(h)-5(d)-6(f)-7(h)-8(c)" with the word "record" being a noun; and the other decoding is "9(g)-10(b) -11(i) -12(f) -13(h) -14(c) -15(e) -16(a)" with the word "record" being tagged as a verb. However, the above-mentioned sentence in which "record" is regarded as a noun is not grammatically accepted, and then a new cognition enlightenment point appears: the decoding process then begins to trace back from 8(C) to 3(g), and a kind of GP phenomenon "8 (c) - 7 (h) - 6 (f) - 5 (d) - 4 (h) -3 (g)" comes into being. To be more specific, the word "record" in the sentence has a dual role: grammatically, the word "record" is always tagged as a noun by default reasoning; while functionally the word "record" must be tagged as a verb to make the whole decoding process go smoothly. And the first dead end in the information processing leads to the second semantic tracing back. The tree diagram of this decoding process of Sentence 12 is shown below:

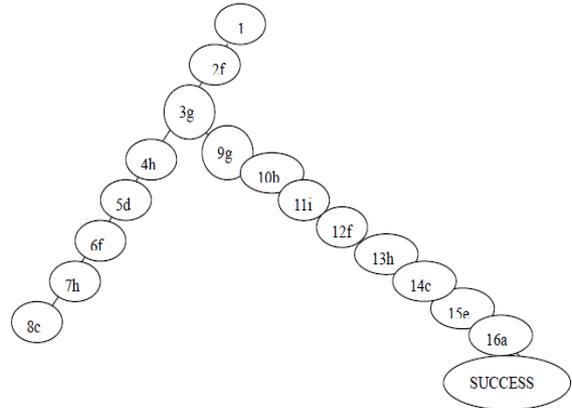


Figure 6. The data structure of "The new record the song"

IV. THE APPLICATION OF AUGMENTED TRANSITION NETWORKS

Augmented Transition Networks (hereafter abbreviated to ATNs) are applied in the computational linguistics to demonstrate the process of the state transitions in NLP[11, 12, 13]. The networks have a main net and some subsidiary ones, all of which are interconnected and interchangeable, allowing the linguistic segments to semantically and syntactically match with other collocates in the main net until they are successfully decoded. ATNs clearly show the hierarchical features of the decoding processes of the sentences with complicated syntactic relations. On basis of the ATNs theory, the following part of the paper endeavors to analyze the decoding processes of the ambiguous sentences and the GP sentences, by which a comparison is made between the two kinds of sentences.

The ambiguous sentences, because of their phonetically, semantically or grammatically various explanations, potentially convey at least two different meanings. Therefore the decoding of the ambiguous sentences necessarily involves a kind of semantic selection. Metaphorically speaking, the decoding of the ambiguous sentences, necessarily involving some semantic options, is like a road with several motorways,

and the motorways are parallel with each other and each of them can take you to the place you want.

Unlike the one-road-with-several-driveway ambiguous sentences, the GP sentences are actually a one-way street. Metaphorically speaking, exploring GP sentences is the same as the fact that someone who is strolling along a zigzagging garden path suddenly finds that the path is a dead end and what he or she can do is to turn around and walk along the same route to the original location just to find another way out. Unlike the multi-explanations of the ambiguous sentences, the GP sentences are to be decoded only in one and the same way. That is to say, in the cognitive decoding process of GP sentences, there will be a semantic stoppage and a sequent backtracking to make the decoding go continuously. This kind of sudden insight will certainly burden and lengthen the decoding process, while it simultaneously develops the decoding capability in a leaping way, which has been empirically tested by scholars from different perspectives [14, 15, 16, 17, 18].

The comparison between ambiguous sentences and the GP sentences has been conducted by the scholars from the perspectives of syntax [19], pragmatics [20], psychology [21], computational science [22] and cognitive science [23, 24, 25, 26]. The present paper, on basis of Kempen's ATN theory, endeavors to compare the ambiguous sentences and GP Phenomena with the aim of exploring the semantic trigger systems contained respectively in ambiguity and GP sentences.

V. THE FRAMEWORK OF KEMPEN'S SYNTACTIC PROCESSORS

In 1996, G. Kempen published his famous article titled *Computational models of syntactic processing in human language comprehension*. In this article, Kempen discusses the basic framework of syntactic processing machine as well as explores both the ambiguous sentences and GP sentences from the perspective of ATNs.

Kempen argues that the framework of a syntactic processor minimally contains five parts, viz. an input buffer, a syntactic processor and working memory, lexicon and grammar, the conceptual knowledge and the syntactic structure. Kempen's model is shown in Figure 7.

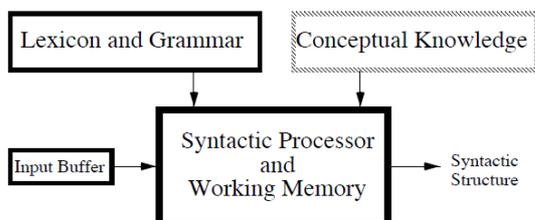


Figure 7. Kempen's (1996) model of syntactic processing procedures

Figure 7 shows that the word strings are firstly stored in the input buffer, and the lexical, morphological, syntactical and conceptual information is fully integrated in the syntactic processor, forming either a complete or a partial syntactic structure. Kempen holds that the detailed

explanation in the information processing is varied according to the corresponding models, which thus forms an interactive model. In the syntactic processing of human languages, grammar, lexicon and most of the conceptual knowledge are stored in our long-time memory, which has laid a solid foundation of the application of ATNs in language processing.

