Korean Journal of English Language and Linguistics, Vol 26, January 2026, pp. 175-197
DOI: 10.15738/kjell.26..202601. 175

KOREAN JOURNAL OF ENGLISH LANGUAGE AND LINGUISTICS

ISSN: 1598-1398 / e-ISSN 2586-7474

http://journal.kasell.or.kr

Separate Dimensions Associated with Functional Head*

Ji-hyun Sung (Dongwon Institute of Science and Technology University)

Received: December 9, 2025
Revised: January 11, 2026
Accepted: January 17, 2026

Sung, Ji-hyun
Lecturer, Department of Nursing at Dongwon Institute of Science and Technology
Email: rebeccasung279@gmail.com

* This work was supported by the Ministry of Education of the Republic of Korea and the National Research Foundation of Korea (NRF-2023S1A5B5A17086566).

ABSTRACT

Sung, Ji-hyun. 2026. Separate dimensions associated with functional head. *Korean Journal of English Language and Linguistics* 26, 175-197.

This study investigates the concept of Separate Dimensions (SD) as discussed in Chomsky (2021), focusing on their nature, generation, and compatibility with the Strong Minimalist Thesis (SMT) and Computational Efficiency (CE). It addresses how asymmetric syntactic complexes can optimally integrate into the primary workspace. The working hypothesis proposes that Head mediates between primary workspace operations and SD creation. The paper analyzes the combination of syntactic objects with SDs generated by Heads and examines the principles linking SDs back to the main workspace. Building on Chomsky's (2019b) concepts of disjunction and conjunction, disjunction is treated as Head-Adjunction and NP formation through adjectival modification within one SD. Conjunction ($n \geq 2$) includes multiple adjectives modifying a noun with referential similarity, as well as relative clauses linking to the stem structure. The analysis follows criteria for computational legitimacy, no violation of the NTC (No Tampering Condition) or ICC (Inclusiveness Condition), valid SD generation, and specific stem-structure linkage. Head-Adjunction is analyzed as the stage where √Root is placed in the workspace. Disjunction (n=1, single SD) examines noun phrases modified by adjectives and verb phrases modified by adverbs. Conjunction analyzes noun phrases modified by multiple adjectival phrases and relative clauses having a separate dimension through adverbial modification, aiming to elucidate the stages of full sentence formation.

KEYWORDS

computational efficiency, Head, separate dimension, form set, disjunction, conjunction, referential similarity, modification relation.

1. Introduction

The central principles of the Minimalist Program lie in simplifying syntactic operations and maximizing their efficiency. Chomsky (2000, 2001) explicitly argues that the reason complex and heterogeneous syntactic computations must be reduced is that the evolution of the human language faculty in the brain cannot track such complexity. In line with this minimalist orientation, Chomsky (2019b) further minimizes the stock of syntactic operations and aims to maximize the use of a small set of indispensable mechanisms in order to optimize Computational Efficiency (CE). Chomsky (2019a) and Chomsky et al. (2019) claim that this basic and indispensable mechanism is Merge. On the basis of the proposal of capital MERGE in Chomsky (2019a, b) and Chomsky et al. (2019), the core principles of Merge are redefined. Chomsky (2019a) proposes six desiderata[1] for the adequacy of natural language and the efficiency of syntax and maintains that only Internal Merge and External Merge satisfy these conditions. This line of argumentation is developed in detail and reiterated in Chomsky (2021). Within Merge, there is Set-Merge, which yields symmetric configurations, and Pair-Merge, which yields asymmetric configurations. Such asymmetric syntactic objects are generally taken to lower the efficiency of computation. The desiderata on Merge and the basic principles of simple Merge are straightforwardly applicable to Set-Merge, but they give rise to unresolved issues when extended to Pair-Merge. Chomsky (2019b, 2021) therefore suggests that, while Pair-Merge is subject to the same principles of Merge, it is executed in a Separate Dimension (SD)[2].

The present study investigates the notion of SD as discussed in Chomsky (2021). What exactly is an SD? How and why is it generated? Is it compatible with the Strong Minimalist Thesis (SMT) and with CE? If it were not compatible, how could the asymmetric complexes that we find in natural language be optimally integrated into the primary workspace? This paper addresses these questions. The working hypothesis is that Head mediates between the operations of the primary workspace and the creation of an SD. More specifically, the paper analyzes how syntactic objects combine with an SD generated by a categorizer, and by what principles this SD is linked back to the main workspace. Section 2 demonstrates the existence of Separate Dimensions and explores their directional properties. Section 3 reviews analyses by other scholars, and based on these, Section 4 systematically forms the concrete generation of Separate Dimensions and their integration with the main sentence in a step-by-step manner. Section 5 presents the conclusion.

