

recursionをめぐる

中 井 悟

1 はじめに

1.1 recursionをめぐる論争

*Science*誌2002年11月22日号でMarc D. Hauser, Noam Chomsky, W. Tecumseh Fitchの三名による“The faculty of language: What is it, who has it, and how did it evolve?”が発表されて以来, recursion (再帰性, 回帰性)をめぐる多くの議論が行われてきた。(この論文は慣例的に三人の名前の頭文字をとってHCFと呼ばれているので, 本稿でも, この論文, あるいは, この三人をHCFと表記することにする。)たとえば, recursionをテーマにして以下のような論文集や雑誌の特集が刊行されている。これらの論文集は, すべて, recursionをテーマにした会議がまず開催され, その会議での発表を論文集にしたものである。

Sauerland, U., & Gärtner, H.-M. (Eds.). (2007). *Interfaces + recursion = Language?: Chomsky's minimalism and the view from syntax-semantics*. Berlin: Mouton de Gruyter.

本書は, 2005年に, Chomskyがベルリンを訪れた際に開催されたシンポジウムでの発表を集めたものであるとのことである。Chomskyの講演と六人の研究者の発表が掲載されている。

van der Hulst, H. (Ed.). (2010a). *Recursion and human language*. Berlin: De Gruyter Mouton.

本書は, 2007年に, ピダハン語(Pirahã)にはrecursionが存在せず, した

がって、recursionはヒトという種に固有のものであるというHCFの仮説は支持できないと主張しているDaniel L. Everettによって企画された、人間言語におけるrecursionをテーマにした国際会議での発表に基づく論文集である。編者のvan der Hulstも、“It is perhaps fair to say that the conference call invited participants to take a critical stance regarding the claims in, or attributed to Hauser, Chomsky and Fitch (2002) and this is certainly reflected in the present very interesting collection.” (pp. xvii-xviii)と述べているように、会議は、当然、HCFに批判的なものである。

Biolinguistics Vol. 5, Nos. 1 & 2 (2011)

*Biolinguistics*という雑誌は基本的にはミニマリスト・プログラムの立場をとっている。この*Biolinguistics* Vol. 5, Nos. 1 & 2 (2011)はrecursionを特集した号である。

Roeper, T., & Speas, M. (Eds.). (2014). *Recursion: Complexity in cognition*. New York: Springer.

この論文集は、2009年5月にUniversity of Massachusetts, Amherstで開催された国際会議での発表に基づくものである。

Lowenthal, F., & Lefebvre, L. (Eds.). (2014a). *Language and recursion*. New York: Springer.

この本も、recursionをテーマとした会議での議論に基づく論文集である。これはプロシーディングズではなく、会議の参加者から選ばれた人たちが会議の後に、会議での討論の基づいて執筆した論文集であると編者たちは述べている。(p. vi)

1.2 Hauser, M. D., Chomsky, N., & Fitch, W. T. (2002). The faculty of language: What is it, who has it, and how did it evolve? *Science*, 298, 1569–1579.

recursionに関する論争が始まったきっかけがHCFであるので、まず、この

論文の内容を整理しておくのがよいであろう。¹

HCFでは、人間の言語能力(faculty of language)を、広い意味での言語能力 (faculty of language in the broad sense, FLBと略す) と狭い意味での言語能力 (faculty of language in the narrow sense, FLNと略す) の二つに分け、狭い意味での言語能力を特徴づけるのはrecursionであると述べている。

狭い意味での言語能力は次のように説明されている。

Faculty of language—narrow sense (FLN). FLN is the abstract linguistic computational system alone, independent of the other systems with which it interacts and interfaces. FLN is a component of FLB, and the mechanisms underlying it are some subset of those underlying FLB. (p. 1571)

FLNのキーとなる構成要素は計算システム (狭い意味でのsyntax) であり、この計算システムが生成するrepresentationが、音韻システムによって感覚 - 運動のインターフェイスに写像され、意味システムによって概念 - 意図のインターフェイスに写像される。そして、FLNの核となる特性はrecursionである。²

We assume, putting aside the precise mechanisms, that a key component of FLN is a computational system (narrow syntax) that generates internal representations and maps them into the sensory-motor interface by the phonological system, and into the conceptual-intentional interface by the (formal) semantic system All approaches agree that a core property of FLN is recursion, attributed to narrow syntax in the conception just outlined. FLN takes a finite set of elements and yields a potentially infinite array of discrete expressions. This capacity of FLN yields discrete infinity (a property that also characterizes the natural numbers). (p. 1571)

HCFは、言語能力の進化に関して三つの仮説が考えられるとしているが、彼らが支持するのは次のHypothesis 3である。

Hypothesis 3: Only FLN is uniquely human. On the basis of data reviewed below, we hypothesize that most, if not all, of FLB is based on mechanisms shared with nonhuman animals (as held by hypothesis 1). In contrast, we suggest that FLN—the computational mechanism of recursion—is recently evolved and unique to our species according to recent linguistic theory, the computations underlying FLN may be quite limited. In fact, we propose in this hypothesis that FLN comprises only the core computational mechanisms of recursion as they appear in narrow syntax and the mappings to the interfaces. (p. 1573)

この仮説では、recursionという計算メカニズムはヒトという種に固有であり、FLNはrecursionという核となる計算メカニズムだけを含むと主張されている。このことからこの仮説はrecursion-only hypothesisとも呼ばれることがある。

1.3 何が問題なのか

以上がHCFの簡単な内容紹介であるが、何が問題となっているのであろうか。

recursionに関して議論されていることは二つある。まず、そもそもrecursionとは何かである。その理由は、HCFがrecursionを明確に定義していないからだと言われている。³

もう一つの問題は、HCFが、“a core property of FLN is recursion, attributed to narrow syntax”, あるいは、“FLN—the computational mechanism of recursion—

is recently evolved and unique to our species”と言っていることである。HCFの、recursionはヒトという種に固有のものであるとしたこの仮説(recursion-only hypothesis)が正しいかどうかの議論である。

1.4 本稿の目的

この二つの問題は互いに独立した問題ではなく、当然、関連性がある。recursionをどう定義するかでrecursionがヒトという種に固有のものかどうかが変わってくるからである。本稿の目的は、最初の問題であるrecursionの定義を検討し、HCFはその定義を明確に述べなかったが、彼らの使っているrecursionという用語の意味は、ミニマリスト・プログラムを理解したうえでHCFが述べていることを正確に読み取れば明確になることを示すことである。

2 Hauser et al. (2002)のrecursionの定義が不明確であること

HCFがrecursionを明確に定義しなかったことがrecursionをめぐる議論を混乱させているという意見は数多くある。それらを年代順にみていこう。

2.1 Tomalin, M. (2007). Reconsidering recursion in syntactic theory. *Lingua*, 117, 1784-1800.

Tomalinはrecursionという用語が使われてきた経過を説明したうえで、最近になってrecursionがtopic of extensive debateになった理由は、recursionという語の定義が曖昧であるからであると述べている。

Given the discussion in section 2, though, it should not be surprising that the Hauser–Chomsky–Fitch (HCF) hypothesis (as it currently stands) is potentially ambiguous, and it should be obvious that this ambiguity is caused primarily by the confusion that has enveloped the term ‘recursion’ since the

late 1930s. (p. 1796)

recursionという語はいろいろな分野で以前から使用されてきており、それぞれの分野でその意味するところが違うのは当然である。

2.2 Kotowski, S., & Härtl, H. (2008). Recursion and the language faculty: Over the evolution of the concept in generative grammar.

この論文のabstractで、著者たちは、recursionをめぐる議論を特徴づけているのはrecursionという用語の定義が混乱していることであると述べている。

Recursion has been a central feature of syntactic theory in generative grammar since its establishment in the 1950s (Bar-Hillel, 1953; Chomsky, 1956; 1957). Yet, since the highly influential 2002 paper by Hauser, Chomsky, and Fitch, and their strong hypothesis regarding the outstanding status of recursion in language, there has been a renewed interest in the subject. The ensuing debates, however, have been characterized by severe terminological confusion and thus been rendered futile at times. (p. 1)

2.3 Coolidge, F. L., Overman, K. A., & Wynn, T. (2010). Recursion: What is it, who has it, and how did it evolve? Wiley Online Library.

Coolidge et al.は、HCFがrecursionを明確に定義しなかった（定義するのを忘れた？）と述べている。⁴

They [Hauser et al. (2002)] also failed to define recursion explicitly, other than stating it was part of a computational system that generated internal representations that mapped onto a sensory–motor interface by the phonological system and onto a conceptual–intentional interface by a formal

semantic system, enabling ‘us to communicate an endless variety of thoughts’ (p. 1574).

2.4 Fitch, W. T. (2010). Three meanings of “recursion”: Key distinctions for biolinguistics. In R. K. Larson, V. Depez, & H. Yamakido (Eds.), *The evolution of human language: Biolinguistic perspectives* (pp. 73-90). Cambridge: Cambridge University Press.

Fitch自身は、HCFの共同執筆者の一人であるが、recursionをめぐる議論においてrecursionが正確に定義されておらず、いろいろな意味で使われていると述べている。⁵

There has recently been a resurgence of scientific interest concerning the importance of various types of phrase structure in human language, and their potential presence in other species (Hauser, Chomsky, and Fitch 2002; Fitch and Hauser 2004; Everett 2005; Fitch, Hauser, and Chomsky 2005; Pinker and Jackendoff 2005; Gentner *et al.* 2006). Following Hauser, Chomsky, and Fitch (2002, HCF hereafter), many of these recent discussions have used the term “recursion,” but the term has rarely been defined explicitly. There are several possible interpretations of this word, which is used somewhat differently in different disciplines, without there being one universally accepted scientific definition. With the recent advent of intense interdisciplinary discussion of these issues, it has become clear that several different interpretations of the term are being used interchangeably. We seem to have reached a point where serious misunderstandings are in danger of propagating through the literature. (p. 73)

2.5 Jackendoff, R. (2011). What is the human language faculty?:

Two views. *Language*, 87, 586-624.