ATN was first put forward by Kaplan (1972) and was mainly used to demonstrate the language processing procedures in a series of states. The different states in language processing are demonstrated with labelled nodes and the transition is shown with directed and labelled arcs. The arcs, however, are further labelled with some particular syntactic rules, and the transition direction is shown with an arrow. Therefore the whole decoding process is clearly explained. With the help of the demonstration of the states involved in the decoding process, together with the ATNs, Kempen probes into the discrepancies between the ambiguous sentences and the GP sentences. See the following three examples.

Sentence 13: The student read the letter to Chrysanne.

Sentence 14: The student read the letter to Chrysanne fainted.

Sentence 15: The student who was read the letter to Chrysanne fainted.

The three sentences cited above are both semantically and syntactically different. Sentence 13 can be paraphrased as "a student read a letter to a woman named Chrysanne". This kind of decoding is cognitively economical, for there is less structural nodes. Besides, Sentence 13 can also be paraphrased as "a student read a letter, but the letter was meant for Chrysanne". The latter decoding is also syntactically correct. It's safe to say that Sentence 13 is syntactically ambiguous.

Sentence 14 is a typical GP sentence. Before the occurrence of the past participle "fainted", the syntactic pattern of "[[The student] NP [[read] V_{transitive} [the letter]NP[to Chrysanne]PP]VP]S", which accords with the Sausage Machine theory, has come into being. But the participation of the following part has the former syntactic pattern changed, and forms a new pattern of "[[The student] NP [[read] V_{passive} [[the letter] NP [to Chrysanne] PP] NP] RRC] NP [fainted] VP] S". This kind of change causes a cognitive insight point and consequently the GP phenomenon occurs.

Sentence 15 is a sentence of commonly used type. The short working memory of the decoder is greatly affected by the word length when decoding the phrase "to Chrysanne", and accords with the Right Association Principle of sentence decoding. Therefore, the syntactic pattern adopted in the process is "[[read] V_{transitive} [[the letter] NP [to Chrysanne] PP] NP] VP" (i.e. "V_{tr}+NP" Pattern), instead of the pattern "[[read] V_{transitive} [the letter] NP [to Chrysanne]PP]VP" (i.e. "V_{tr}+NP + PP" Pattern), and there're no ambiguities or GP phenomenon occurring in Sentence 15. As to the details, see Figure 8 and Figure 9.

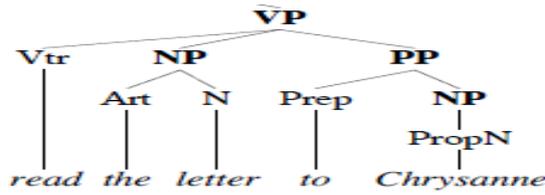


Figure 8. The node pattern of "Vtr+NP+PP"

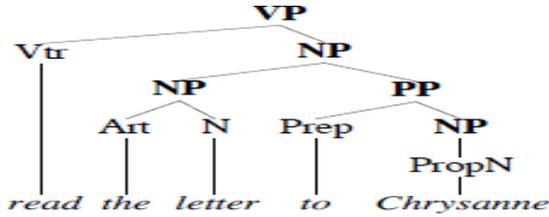


Figure 9. The node pattern of "Vtr+NP"

VI. THE ANALYSES OF THE AUGMENTED TRANSITIONS OF MULTI-STATES IN THE SYNTACTIC PROCESSING

G. Kempen introduces the context-free grammar into his procedure, with the aim of improving the program-based language processing with the help of the ATNs. Figure 10 shows us how the context-free grammar works in Sentence 13 and Sentence 14.

$$G = \{V_n, V_t, S, P\}$$

$$V_n = \{S, NP, RRC, VP, PP, V_{intr}, V_{tr}, V_{pass}, Art, N, PropN, Prep\}$$

$$V_t = \{the, student, read, letter, to, Chrysanne, fainted\}$$

$$S = S$$

P:

- | | | | |
|----------------------------|-----------------------------------|------------------------------------|-----------------------------------|
| 1. $S \rightarrow NP VP$ | 6. $RRC \rightarrow V_{pass} NP$ | 11. $PP \rightarrow Prep NP$ | 16. $N \rightarrow letter$ |
| 2. $NP \rightarrow Art N$ | 7. $VP \rightarrow V_{intr}$ | 12. $V_{intr} \rightarrow fainted$ | 17. $N \rightarrow student$ |
| 3. $NP \rightarrow PropN$ | 8. $VP \rightarrow V_{tr} PP$ | 13. $V_{tr} \rightarrow read$ | 18. $PropN \rightarrow Chrysanne$ |
| 4. $NP \rightarrow NP PP$ | 9. $VP \rightarrow V_{tr} NP$ | 14. $V_{pass} \rightarrow read$ | 19. $Prep \rightarrow to$ |
| 5. $NP \rightarrow NP RRC$ | 10. $VP \rightarrow V_{tr} NP PP$ | 15. $Art \rightarrow the$ | |

/* RRC(reduced relative clause) */

Figure 10. The CFG used in the decoding process of Sentences 13 & 14

A. The State Demonstration of the Main Networks

The procedural decoding of the ATNs of Sentence 13 and Sentence 14 can be explained with the help of the following 5 figures, viz. from Figure 11 to Figure 15. Figure 11 shows us what the states of the output of the main net S are like.