2. The Proofs of 'Separate Dimensions'

2.1 What is a 'SD'?

The idea of 'SD' can be traced back to Chomsky's (2004) notion of a "separate plane." Consider the following

[1] Japanese scholars analyze these desiderata as seven considerations, whereas Chomsky (2019a) mentions two conditions and six desiderata. Among the two conditions, the first is the 'third factor' principle—one of the three factors of language—which addresses how computational operations should behave given the complexity of computation. The second concerns the language-specific property of being an 'organic system,' which is analyzed as 'strict binarity,' apparently yielding "seven desiderata" to be respected in the application of capital MERGE.

[2] One reviewer raises a question regarding the distinction between *separate planes* and *separate dimensions*. In the present study, these two notions are understood by the author to be conceptually equivalent, and they are therefore treated as referring to the same analytical construct. A more fine-grained examination of potential differences between these notions, and of whether they should be formally distinguished within the theory, is left for future research.

remark on adjunction and Pair-Merge in Chomsky (2004, pp. 117-118).

(1) Given the basic properties of adjunction, we might intuitively think of α as attached to β on a separate plane, with β retaining all its properties on the "primary plane," the simple structure.

In the derivation of a sentence, the construction of the basic clausal structure proceeds on the primary plane, while adjunction, where the order in which α is adjoined to β is fixed, takes place on a "separate plane." In Chomsky (2004), this "separate plane" is what is later referred to as a "Separate Dimension" in Chomsky (2019b, 2021). The "primary plane" can be understood as the workspace in which syntactic computation proceeds. One might then ask whether "a separate plane" should be identified with another workspace, but this does not seem to be the case. Chomsky (2019a, b) explicitly rejects operations such as Parallel Merge and multi-dominance structures, which presuppose multiple workspaces, on the grounds that they are incompatible with both the SMT and CE. Komachi et al. (2019) likewise argue that extended forms of Merge such as Parallel Merge, Sideward Merge, and Late Merge cannot be derived from the concept of capital MERGE, since MERGE is constrained by third-factor principles and derivational determinacy, which exclude such operations. In other words, the syntactic computation that underlies the formation of a single sentence must remain within a single original workspace. Taken together, various authors' discussions, and Chomsky (2004, 2019a, 2019b, 2021) suggest that Merge-types like Parallel Merge are problematic precisely because they illegibly introduce multi-dimensionality, whereas Pair-Merge is instead executed in an SD. This insistence, however, is vague. This paper attempts to clarify this point. The claim advanced here is that, once the full course of syntactic computation is taken into account rather than merely considering Merge, the existing assumption effectively entails the existence of SDs.

2.2 Grounds for Positing SDs

2.2.1 Lexicon

The Lexicon is the mental repository of words stored in the brain. Each word that can serve as a component of a sentence corresponds to a lexical item (LI). If we regard the step of taking LIs from the mental lexicon and placing them into the workspace for sentence formation as defining the initial workspace, then the Lexicon and the workspace may at first glance appear to constitute SDs. Chomsky (2019a) assumes that MERGE can access both the Lexicon and the workspace:

(2) The 'search space' in IM is limited to the local phrase maker under consideration, in EM the search space is potentially the entire Lexicon (Chomsky 2019a, p. 273)

External Merge is the operation that takes an LI from the Lexicon and merges it into the workspace. In such cases, MERGE has access to both the Lexicon and the workspace. If this assumption is correct, MERGE ranges over what look like different dimensions, namely the Lexicon and the workspace. If, alternatively, we assume that all LIs required for the derivation are already present in the workspace[3], then External Merge can apply in principle

[3] This is the case where the Lexicon is identical to the initial workspace. Many scholars do not explicitly identify W_1 with the Lexicon, although they presuppose that all LIs necessary for the syntactic derivation are present in W_1. (Gallego and Chomsky 2019; Fong et al. 2019). Chomsky's (2019a, p. 273) reference to 'the entire Lexicon' is analyzed as distinct from W_1, and

within a single dimension. Even so, the initial workspace is still derived from the Lexicon. Considering both scenarios, the conclusion is that the Lexicon and the workspace do not constitute "the same dimension" in the relevant technical sense.