Jackendoffも、HCFが、ミニマリスト・プログラムとMergeに言及せずに recursionがFLNのcoreであると言っていると述べている。⁶

The minimalist program reconstructs syntactic theory around Merge as the central computational operation. Without specifically mentioning the minimalist program and Merge, Hauser, Chomsky, and Fitch (2002) propose that FLN consists solely of the ability to build syntactic structures recursively, plus the mappings from syntax to the ‘sensory-motor interface’—the auditory and motor systems in language perception and production respectively—and to the ‘conceptual-intentional interface’—the formation of thought and meaning, or what I termed ‘general intelligence’ above. (p. 591)

2.6 Speas, M. (2014). Recursion: Complexity in cognition. In T. Roeper & M. Speas (Eds.), *Recursion: Complexity in cognition* (pp. ix-xxi). New York: Springer.

Speasは、この本の序文で、HCFの仮説をめぐる議論は、recursionの定義が明確でないことによって混乱していると述べている。

That property, they [HCF] claim, crucially involves recursion. Much of the debate surrounding this claim has been hampered by a certain lack of clarity in the definition of recursion. (p. x)

2.7 Lowenthal, F., & Lefebve, L. (2014b). Are there any relations between recursion and language? In F. Lowenthal & L. Lefebve (Eds.), *Language and recursion* (pp. v-vii). New York: Springer.

これはこの本の序文であるが、その中で、LowenthalとLevebveは、

recursion をめぐる議論はあるが、recursion が正確に定義されていないと言っている。⁷

Hauser, Chomsky & Fitch (2002) claimed that the faculty to use recursive processes is the unique difference between human and animal communication. This position is contested by certain authors (Pinker & Jackendoff, 2004; Gervain, Macagno, Cogoi, Peña, & Mehler, 2008; Hochmann, Azadpour, & Mehler, 2008) but approved by others (Premack, 2004). Fitch and Hauser (2004) showed that tamarin apes and human adults react in the same way to violations of a Finite State Grammar (FSG—a grammar based on concatenation) but that tamarin apes do not react to violations of a Phrase Structure Grammar (PSG—a grammar based on non-tail recursive hierarchies) and claimed that this was evidence that nonhuman animals do not have access to recursive processes. Gentner, Fenn, Margoliash, & Nusbaum (2006) worked with songbirds and claimed the contrary, but Corballis (2007) considered that none of these papers provided us with conclusive evidence of the presence of recursion in nonhuman animals.

Nevertheless, the above mentioned authors did never specify which exact definition of “recursion” they use. (p. v)

2.8 Watumull, J., Hauser, M. D., Roberts, I. G., & Hornstein, N. (2014). On recursion. *Frontiers in Psychology*, 4, 1-7.

この著者たちも、言語学者たちのconcept of recursionはconfusedであると述べている。

It is a truism that conceptual understanding of a hypothesis is required for

its empirical investigation. However, the concept of recursion as articulated in the context of linguistic analysis has been perennially confused. Nowhere has this been more evident than in attempts to critique and extend Hauser et al's. (2002) articulation. These authors put forward the hypothesis that what is uniquely human and unique to the faculty of language—the faculty of language in the narrow sense—is a recursive system that generates and maps syntactic objects to conceptual-intentional and sensory-motor systems. This thesis was based on the standard mathematical definition of recursion as understood by Gödel and Turing, and yet has commonly been interpreted in other ways, most notably and incorrectly as a thesis about the capacity for syntactic embedding. As we explain, the recursiveness of a function is defined independent of such output, whether infinite or finite, embedded or unembedded—existent or non-existent. And to the extent that embedding is a sufficient, though not necessary, diagnostic of recursion, it has not been established that the apparent restriction on embedding in some languages is of any theoretical import. Misunderstanding of these facts has generated research that is often irrelevant to the FLN thesis as well as to other theories of language competence that focus on its generative power of expression. (p. 1)

後に述べるように、HCFのrecursionは数学におけるrecursionの定義に沿っており、Watumull et al.もそのことを指摘している(“This thesis was based on the standard mathematical definition of recursion”)。ところが、HCFのrecursionが他の意味で解釈され、間違ってrecursionがembeddingと解釈されているとWatumull et al.は述べている(“has commonly been interpreted in other ways, most notably and incorrectly as a thesis about the capacity for syntactic embedding”)。

2.9 Al-Mutairi, F. R. (2014). *The minimalist program: The nature and plausibility of Chomsky's biolinguistics*. Cambridge: Cambridge University Press.

Al-Mutairiは、Chomskyを批判している学者であるが、彼は、まず、“Merge/recursion”という表現を使っている。

As mentioned in several places above (especially in Sections 2.3 and 2.6), Chomsky subscribes to the view that Merge/recursion may be the only property that is unique to humans and to language. (p. 191, note 11)

“Merge/recursion”という表記をするということはrecursionをMergeと解釈していることを示しているのであろう。しかし、ChomskyがMergeで意味することとHCFがrecursionで意味することは同一かという問題がある。この区別が曖昧なので混乱が生じているとAl-Mutairiは言う。

On the basis of the above remarks, one might be inclined to conclude that FLN is, *mutatis mutandis*, identical to UG (i.e. Factor I). However, the qualifications that are required by the *mutatis mutandis* clause have empirical implications that are too important to ignore. This is particularly so when one asks whether what Chomsky means by “Merge” is what Hauser *et al.* recognize as “recursion.” It appears that the latter is much more general and inclusive, assimilating a range of technology beyond Merge into the language-specific recursive device. If this is true, as I will argue in the next chapter, it follows that the claim that FLN contains only recursion will have empirical content different from that of the claim that UG contains only Merge. As the next chapter will illustrate, it is through failure to appreciate this point that the recursion-only hypothesis has created considerable

confusion, not only among critics, but also among supporters. (p. 81)

ここでAl-Mutairiが、recursionは“much more general and inclusive, assimilating a range of technology beyond Merge into the language-specific recursive device”と言っていることに注目しよう。recursionはMergeではなく、Mergeはrecursive deviceを使用しているのである。⁸

3 recursionの二つの解釈

HCFはrecursionを明確に定義していないのであるが、おおざっぱに言えば、recursionという用語の解釈には二つある。

一つの解釈はrecursionはembeddingであるというものである。代表的な例としてEverettを取り上げよう。Everettは、アマゾンの奥地で話されているピダハン語にはembeddingがないので、HCFの仮説(“FLN comprises only the core computational mechanisms of recursion”)は成立しないと主張している学者である。Everett (2005)が彼のピダハン語の研究として代表的なものであるが、この論文ではrecursionという用語は使用されておらず、embeddingという用語が使用されており、embeddingは、“putting one phrase inside another of the same type or lower level, e.g., noun phrases in noun phrases, sentences in sentences, etc.” (p. 622)と説明されているだけである。

Everettがrecursionという用語を使用し、かつ、recursionをembeddingと定義している例はEverett (2007)に見られる。

Recursion is a property of algorithms generally. In its application to linguistics, it implies that one unit (word, phrase, or sentence) appears in another unit of the same type. So ‘John’s brother’s house’ shows that the appearance of the noun phrase ‘brother’s house’ inside (at the position of ‘x’) the larger noun phrase ‘John’s x.’ Or ‘the man who is tall here’ contains

the sentence ‘who is tall’ inside the larger sentence ‘The man is here.’ As a further example we have ‘truck driver’ which contains the two words ‘truck’ and ‘driver’ inside a single larger word. (p. 298)

標準理論の生成文法では、 $S \rightarrow NP VP$, $VP \rightarrow V S$ という句構造規則を適用すれば、Sが二回出てきて、[*s John believes [s the dog sleeps]*]という埋め込み構造が派生される。文が文の中に埋め込まれるのである。これが標準理論で recursion と呼ばれていたものである。

もう一つ、recursionをembeddingと解釈している例をあげよう。Pinker & Jackendoff (2005)である。この論文は、Hauser et al. (2002)への反論なのであるが、PinkerとJackendoffが考えているrecursionはembeddingである。⁹

Recursion refers to a procedure that calls itself, or to a constituent that contains a constituent of the same kind (p. 203)

Recursion consists of embedding a constituent in a constituent of the same type, for example a relative clause inside a relative clause (*a book that was written by the novelist you met last night*), which automatically confers the ability to do so ad libitum (e.g. *a book [that was written by the novelist [you met on the night [that we decided to buy the boat [that you liked so much]]]]*). (p. 211)

もう一つの解釈はrecursionはMerge（併合）の繰り返し適用であるというものである。Nevins et al. (2009)は、ミニマリストの視点から、Everett (2005)を批判して、ピダハン語にもrecursionは存在すると主張している論文であるが、その中で、HCFで使われているrecursionはMergeの繰り返し適用のことであり、embeddingのことではないと述べている。Nevins et al. (2009)は、

Mergeが繰り返し適用されることによって階層的な句が生成されるのであると述べている。

In recent work associated with the minimalist program, hierarchical phrase structure is understood as a reflection of the iterated application of the structure-building rule Merge (Chomsky 1995). (p. 365)

Merge takes two linguistic units as input and combines them to form a set (a PHRASE), in which one element is designated as the phrase's head. Two kinds of linguistic units may serve as input to Merge: (i) lexical items, and (ii) phrases formed by previous applications of Merge. Since Merge may take previous applications of Merge as input, the rule is RECURSIVE. Iterated Merge yields the full variety of phrase structures studied in syntactic research—structures composed of lexical items and phrases that were themselves produced by Merge. (p. 365)

重要なことは、上の引用文中にも述べられているように、recursionがMergeの単なる繰り返し適用ではなく、Mergeを適用して二つの要素を併合した出力が、次のMergeの適用の入力になることである。これがrecursionの重要な特性である。Oxford Advanced Learner's Dictionary (8th edition)のrecursionの定義は次のようである。

[uncountable] (mathematics)

the process of repeating a function, each time applying it to the result of the previous stage

また、『ランダムハウス英語辞典』（CD-ROM版）のrecursionの定義は次の

ようである。

〔数学〕〔コンピュータ〕 帰納, 再帰, 帰納的方法 [定義]:あるルールやステップの結果を逐次用いて, 数や関数を構成したり, 処理手順を構築すること

また, 研究社の『新英和大辞典』(第6版, CD-ROM版)でも, この用語はコンピュータや数学の用語としてその定義が次のように示されている。

1【電算】再帰

2【数学】帰納, 回帰《列をなす数学的対象の各項を, それに先立つ項から一定の規則で決定すること》

これらの定義に示されているように, recursionという用語は, 元来は, 数学やコンピュータの用語であり, HCFもこの定義に沿った意味でrecursionという語を使用しているのである。