Figure 11. The states involved in the output of the main net of "S"

Figure 11 is comprised of four parts, viz., the labelled nodes, the labelling arcs, the denoting lines and the

syntactic rules concerned. Labelled nodes "S₀", "S₁" and "S₂" respectively stand for the initial state, the middle state and the output one. The labelling arcs and the denoting lines in Figure 11 converge into the 3 right-directing lines, demonstrating the transition directions of the decoding. The syntactic rules "(SEEK) NP", "(SEEK) VP" and "(SEND) S" are used to denote the grammatical functions. It's clear that Figure 11 shows us the syntactic analyses of "S → NP VP".

B. The State Demonstration of the VP Subsidiary Net

Figure 12 in the following part shows us the demonstration of the VP subsidiary net.

In Figure 12, the labelling nodes "VP₀", "VP₁", "VP₂" and "VP₃" respectively stand for the different states of the decoding process. There are three labelling arcs and three denoting lines. Besides, six syntactic rules, viz. "CAT V_{tr}", "SEEK NP", "CAT V_{intr}", "SEEK PP", "JUMP", "SEND VP" are involved in the process. In terms of the syntactic rules, "CAT V_{tr}" means the node word is a transitive verb; "SEEK NP" means that a transitive verb needs a noun phrase, and the corresponding syntactic rule is "VP → V_{tr} NP". "CAT V_{intr}" means an intransitive verb is involved, and its syntactic rule is "VP → V_{intr}". "SEEK PP" means that a new verbal phrase is to occur according to the rule "VP → V_{tr} NP PP" or "VP → V_{intr} PP". "JUMP" is used to show that the newly-formed verbal phrase can directly pass by the matching with a preposition phrase and leap to the output stage. "SEND VP" tells us that the ultimate output is a verbal phrase. Theoretically the "VP₃" here can have 4 results. The first one is "VP → V_{tr} NP PP" (as the case in "[[read] V_{tr} [the letter] NP [to Chrysanne] PP]VP"), and its state curve is "4-5-7-9"; The second result is "VP → V_{intr} PP" (as the case in "[[read] V_{intr} [to Chrysanne] PP]VP", and the state curve is "6-7-9". The third one is "VP → V_{tr} NP" (as the case in "[[read] V_{tr} [[the letter] NP [to Chrysanne] PP]NP]VP") with a state curve "4-5-8-9". And the last result is "VP → V_{intr}" (as the case in "[[fainted] V_{intr}]VP"), with a state curve "6-8-9".

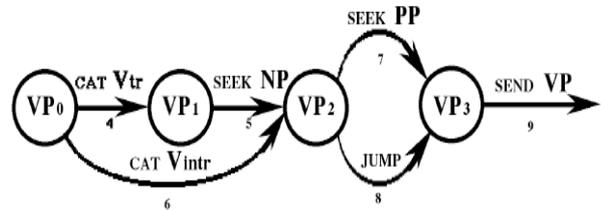


Figure 12. The state demonstration of the output subsidiary net of "VP"

The first and the third models respectively correspond with the state transition model of the two different translations of the verbal phrase "read a letter to Chrysanne", containing two different state curves "4-5-7-9" and "4-5-8-9", verifying that the ambiguous sentences don't cause backtrackings in the decoding process. The fourth result corresponds with the transition states of "VP → V_{intr}" in Sentence 14, while the

second result can't be applied in the analyses of Sentence 13 and Sentence 14. The details are shown in Table IV.

TABLE IV.
THE THEORETICAL MODEL OF GENERATING STATES OF THE
SUBSIDIARY NET OF "VP3"

No.	Program	Trace	Example	Relevance
1	VP → V _{tr} NP PP	4-5-7-9	[[read] V _{tr} [the letter] NP [to Chrysanne] PP]V	Yes
2	VP → V _{intr} PP	6-7-9	[[read] V _{intr} [to Chrysanne] PP]VP	No
3	VP → V _{tr} NP	4-5-8-9	[[read] V _{tr} [[the letter] NP [to Chrysanne] PP]NP]VP	Yes
4	VP → V _{intr}	6-8-9	[[fainted] V _{intr}]VP	Yes

C. The State Demonstration of the NP Subsidiary Net

The figure below tells us how the state demonstration of the output subsidiary NP net works.

The labelling nodes "NP₀", "NP₁" and "NP₂" demonstrate the different decoding stages. There are three labelling arcs "10", "14" and "15", together with three denoting lines "11", "12" and "13". Besides, there involve six syntactic rules, viz. "CAT PropN", "CAT Art", "CAT N", "SEND NP", "SEEK PP" and "SEEK RRC".