2.2.2 Pair-Merge

Chomsky (2019a) assumes that, given a workspace W_1 with a numeration $\sum = (X_1, X_2, \ldots, X_n)$, applying MERGE($\sum$) yields a new workspace $W_2 = \{\{X_1, X_2\}, X_3, \ldots, X_n\}$, where X_1 and X_2 are simply replaced by the set $\{X_1, X_2\}$. This is a case of simple Set-Merge, and the update of the workspace proceeds straightforwardly. By contrast, it is not reasonable to assume that an ordered pair $<X_1, X_2>$, as in Pair-Merge, can be substituted in the same way as the unordered set $\{X_1, X_2\}$. To see this more concretely, note that in the case of Set-Merge, even after X_1 and X_2 are replaced by $\{X_1, X_2\}$, MERGE can still access the individual terms X_1 and X_2 inside $\{X_1, X_2\}$ in the updated workspace W'. In the case of Pair-Merge, however, once the workspace is updated with $<X_1, X_2>$, MERGE can no longer access X_1 and X_2 individually; the only accessible syntactic object is the pair $<X_1, X_2>$ itself. This alone shows that simple replacement in the updated workspace behaves differently in Set-Merge and Pair-Merge. Otherwise, the mode of replacement would be identical, and the difference would have to be reduced solely to a difference in visibility for MERGE in the updated workspace. The conclusion, then, is that Pair-Merge inherently involves an SD. This, arguably, is the reason why Chomsky (2004) speaks of a "separate plane," and why Chomsky (2019b, 2021) recasts this in terms of an SD: adjunction as implemented by Pair-Merge is realized in a dimension distinct from the primary workspace.

2.2.3 Form Copy

Consider the example in Chomsky (2021, p. 16):

(3) Lex = {p, q, r, …, ¬, ∨}
W = p, ¬p, ((¬p) ∨ q)

Chomsky (2021) suggests that, when syntactic objects of the same form 'p' arise during the course of Merge as in (3), the interface systems, in the interest of derivational simplicity, initially treat them as three tokens of the same object. It then becomes necessary to determine whether these are in fact copies of a single object or distinct objects akin to homophones. While respecting the fundamental No Tampering Condition on structure-building, the SM (sensorimotor) interface imposes a linear order, and the CI (conceptual–intentional) interface establishes Form Copies to secure an unambiguous interpretation. The process by which copies are related within a phase in order to form a chain for linearization is what is referred to as Form Copy (FC). Chomsky (2021) argues that FC obeys the stability requirement among the desiderata on Merge, and furthermore that FC constitutes a syntactic operation compatible with third factor considerations. Under the condition on $\sum$[4], structurally matching elements are grouped

---

Chomsky (2021) presupposes that all LIs are in W_1.

[4] In Chomsky (2021), the phrase "under the condition on $\sum$" is interpreted here as referring to the application of syntactic principles. In mathematics, $\sum$ denotes the sum of a sequence. More specifically, it is analyzed as the condition that all syntactic principles applied to W_n of degree *n*—including the Strong Minimalist Thesis (SMT) and the desiderata that Merge must satisfy—should be obeyed. It thus represents the conditions governing the application of syntactic operations to W_n as a whole.

as an ordered pair <X, X> by Minimal Search. Chomsky claims that FC operates under conditions distinct from those governing structure-building Merge. The process is exemplified in detail by (4).

(4) a. Bill, John met yesterday
b. [X_4 [$John_2$ [INFL [X_3 [$John_1$ [v* [X_2 [meetX_1 yesterday]]]]]]]
c. i (X4, X3) ii (X3, X2) iii (X2, X1) iv (John2, John1) Kitahara (2024, pp. 199-200)[5]

Kitahara (2024) shows that X is taken to refer to Bill, while FC regards two structurally identical inscriptions in a c-command configuration as copies for the purposes of CI interpretation. The question then arises: where does the resulting ⟨X, X⟩ reside? Even if FC remains visible to the interfaces, it must occupy a different or separate dimension from the workspace in which Merge applies. Moreover, FC formation is regulated by Minimal Search and yields an ordered pair of the Pair-Merge type. However, FC appears to be distinct in character from the SD associated with Pair-Merge, as can be seen from Chomsky's (2021, p. 20) discussion.