そして, Nevins et al. (2009)は, このMergeはピダハン語でも使われていると主張する。なぜなら, Mergeは, embeddingのことではなく, 複数の語彙項目を結合して句や文を派生していくことであり, ピダハン語には複数の語からなる句や文が存在するのであるから, Mergeがなければ複数の語からなる句や文は存在しないはずであるというのである。¹⁰

In fact, precisely those constructions that were typically presented as evidence for recursion in earlier models are the ones discussed in *CA* under the rubric of absence of embedding. Indeed, in the context of early generative grammar, the absence of this list of constructions from a language might have constituted a demonstration that the language lacks recursion. In a model

with category-neutral Merge, however, a language that lacks recursion would be considerably more exotic. No sentence in such a language could contain more than two words. Pirahã is manifestly not such a language. (p. 366, note 11)

ここでMergeはcategory-neutralと述べられているのは、Mergeによって併合される要素は特定の範疇ではないからである。ある句を同じ範疇の句の中に(名詞句を名詞句の中に、文を文の中に)埋め込むことではないのである。

Nevins et al. (2009)がミニマリストの立場から、Everett (2005)がrecursionをembeddingと解釈しているのは間違いであると主張していることからわかるように、HCFのrecursionはMergeの繰り返し適用のことである。以下の4節と5節で、Nevins et al. (2009)が言うように、HCFはrecursionをMergeの繰り返し適用の意味で使っていることを明らかにしていくことにする。

4 ミニマリスト・プログラムにおけるrecursion

まず、ミニマリスト・プログラムそのものの確認から始めよう。¹¹

4.1 ミニマリスト・プログラム(Chomsky (1995))における recursion

HCFが前提にしている言語理論がミニマリスト・プログラムなので、まず、Chomsky (1995)に基づいてミニマリスト・プログラムの復習をしておこう。¹²

Chomsky (1995)の4.2.1 The Computational Componentという節(pp. 225-235)のミニマリスト・プログラムの説明を見てみよう。

まず、この節以前からの抜粋で必要な用語の意味を確認しておく。

... each language will determine a set of pairs (π, λ) (π drawn from PF [Phonetic Form] and λ from LF [Logical Form]) as its formal representations of sound and meaning, insofar as these are determined by the language

itself. Parts of the computational system are relevant only to π , not λ : the *PF component*. Other parts are relevant only to λ , not π : the *LF component*. The parts of the computational system that are relevant to both are the *overt syntax*—a term that is a bit misleading, in that these parts may involve empty categories assigned no phonetic shape. The nature of these systems is an empirical matter; one should not be misled by unintended connotations of such terms as “logical form” and “represent” adopted from technical usage in different kinds of inquiry. (p. 169)

ある語彙項目には π と λ という二つの表示がある。 π というのはある語彙項目の音韻表示であり、 λ は意味表示である。計算システムの一部は音声部門(PF component)に関係し、ある部分は論理部門(LF component)に関係する。両者に関係する計算システムの部分が overt syntax で作用する。(covert syntax で作用する計算システムもあるということである。)

以下で、4.2.1 The Computational Component という節から3箇所抜粋してある。いずれも同じようなことを説明している。C_{HL} というのは computational system for human language のことである。

We can, then, think of C_{HL} as mapping some array A of lexical choices to the pair (π , λ). What is A? At least, it must indicate what the lexical choices are and how many times each is selected by C_{HL} in forming (π , λ). Let us take a *numeration* to be a set of pairs (LI, i), where LI is an item of the lexicon and i is its index, understood to be the number of times that LI is selected. Take A to be (at least) a numeration N; C_{HL} maps N to (π , λ). The procedure C_{HL} selects an item from N and reduces its index by 1, then performing permissible computations. A computation constructed by C_{HL} does not count as a derivation at all, let alone a convergent one, unless all indices are

reduced to zero. (p. 225)

Given the numeration N , the operations of C_{HL} recursively construct syntactic objects from items in N and syntactic objects already formed. We have to determine what these objects are and how they are constructed. . . .

Suppose that the derivation has reached the stage Σ , which we may take to be a set $\{SO_1, \dots, SO_n\}$ of syntactic objects. One of the operations of C_{HL} is a procedure that selects a lexical item LI from the numeration, reducing its index by 1, and introduces it into the derivation as SO_{n+1} . Call the operation *Select*. At the LF interface, Σ can be interpreted only if it consists of a single syntactic object. Clearly, then, C_{HL} must include a second procedure that combines syntactic objects already formed. A derivation converges only if this operation has applied often enough to leave us with just a single object, also exhausting the initial numeration. The simplest such operation takes a pair of syntactic objects (SO_i, SO_j) and replaces them by a new combined syntactic object SO_k . Call this operation *Merge*. We will return to its properties, merely noting here that the operations *Select* and *Merge*, or some close counterparts, are necessary components of any theory of natural language. (p. 226)

For the moment, then, the syntactic objects we are considering are of the following types:

- (5) a. lexical items
 b. $K = \{\gamma, \{\alpha, \beta\}\}$, where α, β are objects and γ is the label of K

Objects of type (5a) are complexes of features, listed in the lexicon. The

recursive step is (5b). Suppose a derivation has reached state $\Sigma = \{\alpha, \beta, \delta_i, \dots, \delta_n\}$. Then application of an operation that forms K as in (5b) converts Σ to $\Sigma' = \{K, \delta_i, \dots, \delta_n\}$, including K but not α, β . In a convergent derivation, iteration of operations of G_{HL} maps the initial numeration N to a single syntactic object at LF. (p. 243)

以上がミニマリスト・プログラムの説明であるが、下線部の“the operations of C_{HL} recursively construct *syntactic objects* from items in N and syntactic objects already formed”や“ C_{HL} must include a second procedure that combines syntactic objects already formed”は、Mergeが繰り返し適用されて文が派生されることを説明している。 C_{HL} （つまりMerge）は、語彙項目とすでにMergeで生成された要素を再度併合するのである。これを繰り返して句や文が生成されるのである。ミニマリスト・プログラムではrecursionはMergeを繰り返し適用することなのである。1995年の段階ですでにrecursionはMergeの繰り返し適用であると述べられているのである。

ミニマリスト・プログラムには句構造規則などはないのであるから、当然、 $S \rightarrow NP VP$, $VP \rightarrow V S$ という句構造規則はない。実際にどのように文が派生されるのかを、Hornstein et al. (2005: pp. 69-70)の例を借りて具体的に説明しよう。たとえば、that woman might buy that carという文の派生を見てみよう。まず、Lexiconから選んだmight, that, buy, woman, carという5つの語彙項目(lexical item)がある。N (numeration)は、語彙項目とその語彙項目が何回計算に利用できるかを示すindexの組の集合であるから、この場合のNは次のようになる。

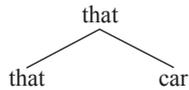
$$N = \{\text{might}_1, \text{that}_2, \text{buy}_1, \text{woman}_1, \text{car}_1\}$$

このNから、まず、carとthatをSelectする。(carのindexは1から0になり、

thatのindexは2から1になる。)そして, thatとcarをMergeしてthat carという(決定詞)句を派生する。これが第一回目のMergeの適用である。¹³

$N = \{\text{might}_1, \text{that}_1, \text{buy}_1, \text{woman}_1, \text{car}_0\}$

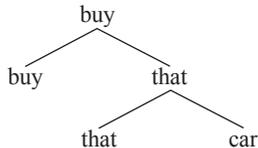
that + car → that car



次に, NからbuyをSelectし(buyのindexも1から0になる), buyとthat carをMergeしてbuy that carという(動詞)句を派生する。これが第二回目のMergeの適用である。

$N = \{\text{might}_1, \text{that}_1, \text{buy}_0, \text{woman}_1, \text{car}_0\}$

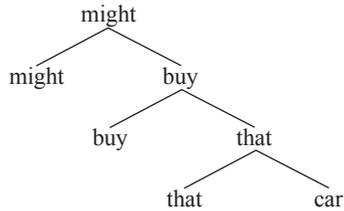
buy + that car → buy that car



次に, Nからmight(mightのindexも1から0になる)をSelectし, mightとbuy that carをMergeするとmight buy that carという句(IP (Inflectional Phrase)あるいはTP (Tense Phrase)と呼ばれる)が派生される。これが第三回目のMergeの適用である。

$N = \{\text{might}_0, \text{that}_1, \text{buy}_0, \text{woman}_1, \text{car}_0\}$

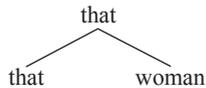
might + buy that car → might buy that car



さらに、NからthatとwomanをSelectし（thatのindexは1から0になり、womanのindexも1から0になる）、thatとwomanをMergeする。これが第四回目のMergeの適用である。

$N = \{\text{might}_0, \text{that}_0, \text{buy}_0, \text{woman}_0, \text{car}_0\}$

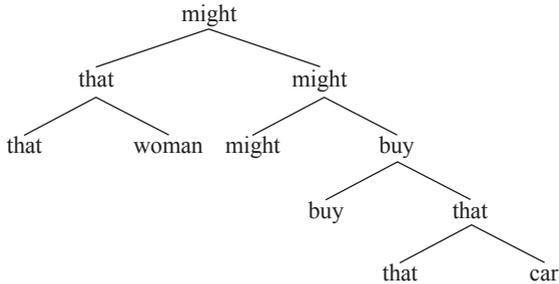
that + woman → that woman



最後に、that womanという句とmight buy that carという句をMergeすると、that woman might buy that carという文が派生できる。これが第五回目のMergeの適用である。Nの語彙項目はすべてindexが0になっており、使い切っているので派生はこれで終了である。

$N = \{\text{might}_0, \text{that}_0, \text{buy}_0, \text{woman}_0, \text{car}_0\}$

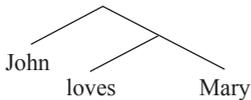
that woman + might buy that car → that woman might buy that car



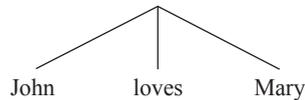
この派生では、Mergeが五回繰り返し適用されている。

重要なことは、recursionは単なる繰り返し(iteration)ではないということである。Mergeの繰り返し適用は階層構造を構築するのである。たとえば、John loves Maryという文は、まず、lovesとMaryをMergeしてloves Maryを作り、次にJohnとloves MaryをMergeしてJohn loves Maryという文を作るが、その構造は以下のaに示す階層構造であり、bに示すようなフラットな構造ではない。

a.



b.