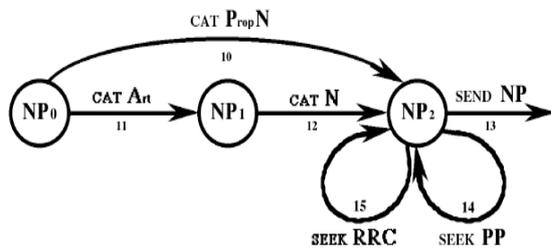


Figure 13. The state demonstration of the output subsidiary net of "NP"

The code "CAT PropN" means that the grammatical category of the node word is a pronoun. According to the generating rule "NP → Prop N", the state curve of the pronoun is "10-13", viz. "[[Chrysanne] PropN] NP". "CAT Art" and "CAT N" stand for "article" and "noun" respectively. According to the rule "NP → Art N", a new noun phrase comes into being, with a state curve of "11-12". Code "SEEK PP" means that a prepositional phrase is to be searched to have a grammatical match. When NP2 comes into being, a new noun phrase can be produced. Theoretically, we'll have two results here. The first is the syntactic pattern of "[[[Chrysanne] Prop N] NP [on the park] PP]NP" with a state curve "10-14-13" according to the syntactic rule "NP → Prop N", "PP → Prep NP" and "NP → NP PP". The second one is a syntactic pattern "[[the letter]NP[to Chrysanne]PP]NP" with a state curve of "11-12-14-13" according to the rules "NP → Art N", "PP → Prep NP" and "NP → NP PP". Practically, only the second pattern is applicable in

the decoding process of Sentence 13 and 14, and the first pattern can't work because there's no prop noun as the subject in the two sentences.

"SEEK RRC" means a simplified sentence, i.e. reduced relative clause (RRC), is needed to match a new phrase. Theoretically speaking, four syntactic states can be produced. (1) The syntactic rules "NP → PropN", "RRC → V_{pass} NP" and "NP → NP RRC" bring in a pattern "[[[Mike] PropN] NP [[read] V_{pass} [the letter] NP] RRC] NP fainted" with a state curve "10-15-13". (2) The syntactic rules "NP → Prop N", "RRC → V_{pass} NP", "NP → NP RRC" and "NP → NP PP" form a syntactic pattern "[[[Mike] PropN] NP[[read] V_{pass} [[the letter] NP [to Chrysanne]PP]NP]RRC]NP fainted" with a state curve "10-14-15-13". (3) A new syntactic pattern "[[The student]NP[[read] V_{pass} [[the letter]NP[to Chrysanne]PP]NP]RRC]NP fainted" with a state curve "11-12-14-15-13" is formed according to the rules "NP → Art N", "RRC → V_{pass} NP", "NP → NP RRC" and "NP → NP PP". (4) According to the rules "NP → Art N", "RRC → V_{pass} NP" and "NP → NP RRC", a syntactic model "[[The student]NP [[read] V_{pass} [the letter]NP] RRC]NP fainted" with a state curve "11-12-15-13" comes into being. Among the above-cited syntactic patterns, only the third one can lead to a successful decoding of Sentence 14.

Considering the analyses of the transition states shown in Figure 13, we know that theoretically there are eight forms concerning the output of NP2, while practically only four of them are plausible. See Table V.

TABLE V.
THEORETICAL STATE PATTERN OF THE SUBSIDIARY NET OF "NP2"

No.	Program	Trace	Example	Relevance
1	NP → PropN	10-13	[[Chrysanne] PropN]NP	Yes
2	NP → Art N	11-12-13	[[The]Art[student] N]NP	Yes
3	NP → PropN PP → Prep NP NP → NP PP	10-14-13	[[[Chrysanne]PropN]NP[on the park]PP]NP	No
4	NP → Art N PP → Prep NP NP → NP PP	11-12-14-13	[[the letter]NP[to Chrysanne]PP]NP	Yes
5	NP → PropN RRC → V _{pass} NP NP → NP RRC	10-15-13	[[[Mike] PropN]NP [[read] V _{pass} [the letter]NP]RRC]NP fainted	No
6	NP → PropN RRC → V _{pass} NP NP → NP RRC NP → NP PP	10-14-15-13	[[[Mike] PropN]NP[[read] V _{pass} [[the letter]NP [to Chrysanne]PP]NP]RRC]NP fainted	No
7	NP → Art N RRC → V _{pass} NP NP → NP RRC NP → NP PP	11-12-14-15-13	[[The student]NP[[read] V _{pass} [[the letter]NP[to Chrysanne]PP]NP]RRC]NP fainted	Yes
8	NP → Art N RRC → V _{pass} NP NP → NP RRC	11-12-15-13	[[The student]NP [[read] V _{pass} [the letter]NP] RRC]NP fainted	No

D. The State Demonstration of Other Subsidiary Nets besides the NP and VP

Figure 14 and Figure 15 below are the state demonstration of the output subsidiary PP net and RRC net.

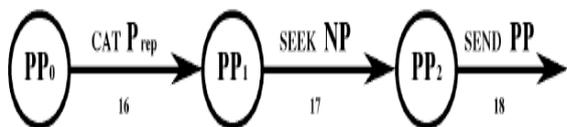


Figure 14. The state demonstration of the output subsidiary net of “PP”

According to the rule “PP → Prep NP” and “NP → Prop N”, the decoding pattern “[to] Prep [[Chrysanne] Prop N] NP] PP” is produced.

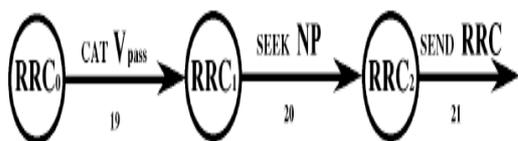


Figure 15. The state demonstration of the output subsidiary net of “RRC”

According to the grammatical rules “RRC → V_{pass} NP”, “NP → Art N” and “NP → NP PP”, a new syntactic pattern “[read] V_{pass} [[[the] Art [letter] N] NP [to Chrysanne] PP] NP] RRC” comes into being.