(5) FC is not subject to conditions that hold for the structure-building operation Merge. We expect, then, to find configurations that are subject to FC but not to Merge, …

Unlike Pair-Merge, FC is a process geared to interface interpretation, specifically to sentence linearization. Pair-merged syntactic objects can feed further operations in the workspace, but FC-configurations such as <$John_1$, $John_2$> in (4) do not participate in subsequent applications of Merge. Once checking is complete, FC can be viewed as a device that guarantees that no interpretive problem arises at the interfaces.

2.2.4 M-gap

The workspace prior to an update and the workspace after that update are separated by what may be called an M-gap, which, on the present view, constitutes a kind of SD. If two syntactic objects a and b are merged in a given workspace, that workspace is updated, and the original a and b are now represented as {a, b} in the updated workspace. That is, in W_1 they exist as a and b, while in the updated W_2 they exist as {a, b}. In this sense, W_1 and W_2 should be regarded as SDs. The members a and b inside {a, b} in W_2 remain accessible to subsequent syntactic operations, but the occurrences of a and b in W_1 are no longer accessible. This is the Markovian nature of the workspace. FC, however, can access both W_1 and W_2, and can pair a in W_1 with $\underline{a}$ inside {a, b} in W_2 as <a, $\underline{a}$>. The difference between these two stages of the workspace is what is referred to as the M-gap. Chomsky (2019a) analyzes the situation in such a way that the original a and b in W_1 exist as {a, b} in W_2, and the tokens of a and b in W_1 are deleted and no longer present in W_2. The requirement that the properties of a and b be preserved across this update underlies the consideration of stability among the desiderata on Merge. If this process is treated as an update, then W_1 and W_2 constitute a syntactic development that imposes no additional computational burden. The different dimension corresponding to the M-gap, in other words, does not itself incur any extra computational cost, and the same holds for SDs more generally.

[5] I have modified examples (1) and (2) from Kitahara (2024) and present them here as example (4). The labels A and A' indicated in examples (1) and (2) of Kitahara (2024) are not relevant to the present argument and are therefore omitted.

2.3 The Orientation of Separate Dimensions

As seen above, the notion of an SD is already implicit in the architecture of syntactic computation. If this assumption is correct, then an SD associated with Pair-Merge is entirely plausible. The question is what constraints govern the creation of such SDs.

First, and most crucially, the formation of an SD must not interfere with Merge operations in the primary workspace. Rather, it should serve to enhance the visibility of relevant material for Merge. Shim (2022, p. 63) argues that, in the case of Pair-Merge, the adjoined syntactic object is treated as invisible to the operation of Merge. If, as Chomsky (2019b, 2021) proposes, Pair-Merge applies in an SD, then the adjoined material is invisible to Merge, and the computation that proceeds in the main workspace becomes more transparent, thereby increasing CE.

Second, the SD must be smaller in size than the primary workspace; it can never exceed the size of the existing workspace. Fong et al. (2019), in discussing Markovian workspaces compatible with Merge, address the size of the workspace. They assume that all terms necessary for clausal structure building are initially present in W_1, and that the workspace is updated by successive applications of Merge. On their view, the size of the workspace is the sum of (i) syntactic objects that have not yet been merged and (ii) all merged syntactic objects that remain accessible to further operations. Chomsky (2019b, 2021) maintains that Merge always proceeds in a strictly binary fashion and that the number of workspaces must not proliferate; there must be exactly one. In other words, the size of the workspace is a crucial parameter for the operation of the Merge system, and the creation of an SD potentially impacts that size. Pan and Du (2024) term the SDs that arise in syntactic computation "derivational dimensions," explicitly distinguishing them from workspaces and from phases. They argue that such dimensions are generated only within the limits of working memory in the brain. Although there is room for debate about how to treat syntactic objects that remain accessible in an updated workspace, it is clear that no SD in which Pair-Merge applies may exceed the size of the original workspace, on pain of violating CE.

Third, SDs can in principle be defined for all LIs, but their creation is optional. Chomsky (2019b, 2021) classifies Pair-Merge as an instance of the language faculty's third factor. Concretely, this means that it is regulated by the general principles governing language—often schematized as the seven desiderata. These principles must be respected, though their concrete implementation may vary across particular languages. The core principles of Merge itself, however, remain uniform.