もう一つ重要なことは、3節でも述べておいたように、recursionは単なる繰り返し(iteration)ではなく、ある操作の適用の出力が同じその操作の次の適用の入力になることである。ミニマリスト・プログラムのMergeの繰り返し適用は元来の数学における定義に沿ったものである。

4.2 2000年代のChomskyのミニマリスト・プログラムにおける recursion

このように、ミニマリスト・プログラムではrecursionはMergeの繰り返し適用である。Chomsky (1995)以降の生成文法理論はミニマリスト・プログラムであるから、当然、recursionはMergeの繰り返し適用である。そのことを確認しておこう。

4.2.1 Chomsky, N. (2007). Approaching UG from below. In U. Sauerland & H.-M. Gärtner (Eds.), *Interfaces + recursion = Language?: Chomsky's minimalism and the view from syntax-semantics* (pp.1-29). Berlin: Mouton de Gruyter.

ChomskyはMergeとMergeに関連する諸問題について次のように述べている。¹⁴

An I-language is a computational system that generates infinitely many internal expressions, each of which can be regarded as an array of instructions to the interface systems, sensorimotor (SM) and conceptual-intentional (CI). To the extent that third factor conditions function, the language will be efficiently designed to satisfy conditions imposed at the interface . . . (p. 5)

In its most elementary form, a generative system is based on an operation that takes structures already formed and combines them into a new structure. Call it Merge. (p. 5)

この説明から、“a computational system that generates infinitely many internal expressions”はMergeというoperationを繰り返し適用していることは明白である。

4.2.2 Chomsky, N. (2014). Minimal recursion: Exploring the prospects. In T. Roeper & M. Speas (Eds.), *Recursion: Complexity in cognition* (pp. 1-15). New York: Springer.

Chomsky (2014)はEverett (2005)より9年後の論文であるが、Roeper & Speas (2014)に掲載されていることもあるので、Chomskyのrecursionの考え方を知っておくためにあえてその内容を紹介しておこう。

この論文では、Chomskyはrecursionを次のように考えている。

For our purposes, we can think of recursion as enumeration of a set of discrete objects by a computable finitary procedure, one that can be programmed for an ordinary digital computer that has access to unlimited memory and time. (p. 1)

recursionを説明するために“procedure”という用語が使われている。繰り返し適用するという手順なのである。

以下の文章中にもrecursionが使われているが、この文脈からしてもrecursionがembeddingでないことは明白である。

The modern sciences have largely adopted Galileo’s methodological guideline that nature is simple, keeping to the “easiest and simplest rules,” as he put it: the task of the scientist is to try to establish the principle, despite the heterogeneity and diversity of phenomena. In our special case, the task is to determine how closely language approximates minimal recursion, within the boundary conditions set by interface satisfaction. That is why a recent collection of essays has the title *Interfaces + Recursion = Language*—followed by a big ?, because there is so much that is not understood. (p. 3)

A finitary computational procedure P will have buried in it in some form an operation —call it *Merge*—that takes objects already constructed and forms from them a new object, beginning with a set of atomic objects (which may have internal structure). To first approximation, we can take the atomic objects to be lexical items drawn from the lexicon, though this is not an innocent move. We can therefore think of P as having available a *work space* consisting of the lexicon (or some subpart extracted from it for the purpose of the computation) and objects that have already been formed by P. The optimal assumption is that Merge takes the simplest form: a binary operation that applies to X, Y, forming $Z = \{X, Y\}$, with X and Y unchanged by the operation (the No-tampering Condition NTC), and also unordered. (p. 7)

もちろん、二つ目の引用はMergeの説明である。Lexiconから語彙項目を選び、Numerationという集合を作る。このNumerationから語彙項目を選択してMergeで新たな組み合わせを派生し、これを繰り返すことによってa set of discrete objectsを派生するのである。

5 Hauser et al. (2002)のrecursionはMergeの繰り返し適用である

5.1 Hauser et al. (2002)のrecursionはMergeの繰り返し適用である

Nevins et al. (2009)は、HCFのrecursionはMergeの繰り返し適用であると述べているが、その主張が間違いでないことを確認しておく必要がある。

Verhagen (2010)は、Hauser et al. (2002, p. 1571)から次のような文章を引用している。

All approaches agree that a core property of FLN is recursion, attributed to

narrow syntax in the sense just outlined. FLN takes a finite set of elements and yields a potentially infinite array of discrete expressions. [. . .].

The core property of discrete infinity is intuitively familiar to every language user. Sentences are built out of discrete units [. . .]. There is no longest sentence (any candidate sentence can be trumped by, for example, embedding it in “Mary thinks that . . .”), and there is no non-arbitrary upper bound to sentence length. [. . .]

At a minimum, then, FLN includes the capacity of recursion. (Verhagen, 2010, p. 96)

この引用された文章の中には、“Mary thinks that . . .”という補文構造の例が挙げられているので、あたかもHCFがrecursion = embeddingと解釈しているかのように著者は説明しているが、この引用が含まれているパラグラフ全体を読むと、HCFはミニマリスト・プログラムの枠組みでrecursion（つまり、Mergeの繰り返し適用）を説明していることは明らかである。以下の文章は、Verhagen (2010)が引用したHCFの文章が含まれている部分であるが、これは、4.1節で紹介したミニマリスト・プログラムそのものを解説したものに他ならない。

All approaches agree that a core property of FLN is recursion, attributed to narrow syntax in the conception just outlined. FLN takes a finite set of elements and yields a potentially infinite array of discrete expressions. This capacity of FLN yields discrete infinity (a property that also characterizes the natural numbers). Each of these discrete expressions is then passed to the sensory-motor and conceptual-intentional systems, which process and elaborate this information in the use of language. Each expression is, in this sense, a pairing of sound and meaning. It has been recognized for thousands

of years that language is, fundamentally, a system of sound-meaning connections; the potential infiniteness of this system has been explicitly recognized by Galileo, Descartes, and the 17th-century “philosophical grammarians” and their successors, notably von Humboldt. One goal of the study of FLN and, more broadly, FLB is to discover just how the faculty of language satisfies these basic and essential conditions.

The core property of discrete infinity is intuitively familiar to every language user. Sentences are built up of discrete units: There are 6-word sentences and 7-word sentences, but no 6.5-word sentences. There is no longest sentence (any candidate sentence can be trumped by, for example, embedding it in “Mary thinks that . . .”), and there is no nonarbitrary upper bound to sentence length. In these respects, language is directly analogous to the natural numbers (see below).

At a minimum, then, FLN includes the capacity of recursion. (p. 1571)

Verhagenが引用している“FLN takes a finite set of elements and yields a potentially infinite array of discrete expressions.”というのは、まさにミニマリスト・プログラムでのMergeの繰り返し適用による句や文の派生のことである。computational systemがLexiconから語彙項目を取り出し、Numerationを作り、そのNumerationにある語彙項目をMergeして文を構築することである。このMergeを繰り返し(recursively)適用することによってa potentially infinite array of discrete expressionsが作られるのである。

Verhagenが引用した文章からもわかるように、HCFはミニマリスト・プログラムを前提にしており、recursionは、Mergeの繰り返し適用のことであって、決してembeddingのことではない。

また、Verhagenは、“Natural languages go beyond purely local structure by including a capacity for recursive embedding of phrases within phrases, which can

lead to statistical regularities that are separated by an arbitrary number of words or phrases.”という文も引用しているが、この文はembeddingそのものの説明ではなく、embeddingによってlong-distance dependencyを説明できると言っているだけである。HCFは次のように説明している。

A second possible limitation on the class of learnable structures concerns the kinds of statistical inferences that animals can compute. Early work in computational linguistics (108–110) suggested that we can profitably think about language as a system of rules placed within a hierarchy of increasing complexity. At the lowest level of the hierarchy are rule systems that are limited to local dependencies, a subcategory of so-called “finite-state grammars.” Despite their attractive simplicity, such rule systems are inadequate to capture any human language. Natural languages go beyond purely local structure by including a capacity for recursive embedding of phrases within phrases, which can lead to statistical regularities that are separated by an arbitrary number of words or phrases. Such long-distance, hierarchical relationships are found in all natural languages for which, at a minimum, a “phrase-structure grammar” is necessary. It is a foundational observation of modern generative linguistics that, to capture a natural language, a grammar must include such capabilities . . . (p. 1577)

下線部が言っていることは、recursionがembeddingということではなく、embeddingを繰り返すことによっていくらかでも長い文を派生できると言っているだけである。

HCFがミニマリスト・プログラムを前提としている証拠をもう一つ紹介しよう。HCFの本文を注意深く読んでみると、次のような記述を発見する。

Recent work on FLN (4, 41–43) suggests the possibility that at least the narrow-syntactic component satisfies conditions of highly efficient computation to an extent previously unsuspected. Thus, FLN may approximate a kind of “optimal solution” to the problem of linking the sensory-motor and conceptual-intentional systems. In other words, the generative processes of the language system may provide a near-optimal solution that satisfies the interface conditions to FLB. (p.1574)

ここで“recent work on FLN (4, 41-43)”とあるのは次の四つである。HCFのReferences and Notesにリストされている形式のままに列挙する。

4. R. Jackendoff, *Foundations of Language* (Oxford Univ. Press, New York, 2002).
41. N. Chomsky, *The Minimalist Program* (MIT Press, Cambridge, MA, 1995).
42. C. Collins, *Local Economy* (MIT Press, Cambridge, MA, 1997).
43. S. D. Epstein, N. Hornstein, *Working Minimalism* (MIT Press, Cambridge, MA, 1999).