With the successful decoding of all the subsidiary nets, viz. the VP subsidiary net, the NP net, the PP net, the RRC net, the main net S, the target sentence is to be encoded and decoded thoroughly.

VII. ANALYSES OF THE “PUSH-POP PATTERNS” IN A MULTI-STATE TRANSITION SYSTEM

Involved in the production of Sentence 13 and 14 are three kinds of “push-pop” patterns: the “Vtr+NP+PP” pattern, the “Vtr+NP” pattern and the GP sentence pattern. Among them, the “Vtr+NP+PP” pattern is the most cognitively economical; the “Vtr+NP” pattern is more complex, with a much higher degree of pushing-down. The GPP pattern is the most cognitively complex and has a most complicated transition state. According to the analyses of Figures 11-15, these different patterns can be shown clearly.

A. The “Push-Pop” Pattern in the “Vtr+NP+PP” Construction

The state curve of Sentence 13 is as follows: Start—1(S₀)—11(NP₀)—12(NP₁)—13(NP₂)—1(S₀)—2(S₁)—4(VP₀)—5 (VP₁)—11(NP₀)—12(NP₁)—13(NP₂)—5(VP₁)—7(VP₂)—16(PP₀)—17(PP₁)—18(PP₂)—7(VP₂)—9(VP₃)—2(S₁)—3(S₂)—success. Among the decoding procedures, a number of pushing-downs and popping-ups are involved. The pushing-downs mean the matching pattern from the main net to the subsidiary nets concerned. Conversely, the popping-ups refer to the stage from the subsidiary net to the main one.

The first pushing-down and popping-up is connected with the matching partner of the noun phrase, viz. “Start—1(S₀)—11(NP₀)—12(NP₁)—13(NP₂)—1(S₀)”. When S₀ is transitioned to S₁, a noun phrase is sought to be matched grammatically, and then the main net pushes down to the subsidiary NP net, selecting an appropriate collocate from the eight potential noun phrases. The NP “the student”, in line with the pattern “11(NP₀)—12(NP₁)—13(NP₂)”, is popped up to the main S net.

The second push-pop occurs in the process of the verbal phrase’s matching and the procedure is “2(S₁)—4(VP₀)—5 (VP₁)—11(NP₀)—12(NP₁)—13(NP₂)—5(VP₁)—7(VP₂)—16(PP₀)—17(PP₁)—18(PP₂)—7(VP₂)—9(VP₃)—2(S₁)—3(S₂)”. This kind of push-pop is much more complicated, involving one first-level push-pop, and two second-level push-pops.

The first-level pushing down occurs during the stage of “2(S₁)—4(VP₀)—5 (VP₁)”. The first second-level push-pop comes into being in the process of “11(NP₀)—12(NP₁)—13(NP₂)—5(VP₁)”. And the second second-level push-pop occurs at the stage of “7(VP₂)—16(PP₀)—17(PP₁)—18(PP₂)—7(VP₂)”. The output procedure is in the stage “3(S₂)—success”. According to the syntactic rule “S → NP VP”, the decoding of Sentence 13 is completed. One first-level and two second-level push-pops are involved. Figure 16 shows the “push-pop” pattern of the “Vtr+NP+PP” construction, in which the upward arrow stands for the popping-up, and the downward arrow refers to the pushing-down and the linear arrow means the directions.

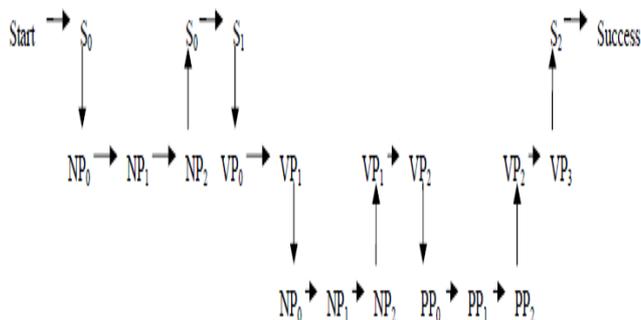


Figure 16. The augmented transition net of “V_{tr}+NP+PP”

B. The “Push-Pop” Pattern in “Vtr+NP”

Another state curve of the decoding of Sentence 13 is as follows: “Start—1(S₀)—11(NP₀)—12(NP₁)—13(NP₂)—1(S₀)—2(S₁)—4(VP₀)—5 (VP₁)—11(NP₀)—12(NP₁)—13(NP₂)—14(PP)—16(PP₀)—17(PP₁)—18(PP₂)—14(PP)—5(VP₁)—8(VP₂)—9(VP₃)—2(S₁)—3(S₂)—Success”. This kind of decoding pattern takes up more cognitive resource than the first pattern, having two first-level push-pops, one second-level push-pop and one third-level push-pop. Because three levels of push-pop are involved, the whole decoding system is more complicated, which empirically verifies the assumption

that the “V_{tr}+NP + PP” pattern takes up much less resource than the “V_{tr}+NP” pattern, which is demonstrated in Figure 17.