3. Other Analyses of 'Separate Dimensions'

As noted above, the SD associated with Pair-Merge should (i) enhance CE, (ii) increase the visibility of relevant material for Merge, (iii) be contained within the size of the existing workspace, and (iv) qualify as a third-factor property of language. Chomsky (2019b) proposes that such Pair-Merge configurations are formed in an SD and that the categorizer functions as the link that connects the SD to the stem structure. The key difference between multi-dominance and SD is that multi-dominance is defined over nodes within a single-dimensional tree, whereas SD is a distinct locus in which complex configurations are handled independently. Clarifying the principles governing such an SD is essential to establishing the legitimacy of syntactic derivations, precisely because doing so amounts to articulating the principles of CE. The following sections review other authors' analyses of SD in more detail.

3.1 Chomsky's (2019b) Analysis of Conjunction and Disjunction

Chomsky (2019b, p. 50) argues that Pair-Merge can adjoin material to a host not only independently but also in such a way that multiple dimensions are created, with the adjoined material residing in SDs and hence remaining invisible to the labeling algorithm. In these adjunct structures, each element is anchored in the host it is linked to; put differently, it is the host that can have SDs. What, then, serves as this host-like link? Chomsky (2019b) identifies the link as a categorial head (categorizer): n as the link for nominal projections and v as the link for verbal projections. Supporting the Root-categorization approach of Borer and Marantz, Chomsky (2019b) holds that Roots in the lexicon are category-neutral, and that their categorial status depends on the categorizer– n, v, or a –with which they combine. He further contends that traditional head-movement of Root R into a categorizer is misguided, since, on the Borer (2014) view, it is the categorizer (the link) that selects the Root, and its categorial specification (n, v, a) is determined by context. Chomsky (2019b) distinguishes four types of categorial heads according to whether the resulting syntactic object bears substantive and/or predicative properties. These four heads—n, v, a, and p—can all function as links and, moreover, are taken to serve as phase markers.

(6) [+S, -P] = noun
[-S, +P] = verb
[+S, +P] = adjective
[-S, -P] = preposition

Chomsky (2019b) claims that, together with C, the categorial heads n and v function as links that define phases, and that RP[6] attaches to these links, which in turn assign a category to the RP. The issue of how categories are assigned to Roots will be addressed in more detail in Section 4. Chomsky (2019b) further argues that adjunction structures formed by Pair-Merge come in two types: a conjunctive type and a disjunctive type. Conjunctive adjunction involves an unbounded, unstructured sequence of elements anchored in a common link, whereas disjunctive adjunction involves only a single element combining with a given link.

3.1.1 Disjunctive adjunction

In Chomsky's (2019b) two-way classification of adjunction as conjunction vs. disjunction, the disjunctive type is the case where exactly one lexical item adjoins in an SD. More concretely, disjunction is the configuration in which there is a single sequence, and hence just one SD. This sequence is attached to a link, and the link determines the category of the lexical item; the entire sequence associated with that category then participates in the subsequent derivation. Since MERGE cannot see the adjoined material in the SD, but only the link (i.e. the categorial head), there is therefore no violation of the Resource Restrictions in capital MERGE. Chomsky (2019b) analyzes the sentence "books fell" such that the Root 'book' is Pair-Merged with n in an SD, yielding 'books' as

[6] RP (Root Phrase) refers to the combination of a category-neutral Root with a head such as *N, V*, or *A*. As a concrete illustration, let us examine the derivation of *books*.

a. {N, book}
b. {N*$_{plural}$, {N, book}}

The author assumes that *books* is derived through the stages illustrated above. The Root *book* combines with a head, and the plural feature of N is interpreted at the interface as *books*. A more detailed discussion of this derivational process is provided in Section 4; see also footnote 7.

an NP[7] when the Root is brought into the derivation.