“Minimalist”や“Minimalism”をタイトルに含む文献である。

しかも、“[. . .] FLN may approximate a kind of ‘optimal solution’ to the problem of linking the sensory-motor and conceptual-intentional systems. In other words, the generative processes of the language system may provide a near-optimal solution that satisfies the interface conditions to FLB.”というのはChomsky (1995)でも述べられているミニマリスト・プログラムのinterface conditionsのことである。

interface conditionsに関しては、Chomsky (1998)に次のような説明がある。

To introduce some terminology of MP [Minimalist Program], We [*sic*] say that a computation of an expression EXP converges at an interface level IL if EXP is legible at IL, consisting solely of elements that provide instructions to the external systems at IL and arranged so that these systems can make use of them; otherwise it crashes at IL. The computation converges if it converges at all interfaces. Call the expression EXP so formed convergent as well. As in MP, we keep here to a restricted version of the concept of convergence, setting aside the matter of legible arrangement (which raises all sorts of complex issues), and tentatively assuming it to be irrelevant—no slight simplification. Certain features of lexical items are interpretable, that is, legible to the external systems at the interface; others are uninterpretable. We assume, then, that if an expression contains only features interpretable at IL, it converges at IL. (pp. 7-8) (下線は原文のまま)

Mergeの繰り返し適用によって派生された表現に含まれるfeatureはsensorimotor interfaceとconceptual-intentional interfaceで読み取り可能(legible), つまり, 解釈可能(interpretable)でなければならないのである。これがinterface conditionsである。

Chomsky (2007)の次の説明も参考になる。

An I-language is a computational system that generates infinitely many internal expressions, each of which can be regarded as an array of instructions to the interface systems, sensorimotor (SM) and conceptual-intentional (CI). To the extent that third factor conditions function, the language will be efficiently designed to satisfy conditions imposed at the interface . . . (p. 5)

HCFのrecursionがMergeの繰り返し適用であることを示すさらなる証拠を紹介しよう。

Fitch et al. (2005)は、HCFを批判したPinker & Jackendoff (2005)への回答であるが、この回答の中で、“recursive operation”とか“place for recursion”とか“recursive mapping capability”とか“recursive machinery”とか“mechanisms of recursion”という表現が使われている。このような文脈ではrecursionをembeddingと解釈することはできない。embedding operationとかplace for embeddingとかembedding mapping capabilityとかembedding machineryとかmechanism of embeddingとかいう表現は奇妙であろう。^{15,16}

To be precise, we suggest that a significant piece of the linguistic machinery entails recursive operations, and that these recursive operations must interface with SM and CI (and thus include aspects of phonology, formal semantics and the lexicon insofar as they satisfy the uniqueness condition of FLN, as defined). (p. 182)

The only assumption made in HCF, and here, about syntactic theory is the uncontroversial one that, minimally, it should have a place for recursion. (p. 183)

FLN is restricted to a simple but powerful recursive mapping capability by definition, unique to humans and unique to the language faculty. This recursive mechanism has some plausible precursors in cognitive domains other than communication. (p. 189)

our Hypothesis 3 can simply be restated as specific to the recursive machinery and associated mappings (pp. 189-190)

Second and less trivially, we argued that the mechanisms of recursion and its mappings are simple enough to nullify the adaptationist's "argument from design," a proposition one can question. (p. 190)

Chomsky et al. (2005)は、そのタイトルの“Appendix. The minimalist program. (Appendix to Fitch et al. (2005))”が示すように、元々は、Fitch et al. (2005)に含まれていたものであるが、*Cognition*の編集者の要請で元の論文から削除したので、appendixという形でオンラインで公開されたものである。

このappendixが元々含まれていたFitch et al. (2005)は、HCFを批判しているPinker & Jackendoff (2005)への回答であるので、HCFで主張されていることを改めて説明したものである。このappendixで、Chomskyたちは“The ‘core computational mechanisms of recursion’ include the indispensable operation *Merge* and the principles it satisfies.”と述べている。Hauser et al. (2002)への批判であるPinker & Jackendoff (2005)への回答であるFitch et al. (2005)のAppendixに“operation *Merge*”という表現があるのであるから、Hauser et al. (2002)のrecursionがembeddingでなくMergeの繰り返し適用であることは明らかである。

5.2 Hauser et al. (2002)のrecursionをMergeの繰り返し適用であると解釈している例

Nevins et al. (2009)は、HCFのrecursionはミニマリスト・プログラムのrecursionであり、そのrecursionはMergeの繰り返し適用のことであると主張しており、彼らのその主張が正しいことは5.1節で確認したが、Nevins et al. (2009)だけでなく、他にも、HCFのrecursionはミニマリスト・プログラムのrecursionであり、そのrecursionはMergeの繰り返し適用であると正しく解釈している研究者はいる。そのような研究者のHCFのrecursionに関する記述を見てみよう。

5.2.1 Zwart, J.-W. (2011). Recursion in language: A layered-derivation approach. *Biolinguistics*, 5, 43-56.

著者は、まずMergeの説明をしたうえで、Mergeによって生成される句構造はTransferというprocedureの繰り返しによっても生成できると言っているが、Mergeがprocessであることは強調している。¹⁷

The point of the example is not to argue that Merge does not exist or that Transfer is superior to Merge. The point is that Merge and Transfer are equivalent in important respects, and though Merge may be a useful shorthand for Transfer, recursion should be identified in terms of the process, not its notation.

More generally, we cannot decide that a language (or natural language) 'is recursive' by simply looking at its structures. We have to know about the procedure by which these structures are derived (see also Everett 2009: 438).
(p. 46)

この引用を読めば、この著者はrecursionをある操作を繰り返し適用することと解釈していることがわかる。

5.2.2 Roeper, T. (2011). The acquisition of recursion: How formalism articulates the child's path. *Biolinguistics*, 5, 57-86.

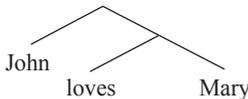
Roeperは、recursionはMergeを繰り返し適用することであるという立場をとっている。

First, let us distinguish between a completely universal form of recursion, namely Merge, and language-specific forms. Merge is a binary recursive

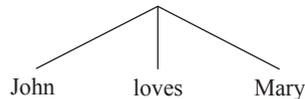
operation that is invoked as soon as more than two words are combined.
Therefore all languages with 3 word combinations are examples of recursion
over two binary acts of Merge. It is possible to imagine a three-term concatenation without a binary substructure, but empirical arguments exist to demonstrate that this is not the case for humans in structure-building beyond conjunctive relations, which we will elaborate in what follows. The presence of recursive Merge means that all languages must be recursive in a fundamental sense, just as Hauser *et al.* (2002) have claimed, which constitutes a strong biological claim. (pp. 58-59)

三語からなる句あるいは文は、二つの要素をMergeするという操作を二回適用して生成される。たとえば、John loves Maryという文は、まず、lovesとMaryをMergeし、次に、Johnとloves MaryをMergeして生成される。Mergeはbinary recursive operationなのである。Mergeの繰り返し適用によって次のaのような構造が生成され、bのような構造は生成されない。

a.



b.



5.2.3 Speas, M. (2014). Recursion: Complexity in cognition. In T. Roeper & M. Speas (Eds.), *Recursion: Complexity in cognition* (pp. ix-xxi). New York: Springer.

これはこの本の序文である。そこで、Speasは明確にembeddingとMergeを区別している。さらに、Mergeがprocedureであることも指摘している。Mergeが二つの要素を併合し、その結果にMergeが再び適用されることも述べてい

る (“to which MERGE can then apply”に注目してほしい)。

In short, the central claim of HCF is not that the language faculty involves embedding some specific XP within another XP. Rather, the claim is simply that the only computational procedure for language is MERGE, which combines two linguistic objects into a labeled linguistic object (to which MERGE can then apply). (pp. xi-xii)

5.2.4 Watumull, J., Hauser, M. D., Roberts, I. G., & Hornstein, N. (2014). On recursion. *Frontiers in Psychology*, 4, 1-7.

Watumullたちは、recursionをembeddingと解釈するのは概念の混同であると述べている。

Consider a recent critique of FLN-style theories: “there is little evidence that unlimited recursion, understood as center-embedding, is typical of natural language syntax. [T]his fits ill with the claim (Hauser et al., 2002) [. . .] that ‘recursion’ (understood as embedding) may be the one crucial domain-specific feature of linguistic ability” (Levinson, 2013: 149, 152). This quote embodies the conceptual confusion discussed in the earlier sections above.

First, as discussed in section Three Properties of Recursion, to understand recursion as embedding is actually to misunderstand recursion: to equate recursion—a property of the generative procedure (applicable to any input)—with possible properties (e.g., embedded structure) of its (potential) output is simply a mathematical error. (p. 4)

Watumullたちは、recursionを“a property of the generative procedure (applicable to any input)”と説明している。procedureという用語が使われていることに注

意してほしい。彼らは, “recursion and embedding are not synonymous” (p. 5)と明言している。

6 なぜHauser et al. (2002)のrecursionがembeddingと誤解されたのか

では, Everett (2005)やPinker & Jackendoff (2005)をはじめ, 多くの学者が recursionをembeddingと間違って解釈したのであろうか。それは, すでに指摘しておいたように, HCFに次のような記述があるからであろう。

The core property of discrete infinity is intuitively familiar to every language user. Sentences are built up of discrete units: There are 6-word sentences and 7-word sentences, but no 6.5-word sentences. There is no longest sentence (any candidate sentence can be trumped by, for example, embedding it in “Mary thinks that . . .”), and there is no non-arbitrary upper bound to sentence length. In these respects, language is directly analogous to the natural numbers . . . (p. 1571)

Natural languages go beyond purely local structure by including a capacity for recursive embedding of phrases within phrases, which can lead to statistical regularities that are separated by an arbitrary number of words or phrases. Such long-distance, hierarchical relationships are found in all natural languages for which, at a minimum, a “phrase-structure grammar” is necessary. It is a foundational observation of modern generative linguistics that, to capture a natural language, a grammar must include such capabilities . . . (p. 1577)

一番目の引用には“Mary thinks that . . .”という補文構造が例として挙げられ

ており、これは初期の生成文法でrecursionと呼んでいたものである。二番目の引用には“recursive embedding of phrases within phrases”という表現がある。この記述を読めば、HCFはrecursionをembeddingと解釈していると思っても仕方がないであろう。

実際、この記述に引きずられてrecursionをembeddingと解釈しているのがMithun (2010)である。

One goal of linguistics is the identification of properties shared by all human languages and no other communication systems. In their much-cited 2002 paper, Hauser, Chomsky, and Fitch proposed a universal: “FLN (the Faculty of Language in the Narrow sense) only includes recursion and is the only uniquely human component of the faculty of language” (2002: 1569). While they provide no explicit definition of recursion, a survey of definitions current in the linguistics and computer science literatures can be found in Parker (2006: 167-190). One that is consistent with much current work in linguistics is that of Pinker and Jackendoff (2005: 4), whereby a recursive structure is characterized as “a constituent that contains a constituent of the same kind”. Recursive structures are indeed pervasive cross-linguistically. A closer look at the variety of such structures that occur, however, indicates that recursion may not be the fixed, fundamental, hard-wired property envisioned. (p. 17)