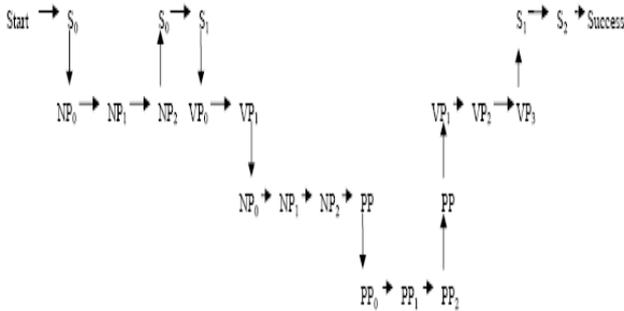


Figure 17. The “push-pop” pattern in the “V_{tr}+NP” construction

We can find that as an ambiguous sentence, Sentence 13 can be decoded by way of two constructions of “V_{tr}+NP+PP” and “V_{tr}+NP”. Provided with the same amount of working memory, the “V_{tr}+NP+PP” construction is preferred because it’s much more cognitively economical. Besides, the two constructions don’t cause a backtracking.

C. The “Push-Pop” Pattern in the Garden Path Sentences

The decoding system of Sentence 14 is much more complicated because a backtracking occurs in the decoding process and two kinds of decoding procedures are involved and coincided, with a much heavier cognitive burden. According to the Minimal Attachment Principle, the most cognitively economical pattern “V_{tr}+NP + PP” is adopted. Sentence 14, however, complies with the Right Association Principle, and the most complicated syntactic pattern “V_{tr}+NP” is chosen. The following 3 figures show the difference of the corresponding tree diagrams.

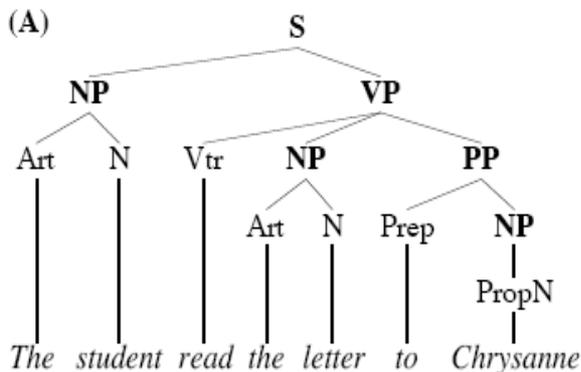


Figure 18. The tree diagram of the ambiguous ” V_{tr}+NP+PP” construction governed by the Minimal Attachment Principle

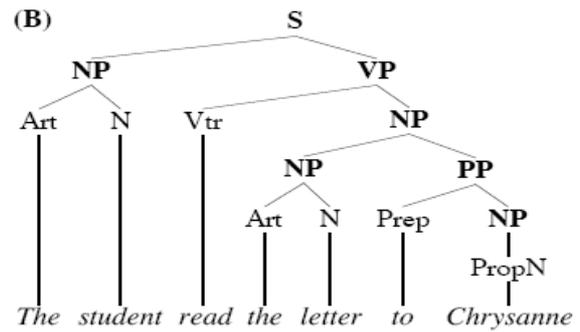


Figure 19. The tree diagram of the ambiguous “V_{tr}+NP” construction governed by the Right Association Principle

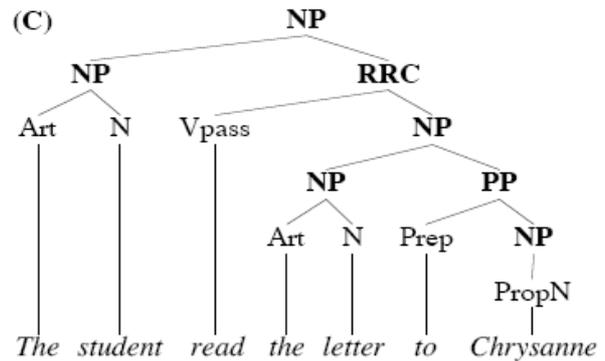


Figure 20. The tree diagram of the Garden Path sentences

The variation of the PP modifiers is the main reason which causes the GP phenomenon. Besides, the change of the predicate verb can also produce GP phenomenon. Before the occurrence of “fainted”, the predicate of Sentence 14 is assumed to be “read” and the corresponding syntactic rule is “VP → V_{tr} NP”. All is changed after the appearance of “fainted”. The word “read”, according to the “RRC → V_{pass} NP” rule, is downgraded as the verb in the clause, and “fainted”, however, is upgraded as the main verb of the whole sentence. The readjustment of the main verb also causes the GP phenomenon.

Sentence 14, being one of the typical GP sentences, entails a modifier variation and a verb change, and its decoding state curve is divided into 4 phases.

The first phase is the transition state before the occurrence of the word “fainted”, something like the “V_{tr}+NP + PP” pattern in line with the Minimal Attachment Principle. The curve is “Start – 1(S₀) – 11(NP₀) – 12(NP₁) – 13(NP₂) – 1(S₀) – 2(S₁) – 4(VP₀) – 5(VP₁) – 11(NP₀) – 12(NP₁) – 13(NP₂) – 5(VP₁) – 7(VP₂) – 16(PP₀) – 17(PP₁) – 18(PP₂) – 7(VP₂) – 9(VP₃) – 2(S₁) – 3(S₂) – backtracking”.