3.1.2 Conjunctive adjunction

Conjunctive adjunction is the configuration in which two or more sequences are attached in an SD. These sequences take the following form:

(7) Disjunction: $n = 1 \rightarrow \langle S_1, L_1 \rangle$
Conjunction: $n \geq 2 \rightarrow \langle CONJ, \langle S_1, L_1 \rangle \ldots \langle S_n, L_n \rangle \rangle$

Each sequence Pair-Merges with a common link L, and L represents the shared element among them. The sequences are generated simultaneously and are predicated of the link. For instance, for a phrase like "the young happy eager man", one may assume that the SD builds a structure of the following type:

(8) Disjunction: $\langle the_{S1}, man_{L1} \rangle$
Conjunction: $\langle \langle young_{S1}, man_{L1} \rangle \wedge \langle happy_{S2}, man_{L2} \rangle \wedge \langle eager_{S3}, man_{L3} \rangle \wedge man_{L=N^*(The)} \rangle$

That is, each sequence is generated in parallel by virtue of the link, but the actual attachment to the link proceeds in the order S_1, S_2, S_3, and this ordering is likewise determined by the link. While these conjunctive adjunction structures are formed in an SD, what is visible to the main syntactic computation is only the NP that serves as the link. Consequently, no Resource Restriction is violated.

The problem, however, is that this kind of sequence-based pairing cannot be justified mathematically. Marcolli et al. (2025) show that sequence-type Pair-Merge is not mathematically well-defined. In the disjunctive case, the categorial head effectively "sticks" to the lexical item, which is analogous to adding an index; as will be shown below, this runs afoul of both the No Tampering Condition (NTC) and the Inclusiveness (ICC). In sequence-type Pair-Merge, the output of Merge is defined not as a set but as an ordered pair. Thus, when a categorial head K merges with a lexical item L, the resulting object is taken to be either <K, L> or <L, K>, depending on the assumed direction of Merge. Crucially, these two outcomes are formally distinct objects. This assumption immediately raises a mathematical concern: for a formal operation to be well-defined, identical inputs must yield a unique output. However, given the same input set {K, L}, sequence-type Pair-Merge permits multiple non-equivalent outputs, with no intrinsic criterion within the operation itself to determine which ordering should be selected. Since the ordering information is not present in the input prior to Merge, it must be introduced during the derivation,

[7] Whether nominal phrases should be analyzed as NPs or DPs falls outside the scope of this paper and is therefore set aside here. I notate them as NPs because I assume that determiners should be treated as one of the properties of NP. The following presents Oishi's (2015) analysis of Root categorization, applying Chomsky's framework. The author regards both '*the*' in '*the book*' and the plural marker '*-s*' in '*books*' as properties of NP. Chomsky (2007) argues that there are two nominal elements: N* for definite nominals and N for indefinite nominals. Furthermore, definiteness is distinguished by the presence of an element D and the absence of the element. Accordingly, a definite nominal phrase bears D within it, whereas an indefinite nominal phrase lacks the element. A head N* takes another head D, which in turn takes an undifferentiated nominal Root N as its complement as illustrated below.
a. {the, book}
b. {N*, {the, book}}
c. {N*, {book, {the, book}}}
d. {<N*, the>, {book, {the, book}}} (Oishi 2015, pp. 324-325)

rendering the operation problematic as a function in the strict mathematical sense. This is the core of the objection raised by Marcolli et al. (2025), who argue that an operation whose output contains information absent from its input cannot be rigorously defined as a function.

The problem becomes even more apparent in disjunctive configurations. When the result of Pair-Merge is defined as <K, L>, the categorial head K no longer behaves as an independent syntactic object but instead appears to attach to L in a way analogous to the addition of an index. Formally, this is comparable to mapping an object L to a modified object <L, i>, where the element i encodes the ordering or categorial contribution of K. Under such an interpretation, Merge no longer constructs a new composite structure from two objects but rather alters the internal representation of one of them. As a result, the operation resembles a transformation of an existing object rather than a genuine combinatorial process. It is this "sticking" or index-like effect that further undermines the mathematical coherence of sequence-based Pair-Merge, reinforcing the conclusion that order-sensitive formulations of Merge are ill-suited for a formally stable theory of syntactic computation.[8] The author assumes that disjunction and conjunction within SDs in Chomsky (2019b) are treated as instances of Form Sequence, whereas in Chomsky (2021) they are reanalyzed in terms of Form Set, precisely for these reasons.