Mithunはrecursionをembeddingと解釈しているが、その理由は、HCFがrecursionの例としてMary believes that Sという埋め込み文をあげているからであるのは間違いない。次の文からそのことが読み取れる。¹⁸

The single structure cited by Hauser, Chomsky, and Fitch as an example of recursion is complementation: *Mary believes that S*. (p. 18)

しかし、すでに説明しておいたように、この二つの引用文で述べられているのは文の長さのことである。文の長さには限界がないと言っているのであり、recursionそのもののことを言っているのではない。recursive embeddingという表現が使われているが、これはembeddingを繰り返すということである。embeddingを繰り返すのはrecursionの一つの事例にすぎないのである。人間の言語にはrecursionがあるので、このrecursionを使ってembeddingを繰り返したということにすぎない。決してembeddingがrecursionであると言っているわけではない。

recursionがembeddingと誤解されていることは、実は、Chomsky自身よく知っていたことなのである。Chomsky (2015)に次のような記述がある。

The core concept of recursion is often confused in the literature with center-embedding, a very different notion. Sometimes recursion is incorrectly assumed to be necessarily infinite. A recursive function may yield a single output (or nothing). If the output is finite, then the generative mechanism is in effect a list—for language, a list of astronomical size even for fairly simple expressions, not a serious proposal. If the list replaces minimal elements by categories, the same hopeless problem remains. If it introduces phrases, then it is presupposing some generative mechanism, and we are back to generative grammar. (p. 94)

この文章から、Chomskyが、HCFのrecursionはembeddingのことではないと言っているのは明らかであろう。

7 二種類のrecursionを区別している例

7.1 Coolidge, F. L., Overman, K. A., & Wynn, T. (2010). Recursion: What is it, who has it, and how did it evolve? Wiley Online Library.

ここまで見てきたように、HCFのrecursionはMergeの繰り返し適用のことなのであり、recursionをembeddingと解釈する学者はミニマリスト・プログラムとHCFを正しく理解できていないのである。このようにrecursionの定義が混乱しているのであるが、recursionの定義を整理した論文がある。それがCoolidge et al. (2010)である。

この論文は、そのタイトルの“Recursion: What is it, who has it, and how did it evolve?”が示すように、Hauser et al. (2002)の“The faculty of language: What is it, who has it, and how did it evolve?”を意識して書かれたものである。

Coolidge et al. (2010)は論文の冒頭のabstractで次のように述べている。¹⁹

Recursion is a topic of considerable controversy in linguistics, which stems from its varying definitions and its key features, such as its universality, uniqueness to human language, and evolution. Currently, there appear to be at least two common senses of recursion: (1) embeddedness of phrases within other phrases, which entails keeping track of long-distance dependencies among phrases and (2) the specification of the computed output string itself, including meta-recursion, where recursion is both the recipe for an utterance and the overarching process that creates and executes the recipes.

embeddingの意味でのrecursionとある操作を繰り返すという意味でのrecursionがはっきりと区別されている。(1)は生成文法の標準理論時代からある埋め込みのことである。Coolidge et al.もINTRODUCTIONの節でそのことを説明している。

Recursion has often been defined as embedding a phrase within another phrase, although there is great debate about this definition or any other single standard definition. When the term is applied to language, there is a definitional problem from recursion's original use in the fields of mathematics, formal logic, and computer science. In these fields, recursion is any instruction that calls another instruction of like type and which is subject to a terminating condition.

元々、数学や形式論理学やコンピュータ科学では、recursionというのは、ある操作の出力が同じタイプの操作の入力になるということであるが、言語学では句を他の句に埋め込むという意味に変わってしまったということである。

Coolidge et al. はrecursionの定義には統一されたものはないとも述べている。

... there is no single, universally accepted definition of recursion. Its definition varies across disciplines (e.g., mathematics, logic, computer science, and linguistics), and it varies within these disciplines, particularly within linguistics. In fact, much of the current interest and controversy among scholars about recursion stems from its varying definitions and debate about its key features, such as its universality, uniqueness to human language, and evolution.

確かに、HCFはrecursionを明確には定義しなかったが、Chomskyの考えているrecursionは、Coolidge et al. (2010)が³“Chomsky's original notion of recursion as discrete infinity and their mutual idea of recursion enabling the production of an infinite variety of thoughts”と言っているように、ある操作の繰り返し適用のこ

とである。

Coolidge et al. (2010)は、埋め込みをrecursionの定義のstrong versionと呼び、Mergeをweak versionと呼んでいる。以下はstrong versionについての説明である。

Finally, some of the confusion surrounding recursion derives from the two rather different ways it is used in the literature, a strong version and a weak version [or, in the terms used by Van der Hulst, general (generative or *Merge*) and specific (nested and tail)]. The strong version (i.e., which is the most common in the literature) of linguistic recursion is embeddedness, in which the object embedded is a word, phrase, or sentence, and this creates ‘long distance dependencies and the need to keep track, or add to memory’ (Parker, Ref 14, p. 3; also see Van der Hulst, Ref 16).

以下はweak versionのrecursionの説明である。

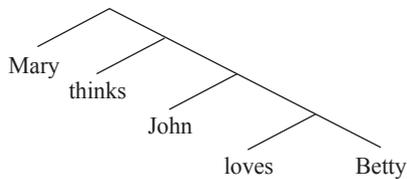
The weak version (i.e., less common) of linguistic recursion is sometimes overlooked in the focus on word, phrase, and sentence embeddedness or skewed by the use of terminology from mathematics and computer science. It is this sense of recursion that specifies the computed output string itself and gives language its power to generate new utterances by enabling the combination of discrete units into larger units (linguists now prefer to label this instruction *Merge*); it also enables specific recursion.

Coolidge et al. (2010)の説明の順序のせいで誤解を生みやすいが、Coolidge et al.とvan der Hulstの用語の関係は次のようになる。²⁰

strong version (embedding) = specific

weak version (Merge) = general

確かに、Mergeがgeneral (weak version)ならばembeddingはspecific (strong version)である。embeddingはMergeの繰り返し適用によって派生される多くの構造の一つにすぎないからである。同じタイプの句を繰り返しMergeすればembeddingとなる。



第一回目のMergeは動詞のlovesと名詞のBettyの間である。第二回目のMergeは名詞のJohnと動詞句のloves Bettyの間である。第三回目のMergeは動詞のthinksと文のJohn loves Bettyの間である。第四回目のMergeは名詞のMaryと動詞句のthinks John loves Bettyの間である。結果として文(John loves Betty)が文(Mary thinks)の中に埋め込まれることになる。

7.2 藤田耕司. (2014b). 投射の進化的問題. 藤田耕司他(編), 『言語の設計・発達・進化—生物言語学探究—』所収 (pp. 279-307). 東京：開拓社.

Mergeが繰り返し適用されることによって埋め込み構造が派生されることは藤田(2014b)でも指摘されている。藤田(2014b)は回帰的Mergeを「派生的回帰性」、自己埋め込みを「表示的回帰性」と呼び、両者を区別している。

回帰性は、Hauser et al. (2002)が人間言語固有の唯一の特性としてと

りあげて以来、再び多くの注目を集めてきた性質であるが、正確にそれが何を指しているのかが研究者の間で見解が一致していなかったこともあり、さまざまな誤解や批判も招いてきた。MPの統語演算理論の観点からは、回帰性とはMergeがその出力に対して無限に繰り返し適用するという操作上の特性（これを「派生的回帰性(derivational recursiveness)」と呼んでおく）であり、Hauser et al.の提案もこの回帰的Mergeのみが言語に固有であるとの主旨で理解すべきである。他方、生成文法の初期から、回帰性を「S内S」や「NP内NP」といった自己埋め込みに代表される構造上の特性（「表示的回帰性(representational recursiveness)」）と見る立場も根強く残っている。表示的回帰性が派生的回帰性によってもたらされる結果の1つに過ぎないことは明らかであって、表示的回帰性が存在しない場合にも、派生的回帰性がなければ（つまりMergeが回帰的に適用しなければ）階層的句構造は原則、生成不可能である。(pp. 282-283)

藤田(2014b)は、注2で、Everettが派生的回帰性と表示的回帰性を混同していると述べている。

Everett (2005)のPirahãには補文構造がないという観察に基づく不適切なUG批判は、この表示的回帰性と派生的回帰性の混同に由来している。この言語の単文を生成する際にも、回帰的Mergeは必要である。一方、非階層的な句構造を生成するには、回帰的Mergeは必要ではなく、実際、そのような非階層的構造も存在する(e.g., *very very very happy*)。したがって人間言語はこのような単純な反復に加えて、回帰的Mergeも備えている点にその独自性があるということになる。(p. 283, 注2)

藤田(2014b)は同じ注2で, Jackendoff (2011)も批判している。

ちなみに, Jackendoff (2011)はここでいう表示的回帰性と派生的回帰性をそれぞれ‘structurally recursive’, ‘formally recursive’と呼んで区別し, 前者のほうが有効な概念であると主張してMPの立場を批判している。これは結局, JackendoffがMPのような派生的文法理論ではなく, 表示のみに基づく文法理論を採用していることの反映でしかないのだが (Culicover and Jackendoff (2005)ほか), この理由によりJackendoffらの‘Simpler Syntax’の主張も実はMPへの正当な批判とはなっていない。(p. 283, 注2)

8 なぜHauser et al. (2002)はrecursionの定義をしなかったのか

HCFがrecursionを明確に定義しなかったことが, recursionをめぐる議論で混乱を引き起こしたと何人もの学者が言っているが, なぜHCFはrecursionを明確に定義しなかったのであろうか。一つ考えられることは, そもそもHCF (あるいはChomsky) にとってrecursionは定義する必要のない当たり前の用語であったのではないかということである。HCFには, 上で指摘したように, (彼らはミニマリスト・プログラムの説明をするとは言っていないが) ミニマリスト・プログラムの説明がある。また, References and Notesにはミニマリスト・プログラムに関する文献が挙げられている。これは, HCFがミニマリスト・プログラムを前提にしているということである。そしてミニマリスト・プログラムでは, Chomsky (1995)ですでに説明されているように, recursionはMergeの繰り返し適用のことである。(Mergeに限定しないのであれば, HCFが言っているrecursionは単にある操作(operation)の繰り返し適用のことにすぎないのである。)