The second phase is from the restart of the decoding to the match of the NP. The state curve of the decoding is “Restart – 1(S₀) – 11(NP₀) – 12(NP₁) – 13(NP₂) – 19(RRC₀) – 20(RRC₁) – 11(NP₀) – 12(NP₁) – 13(NP₂)”.

The third phase is the backtracking after the occurrence of the prepositional phrase. Its decoding state curve is as follows: “20(RRC₁) – 21(RRC₂) – 15(RRC) – 13(NP₂) – 1(S₀) – 2(S₁) – backtracking”. The

backtracking in the third stage is the second reentry, which is represented with dotted lines.

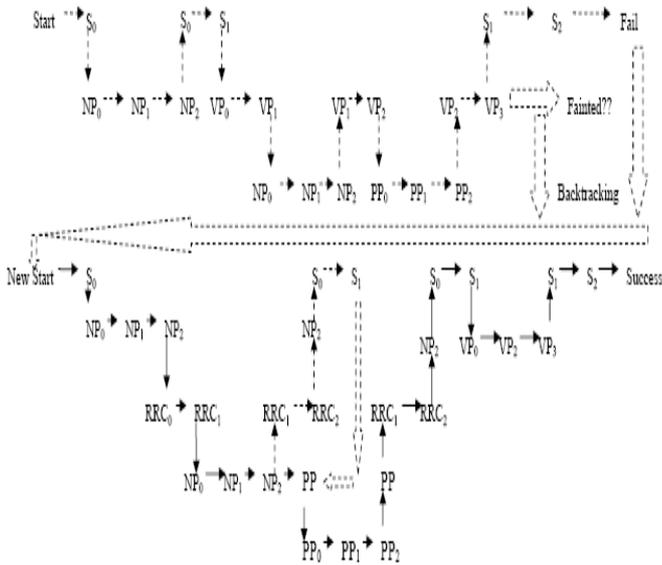


Figure 21. The “push-pop” pattern of Garden Path sentences

The fourth stage is from the end of the backtracking to the success of the decoding. After a second reentry, the grammatical categories of the word “read” and the phrase “to Chrysanne” are determined, which gives a rapid processing of Sentence 14. The state curve of the “push-pop” concerning the stage in discussion is as follows: 14(PP) – 16(PP₀) – 17(PP₁) – 18(PP₂) – 14(PP) – 20(RRC₁) – 21(RRC₂) – 13(NP₂) – 1(S₀) – 2(S₁) – 6(VP₀) – 8(VP₂) – 9(VP₃) – 2(S₁) – 3(S₂) – success. The following figure shows the detailed procedures.

D. Contrasting the Complexity of the Transition States in Line with Varied Patterns

As Figure 21 shows, there are two backtrackings, four levels of pushing down and five hierarchical layers in the decoding process of Sentence 14. The decoding of Sentence 14 is clearly much more complicated than that of Sentence 13. The following table is the contrastive data concerning the 3 kinds of decoding procedures of Sentence 13 and Sentence 14.

As shown in Table VI, the hierarchical levels of the patterns of “Vtr+NP + PP”, “Vtr+NP” and the GP sentences are respectively three, four and five, and the first two patterns have no backtrackings. Among the three patterns, the levels of pushing downs of GP sentences amount to as high as four levels, showing a much higher complexity than the former two patterns.

VIII. CONCLUSION

Globally ambiguous sentences and GP sentences have different syntactic structures. The ambiguous sentences entail a number of decoding selections and the successful decoding of GP sentences must exert a kind of single-selection system of semantic trigger, showing a higher cognitive burden. The reason why the decoding of GP sentences is more complex than the others lies in that

the data structure of the GP sentences turns out to be a cognitive tree diagram with an indispensable understanding repetition. The augmented transition networks (ATNs) allow greater application of showing various states of decoding, thus significantly contributing to our understanding of the difference between globally ambiguous sentences and GP sentences.

TABLE VI. THE CONTRAST OF THE COMPLEXITY OF THE 3 KINDS OF DECODING PATTERNS OF SENTENCES 13 AND 14

Model	“Vtr+NP+PP” Model	“Vtr+NP” Model	Garden Path Sentence Model	
Trace	Start-1-11-12-13-1-2-4-5	Start-1-11-12-13-1-2-4	Start-1-11-12-13-1-2-4-5-11-12-13-5-7-16-17-18-7-9-2-3-backtracking	
	-11-12-13-5-7	-5-11-12-13	Restart-1-11-12-13-19-20-11-12-13-20-21-15/13-1-2-backtracking	
	-16-17-	-14-16-17-	14-16-17-18-14-20-21-15/13-1-2-6	
	18-7-9-2-3-	18-14-5-8-9	-8-9-2-3-Success	
	Success	-2-3-	Success	
Hierarchy	3	4	5	
Backtracking	0	0	2	
Push	1-level	2	2	2(backtracking)+1(backtracking)+2
	2-level	2	1	2(backtracking)+1(backtracking)+1
	3-level	0	1	1(backtracking)+1
	4-level	0	0	1
	Total Push	4	4	7(backtracking)+5
Pop	1-level	2	2	2(backtracking)+1(backtracking)+2
	2-level	2	1	2(backtracking)+1(backtracking)+1
	3-level	0	1	1(backtracking)+1
	4-level	0	0	1
	Total Pop	4	4	7(backtracking)+5
Push and Pop Times	8	8	14(backtracking)+10	
Complexibility	★	★★	★★★	

ACKNOWLEDGMENT

This research is supported in part by grants of YB115-29 and YB115-41 from “the Eleventh Five-year” research projects of Chinese language application, and grant 08BYY046 from the National Social Science Foundation.

REFERENCES

[1] T. G.Bever, “The cognitive basis for linguistic structures,” In J. R. Hayes, Ed. *Cognition and the Development of Language*, New York: John Wiley and Sons, 1987, pp. 279-352.
 [2] B. L.Pritchett, “Garden path phenomena and the grammatical basis of language processing,” *Language*, 1988, pp. 539-576.
 [3] Y. H. Jin, “Semantic analysis of Chinese garden-path

- sentences,” in *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, Sydney, 2006.
- [4] K.Christianson, et al, “Thematic roles assigned along the garden path linger,” *Cognitive Psychology*, 2001, (42), pp. 368-407.
- [5] D. J. Foss and C. M. Jenkins, “Some effects of context on the comprehension of ambiguous sentences,” *Journal of Verbal Learning and Verbal Behavior*, 1973, (12), pp. 577.
- [6] K. G. D. Bailey and F. Ferreira, “Disfluencies affect the parsing of garden-path sentences,” *Journal of Memory and Language*, 2003, (49): pp.183-200.
- [7] Zh. W. Feng, “The automatic arithmetic of Garden Path Sentences,” *Contemporary Linguistics*, 2003, (4), pp 339-349.
- [8] B.Roark, “Robust garden path parsing,” *Natural Language Engineering*, 2004, (10), pp. 1 - 24.
- [9] P. F.Yu and J. L.Du, “Automatic analysis of textual garden path phenomenon: A computational perspective,” *Journal of Communication and Computer*, 2008, (10), pp. 58-65.
- [10] J. L. Du and P. F. Yu, “Towards the processing breakdown of syntactic garden path phenomenon: A semantic perspective of natural language expert system,” *Journal of Communication and Computer*, 2008, 5(11), pp. 53-61.
- [11] W. A. Woods, “Transition network grammars for natural language analysis,” *CACM*, 1970, (13), pp. 591-606.
- [12] R.M. Kaplan, “Augmented transition networks as psychological models of sentence comprehension,” *Artificial Intelligence*, 1972, (03), pp. 77-100.
- [13] G. Kempen, “Computational models of syntactic processing in human language comprehension,” In: T. Dijkstra and K. De Smedt, Eds., *Computational Psycholinguistics: Symbolic and Subsymbolic Models of Language Processing*, London: Taylor & Francis, 1996, pp.192-220.
- [14] N. D. Maxfield, J. M. Lyon, and E. R. Silliman, “Disfluencies along the garden path: Brain electrophysiological evidence of disrupted sentence processing,” *Brain and Language*, 2009, 111(2), pp. 86-100.
- [15] M. P. Wilson and S. M. Garnsey, “Making simple sentences hard: Verb bias effects in simple direct object sentences,” *Journal of Memory and Language*, 2009, 60(3), pp. 368-392.
- [16] Y. Choi and J. C. Trueswell, “Children’s inability to recover from garden paths in a verb-final language: Evidence for developing control in sentence processing,” *Journal of Experimental Child Psychology*, 2010, 106(1), pp. 41-61.
- [17] K. R.Christensen, “Syntactic reconstruction and reanalysis, semantic dead ends, and prefrontal cortex,” *Brain and Cognition*, 2010, 73(1), pp. 41-50.
- [18] K. Christianson, S. G. Luke, and F. Ferreira, “Effects of plausibility on structural priming,” *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 2010, 36(2), pp. 538-544.
- [19] M. Bader and J. Haussler, “Resolving number ambiguities during language comprehension,” *Journal of Memory and Language*, 2009, 61 (03), pp. 352-373.
- [20] W. Teubert, *Meaning, Discourse and Society*. Cambridge: United Kingdom, 2010.
- [21] N. D.Patson, et al, “Lingering misinterpretations in garden-path sentences: Evidence from a paraphrasing task,” *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 2009, 35(1), pp. 280-285.
- [22] J. L. Du and P. F. Yu, “Towards an algorithm-based intelligent tutoring system: Computing methods in syntactic management of garden path phenomenon,” *IEEE International Conference on Intelligent Computing and Intelligent System*, 2010, pp. 521-525.
- [23] E.Malaia, R. B. Wilbur, and C. Weber-Fox, “ERP evidence for telicity effects on syntactic processing in garden-path sentences,” *Brain and Language*, 2009, 108(3), pp. 145-158.
- [24] N. D. Patson and F. Ferreira, “Conceptual plural information is used to guide early parsing decisions: Evidence from garden-path sentences with reciprocal verbs,” *Journal of Memory and Language*, 2009, 60(4), pp. 464-486.
- [25] T. F.Jaeger, “Redundancy and reduction: Speakers manage syntactic information density,” *Cognitive Psychology*, 2010, 61(1), pp. 23-62.
- [26] A.Staub, “Eye movements and processing difficulty in object relative clauses,” *Cognition*, 2010, 116(1), pp. 71-86.