もしrecursionが重要な概念であれば, 当然, HCFも論文の中でrecursionを定義したはずである。定義する必要もない当然の用語なのであろう。²¹

9 まとめ

本稿では、HCFのrecursionの定義についての議論を整理し、HCFはミニマリスト・プログラムを前提としており、HCFの言うrecursionはMergeの繰り返し適用であることを確認し、HCFのrecursionがembeddingであるという解釈は、ミニマリスト・プログラムを理解せず、HCFのrecursionを間違って解釈していることに起因すると結論づけた。

謝 辞

本稿の最終稿を完成させるにあたっては、『同志社大学英語英文学研究』の二名の査読者のコメントが大変参考になった。御礼を申し上げる。

注

1 このHCFの内容の紹介は、中井(2014, pp. 60-66)を短くし、修正を加えたうえでの再掲である。

2 引用文中の下線は原文にはなく、強調のために筆者が加えたものである。以下同じである。

“FLN takes a finite set of elements and yields a potentially infinite array of discrete expressions. This capacity of FLN yields discrete infinity (a property that also characterizes the natural numbers).”の文は、まさに、ミニマリスト・プログラムにおけるMerge（併合）の繰り返し適用による文の派生を説明したものであることは覚えておいてほしい。

“discrete infinity”という表現が何を意味するのかわかりにくいですが、HCFの次の説明が参考になる。

Discrete infinity and constraints on learning. The data summarized thus far, although

far from complete, provide overall support for the position of continuity between humans and other animals in terms of FLB. However, we have not yet addressed one issue that many regard as lying at the heart of language: its capacity for limitless expressive power, captured by the notion of discrete infinity. It seems relatively clear, after nearly a century of intensive research on animal communication, that no species other than humans has a comparable capacity to recombine meaningful units into an unlimited variety of larger structures, each differing systematically in meaning. However, little progress has been made in identifying the specific capabilities that are lacking in other animals.

The astronomical variety of sentences any natural language user can produce and understand has an important implication for language acquisition, long a core issue in developmental psychology. (pp. 1576-1577)

『ランダムハウス英語辞典』(CD-ROM版)によれば、“discrete”という語の定義は「離散的な、相互に区別された」ということであり、“infinity”という語の定義は「無限の範囲 [量, 数], 無辺, 無量, 無数」である。つまり、独立した語(形態素)を組み合わせる無限の長さの、あるいは、無限の数の文を派生するということである。この派生された句や文ももちろんdiscreteなものである。“infinity”を「無限」と解釈するのか「無数」と解釈するか迷うところであるが、Pullum & Scholz (2010)は、まず、Epstein & Hornstein (2004)のdiscrete infinityの説明を引用し、

This property of discrete infinity characterizes EVERY human language; none consists of a finite set of sentences. The unchanged central goal of linguistic theory over the last fifty years has been and remains to give a precise, formal characterization of this property and then to explain how humans develop (or grow) and use discretely infinite linguistic systems. (as cited in Pullum & Scholz, 2010, p. 113)

この引用の直後に、“Here again, ‘discrete infinity’ (which we take to mean denumerable infinity in sets of discrete elements such as symbol strings)” (p. 114)と述べ、自分たちの解釈を示している。

もちろん、一つの文に新しく要素を付加すればより長い文ができ、これを無限に行えば無限の長さの文ができるし、無数の文ができることにはなる。

文のinfiniteudeに関しては、Pullum & Scholz (2010)を参照。

- 3 私の意見(あるいは推測)は、HCFにとってはrecursionはわざわざ定義する必要がない当たり前の用語であるということである。この件に関しては8節で説明する。

- 4 Coolidge et al. (2010)はオンラインの論文でページ番号がない。
- 5 Fitchが共同執筆者の一人であるMartins & Fitch (2014)も, “the term recursion has been used to characterize the process of embedding a constituent inside another constituent of the same kind” (p. 17)と述べ, recursionをembeddingと解釈している。この解釈は, 後に述べるrecursionをMergeの繰り返し適用とするHCFの解釈とは異なる。HCFの共同執筆者であるFitchがこのような解釈をしているのは不思議である。
- 6 確かに, HCFはミニマリスト・プログラムとMergeに明確には言及していない。本文中にはMinimalist ProgramやMergeといった表現は出てこない。しかし, ReferencesにはChomsky (1995)の*The Minimalist Program*が挙げてあり, ミニマリスト・プログラムに関する他の文献も挙げてある。
- 7 引用文中に“non-tail recursive hierarchy”という表現があるが, “tail recursion”という用語はKinsella (2009)による。tail recursionは, 句あるいは文の端(edge)に埋め込みがある場合で, 句や文の左端か右端である。Kinsella (2009)は次のように説明している。

In natural language, representational recursion is exhibited by phrases embedded inside phrases of the same type:

- (1) Mary’s aunt’s brother’s dog’s ball is in the garden.

[*s*[*NP*[*NP*[*NP*[*NP* Mary’s] aunt’s] brother’s] dog’s] ball] is in the garden]

- (2) the man that kissed the girl that John met in the bar that Bill recommended.

[*NP* the man [*CP* that kissed [*NP* the girl [*CP* that John met in [*NP* the bar [*CP* that Bill recommended]]]]]]]

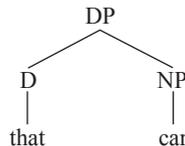
- (3) The mouse the cat the dog chased bit ran.

[*s* The mouse [*s* the cat [*NP* the dog chased] bit] ran]

(1) and (2) exhibit tail recursion, where phrasal embedding occurs at the beginning or end of the sentence or phrase. In (1), the recursion is left-branching, the embedding occurring at the left edge of the sentence. The subject Noun Phrase (NP) contains within it a smaller NP, which contains a smaller NP, which contains a smaller NP, and so on. In (2), the recursive embedding happens at the right-hand edge. The entire NP contains within it a modifying Complementiser Phrase (CP), which itself contains a smaller NP, and that NP contains another smaller CP, and so on. Example (3) exhibits

nested recursion, where the embedding occurs in the middle of the sentence, such that material exists on both sides of the embedded component. Thus, the entire sentence has embedded inside it a smaller one, which has embedded inside it an even smaller one. (p. 114)

- 8 確かに、Al-Mutairiが言うように、HCFで主張されていることがすべてChomskyの意見なのかどうかは不明である。しかし、HCFが前提としている言語理論はChomskyのミニマリスト・プログラムと解釈するのが妥当であると思われる。
- 9 Jackendoff & Pinker (2005)以外にも筆者が目にしたrecursionをembeddingと解釈している例としては、Kinsella (2009), Laury & Ono (2010), Martins & Fitch (2014), Mithun (2010), Sauerland & Trotzke (2011), Verhagen (2010)などがある。Kinsella (2009)は、もともとは彼女の博士論文であるParker (2006)であるが、この本の中でrecursionが詳しく論じられており、recursionを論じる時にはこの本はよく言及される。
- 10 CAとはEverett (2005)のことである。なお、ピダハン語にrecursionがあるかどうかのEverettとNevis et al.の間の論争に関しては中井(2014)を参照。
- 11 recursionはある操作(operation)なりメカニズムを繰り返し適用することであり、recursiveに適用される操作はMergeに限定されるわけではないが、本稿ではミニマリスト・プログラムの話をしているので、recursionはMergeの繰り返し適用と表現しておくことにする。
- 12 本節は中井 (2014), pp. 112-117に修正を加えたうえで再掲したものである。
- 13 Chomsky (1994)に従えば、二つの要素を併合(Merge)してできる句のラベルは主要部(head)と同じである。NPとかVPとかいったラベルはつけない。thatはdeterminerであり、carはNであり、that carはDP (Determiner Phrase)であるので以前の方法ならば次のように表示される。



DPやNPといったラベル(label)をつけないのは、Bare Phrase Structureの考え方に従っているからである。that carという句は、Determiner Phraseであるが、句の統語範疇は主要部であるthatの特性を受け継いでいるので、Mergeによって派生された句の

ラベルはthatである。この後も、Mergeの結果できる句の表示(label)はその句の主要部の語彙項目である。なぜこのように表示するのかの説明は省略するが、興味のある方はChomsky (1994)を読んでいただきたい。また、ラベル付けに関する問題については、Fujita (2014a)、藤田(2014b)等を参照。

14 “third factor”に関しては、Chomsky (2005)の以下の説明を参照。

Assuming that the faculty of language has the general properties of other biological systems, we should, therefore, be seeking three factors that enter into the growth of language in the individual:

1. Genetic endowment, apparently nearly uniform for the species, which interprets part of the environment as linguistic experience, a nontrivial task that the infant carries out reflexively, and which determines the general course of the development of the language faculty. Among the genetic elements, some may impose computational limitations that disappear in a regular way through genetically timed maturation. Kenneth Wexler and his associates have provided compelling evidence of their existence in the growth of language, thus providing empirical evidence for what Wexler (to appear) calls “Lenneberg’s dream.”

2. Experience, which leads to variation, within a fairly narrow range, as in the case of other subsystems of the human capacity and the organism generally.

3. Principles not specific to the faculty of language.

The third factor falls into several subtypes: (a) principles of data analysis that might be used in language acquisition and their domains; (b) principles of structural architecture and developmental constraints that enter into canalization, organic form, and action over a wide range, including principles of efficient computation, which would be expected to be of particular significance for computational systems such as language. It is the second of these subcategories that should be of particular significance in determining the nature of attainable languages. (p. 6)

15 HCFとPinker & Jackendoffの間ではHCFの主張をめぐって論争が行われた。以下の論文でその論争が行われている。

Hauser, M. D., Chomsky, N., & Fitch, W. T. (2002). The faculty of language: What is it, who has it, and how did it evolve? *Science*, 298, 1569–1579.

- Pinker, S., & Jackendoff, R. (2005). The faculty of language: what's special about it? *Cognition*, 95, 201-236.
- Fitch, W. T., Hauser, M. D., & Chomsky, N. (2005). The evolution of the language faculty: Clarifications and implications. *Cognition*, 97, 179-210.
- Chomsky, N., Hauser, M. D., & Fitch, W. T. (2005). Appendix. The minimalist program. (Appendix to Fitch et al. (2005))
- Jackendoff, R., & Pinker, S. (2005). The nature of the language faculty and its implications for evolution of language. *Cognition*, 97, 211-225.
- 16 Chomsky et al. (2005)はオンラインで公開されているのでページ番号はない。
- 17 transferが何を意味するかはここでは関係がないのでtransferの説明は省略する。
- 18 HCFが挙げている例は，“Mary believes that S”ではなく，“Mary thinks that . . .”である。
- 19 本節の引用はオンラインの論文からの引用であるのでページ番号はない。
- 20 van der Hulst, H. (2010b)はspecific recursionとgeneral recursionを次のように説明している。

A factor that must be recognized in evaluating the ‘clash’ between the rarity of recursion and its alleged central role in language is that studies of the actual occurrence of recursion focus on *specific* recursion while Chomsky’s more recent claims about the centrality of recursion seem to emphasize *general* recursion, i.e. hierarchical grouping. (p. xxxiii)

- 21 ちなみに、Chomsky (1995)のindexにはrecursionという項目はない。

参考文献

- Al-Mutairi, F. R. (2014). *The minimalist program: The nature and plausibility of Chomsky’s biolinguistics*. Cambridge: Cambridge University Press.
- Chomsky, N. (1994). *Bare phrase structure*. MIT Occasional Papers in Linguistics, No. 5.
- Chomsky, N. (1995). *The minimalist program*. Cambridge, Mass.: MIT Press.
- Chomsky, N. (1998). *Minimalist Inquiries: The Framework*. MIT Occasional Papers in Linguistics, No. 15.
- Chomsky, N. (2005). Three factors in language design. *Linguistic Inquiry*, 36, 1-22.

- Chomsky, N. (2007). Approaching UG from below. In U. Sauerland & H.-M. Gärtner (Eds.), *Interfaces + recursion = Language?: Chomsky's minimalism and the view from syntax-semantics* (pp. 1-29). Berlin: Mouton de Gruyter.
- Chomsky, N. (2014). Minimal recursion: Exploring the prospects. In T. Roeper & M. Speas (Eds.), *Recursion: Complexity in cognition* (pp. 1-15). New York: Springer.
- Chomsky, N. (2015). Some core contested concepts. *Journal of Psycholinguistics Research*, 44, 91-104.
- Chomsky, N., Hauser, M. D., & Fitch, W. T. (2005). Appendix. The minimalist program. (Appendix to Fitch et al. (2005)). Retrieved from <http://arti.vub.ac.be/cursus/2005-2006/mwo/03b-mp.pdf>.
- Coolidge, F. L., Overman, K. A., & Wynn, T. (2010). Recursion: What is it, who has it, and how did it evolve? Wiley Online Library. Retrieved on September 8, 2014 from <http://wires.wiley.com/cogsci>.
- Epstein, S., & Hornstein, N. (2004). Letter on 'The future of language'. *Language*, 81, 3-6.
- Everett, D. L. (2005). Cultural constraints on grammar and cognition in Pirahã: Another look at the design features of human language. *Current Anthropology*, 46, 621-646.
- Everett, D. L. (2007). Challenging Chomskyan linguistics: The case of Pirahã. *Human Development*, 50, 297-299.
- Fitch, W. T. (2010). Three meanings of "recursion": Key distinctions for biolinguistics. In R. K. Larson, V. Deprez, & H. Yamakido (Eds.), *The evolution of human language: Biolinguistic perspectives* (pp. 73-90). Cambridge: Cambridge University Press.
- Fitch, W. T., Hauser, M. D., & Chomsky, N. (2005). The evolution of the language faculty: Clarifications and implications. *Cognition*, 97, 179-210.
- Fujita, K. (2014a). Recursive merge and human language evolution. In T. Roeper & M. Speas (Eds.), *Recursion: Complexity in cognition* (pp. 243-264). New York: Springer.
- 藤田耕司. (2014b). 投射の進化的問題. 藤田耕司他(編), 『言語の設計・発達・進化—生物言語学探究—』所収 (pp. 279-307). 東京: 開拓社.
- Hauser, M. D., Chomsky, N., & Fitch, W. T. (2002). The faculty of language: What is it, who has it, and how did it evolve? *Science*, 298, 1569-1579.
- Hornstein, N., Nunes, J., & Grohmann, K. K. (2005). *Understanding minimalism*. Cambridge: Cambridge University Press.
- Jackendoff, R. (2011). What is the human language faculty?: Two views. *Language*, 87, 586-624.
- Jackendoff, R., & Pinker, S. (2005). The nature of the language faculty and its implications for evolution of language. *Cognition*, 97, 211-225.

- Kinsella, A. R. (2009). *Language evolution and syntactic theory*. Cambridge: Cambridge University Press.
- Kotowski, S., & Härtl, H. (2008). Recursion and the language faculty: Over the evolution of the concept in generative grammar. Retrieved from http://www.uni-kassel.de/fb02/fileadmin/datas/fb02/Institut_f%C3%BCr_Anglistik_Amerikanistik/Dateien/Linguistik/Kotowski_H%C3%A4rtl_NLK_Proceedings_Recursion.pdf.
- Laury, R., & Ono, T. (2010). Recursion in conversation: What speakers of Finnish and Japanese know how to do. In H. van der Hulst (Ed.), *Recursion and human language* (pp. 69-91). Berlin: De Gruyter Mouton.
- Lowenthal, F., & Lefebve, L. (Eds.). (2014a). *Language and recursion*. New York: Springer.
- Lowenthal, F., & Lefebve, L. (2014b). Are there any relations between recursion and language? In F. Lowenthal & L. Lefebve. (Eds.), *Language and recursion* (pp. v-vii). New York: Springer.
- Martins, M. D., & Fitch, W. T. (2014). Investigating recursion within a domain-specific framework. In F. Lowenthal & L. Lefebve (Eds.), *Language and Recursion* (pp. 15-26). New York: Springer.
- Mithun, M. (2010). The fluidity of recursion and its implications. In H. van der Hulst (Ed.), *Recursion and human language* (pp. 17-41). Berlin: De Gruyter Mouton.
- 中井悟. (2014). ビダハン論争をめぐるって. 『同志社大学英語英文学研究』, 92号, 57-152.
- Nevins, A., Pesetsky, D., & Rodrigues, C. (2009). Pirahã exceptionality: A reassessment. *Language*, 85, 355-404.
- Parker, A. R. (2006). *Evolution as a constraint on theories of syntax: The case against minimalism*. Doctoral dissertation. University of Edinburgh.
- Pinker, S., & Jackendoff, R. (2005). The faculty of language: what's special about it? *Cognition*, 95, 201-236.
- Pullum, G. K., & Scholz, B. C. (2010). Recursion and the infinitude claim. In H. van der Hulst (Ed.), *Recursion and human language* (pp. 113-137). Berlin: De Gruyter Mouton.
- Roeper, T. (2011). The acquisition of recursion: How formalism articulates the child's path. *Biolinguistics*, 5, 57-86.
- Roeper, T., & Speas, M. (Eds.). (2014). *Recursion: Complexity in cognition*. New York: Springer.
- Sauerland, U., & Gärtner, H.-M. (Eds.). (2007). *Interfaces + recursion = Language?: Chomsky's minimalism and the view from syntax-semantics*. Berlin: Mouton de Gruyter.
- Sauerland U., & Trotzke, A. (2011). Biolinguistic perspectives on recursion: Introduction to the special issue. *Biolinguistics*, 5, 1-9.

- Speas, M. (2014). Recursion: Complexity in cognition. In T. Roeper & M. Speas (Eds.), *Recursion: Complexity in cognition* (pp. ix-xxi). New York: Springer.
- Tomalin, M. (2007). Reconsidering recursion in syntactic theory. *Lingua*, 117, 1784-1800.
- van der Hulst, H. (Ed.) (2010a). *Recursion and human language*. Berlin: De Gruyter Mouton.
- van der Hulst, H. (2010b). Re Recursion. In H. van der Hulst (Ed.), *Recursion and human language* (pp. xv-liv). Berlin: De Gruyter Mouton.
- Verhagen, A. (2010). What do you think is the proper place of recursion?: Conceptual and empirical issues. In H. van der Hulst (Ed.), *Recursion and human language* (pp. 93-110). Berlin: De Gruyter Mouton.
- Watumull, J., Hauser, M. D., Roberts, I. G., & Hornstein, N. (2014). On recursion. *Frontiers in Psychology*, 4, 1-7.
- Zwart, J.-W. (2011). Recursion in language: A layered-derivation approach. *Biolinguistics*, 5, 43-56.

『新英和大辞典』（第6版, CD-ROM版）東京：研究社.

Oxford Advanced Learner's Dictionary (8th edition) (CD-ROM版).

『ランダムハウス英語辞典』（CD-ROM版）東京：小学館.

Synopsis

Some Remarks on “Recursion”

Satoru Nakai

Since the publication of “The faculty of language: What is it, who has it, and how did it evolve?” by Marc D. Hauser, Noam Chomsky, and W. Tecumseh Fitch in 2002, in which they propose the hypothesis that “FLN [faculty of language in the narrow sense] comprises only the core computational mechanisms of recursion” (p. 1573), there have been lively debates on recursion, but the debates have been confused because Hauser et al. did not define the term “recursion” explicitly.

There have been two interpretations of the term “recursion.” One is that recursion is self-embedding. That is, a constituent is embedded in another constituent of the same kind (e.g., noun phrases in noun phrases or sentences in sentences). The other is that recursion is an iterative application of an operation (typically Merge in Minimalist Program).

Though Hauser et al. did not define the term “recursion” explicitly, the careful reading of the article leads us to realize that the article is written within the theoretical framework of Minimalist Program and therefore the recursion in the article should be interpreted as the iterative application of Merge. Chomsky explains Merge with this concept of recursion in mind in his *The Minimalist Program*, which was published in 1995 and the recursion in this sense was common among generative linguists when the article by Hauser et al. was published in 2002. Those who have taken it that the recursion in Hauser et al. (2002) is self-embedding have failed to understand what Minimalist Program is

and what Hauser et al. (2002) mean by “recursion.”