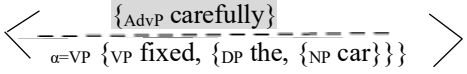
3.2 Pan and Du's (2024) Analysis of Separate Dimensions

More recent work has developed the notion of SDs in greater detail. Among these proposals, Pan and Du (2024) assume that the capacity for multiple dimensions is an inherent property of the human language and that such dimensions can in principle be generated without bound. In their analysis, given two syntactic objects undergoing Merge, one object is taken to reside in the main WS as the stem structure, while the other is located in an SD. They further claim that Set-Merge and Pair-Merge are, at base, the same kind of operation. Concretely, they allow structures such as $<_{NP}\{_{AP}$ young$\}$, $\{_{NP}$ man$\}>$ and $<_{AP}\{_{AP}$ young$\}$, $\{_{NP}$ man$\}>$ to be generated in syntax, with interpretive filtering taking place at the CI interface. In $<_{NP}\{_{AP}$ young$\}$, $\{_{NP}$ man$\}>$, $\{_{NP}$ man$\}$ is in the main workspace and $\{_{AP}$ young$\}$ in an SD; for a reading corresponding to $<_{AP}\{_{AP}$ young$\}$, $\{_{NP}$ man$\}>$, $\{_{AP}$ young$\}$ is in the main workspace and $\{_{NP}$ man$\}$ in an SD. Both structures are freely generated by the syntax, but the CI interface filters out the unwanted adjective phrase reading. Elements in an SD are invisible to the labeling algorithm and therefore irrelevant for labeling. On this view, syntax "blindly" generates structures much like a factory line, and the interfaces simply make use of what they need.

This raises the question of whether such a mechanism is compatible with CE. Pan and Du (2024) appeal to CI filter conditions, in particular the legibility condition: syntactic objects must be composed of features interpretable at the interfaces, and any material that is not legible at the interface is filtered out. To illustrate, consider the following example:

(9) The mechanic who fixed the car carefully packed his tools.

a. $<_{\alpha=VP}\{_{AdvP}$ carefully$\}$, $\{_{VP}$ fixed, $\{_{DP}$ the, $\{_{NP}$ car$\}\}\}>$

b. 

[8] Chomsky (1995) claims that Merge applies only to Roots and nothing can be inserted into already formed syntactic objects. This is analyzed in terms of the NTC and ICC principles: NTC prohibits altering existing structures during syntactic operations, and ICC forbids introducing any new elements not present in the Lexicon.

c. $<_{\alpha=VP}$ {$_{AdvP}$ carefully}, {$_{VP}$ packed, {$_{DP}$ his, {$_{NP}$ tool}}} >

d. $\left\langle \begin{array}{c} \{_{AdvP}\ \text{carefully}\} \\ \alpha=VP\ \{_{VP}\ \text{packed}, \{_{DP}\ \text{his}, \{_{NP}\ \text{tool}\}\}\} \end{array} \right\rangle$ (Pan and Du 2024, p. 93)

This sentence allows two readings, depending on whether *carefully* modifies *fixed the car* or *packed his tools.* Depending on where the adverbial phrase (AdvP) is adjoined in the structure above, the sentence receives different interpretations, which shows that adjunction is an integral part of the syntactic derivation. Pan and Du (2024) further distinguish cases where the targets of Pair-Merge belong to the same category from those where they belong to different categories: Pair-Merge between syntactic objects of the same category is treated as coordination, while Pair-Merge between objects of different categories is treated as modification. Their analysis includes the discussion of Pair-Merge in SDs by addressing issues of labeling and of Markovian vs. non-Markovian interpretations. Nevertheless, although Pan and Du (2024) provide a rich and visually oriented perspective on SDs, the principles governing how Pair-Merge accesses SDs remain somewhat unclear, particularly with respect to the nature of the link. Even if the capacity to generate SDs is taken to be an inherent property of language, the principles that regulate the formation of Set-Merge and Pair-Merge must be made more explicit. An approach that secures such ordered-pair effects while treating adjunction as a well-formed instance of Set-Merge under the general Merge system is provided by Fong and Oishi's (2025) application of Form Set (FS).

3.3 Fong and Oishi's (2025) Analysis of parallelism

One of the most important recent studies on XP-adjunction is Fong and Oishi's (2025) analysis of the principle of Form Set (FS). Chomsky (2021, 2024) adopts the view that although set formation is the most basic principle of the Merge system, natural language also exhibits Unbounded Unstructured Sequences (UUS), and that FS is required to model such sequences. On this basis, Fong and Oishi (2025, p. 3) define FS as in (10):

(10) FS, i.e. n-ary set formation for n ≥ 2, indicated using underscore boldface braces "{…}," is another fundamental operation that constructs sets whose members must obey a special requirement (a requirement that sets built by Merge do not meet).

Sets formed in this way are called FS-sets, and the members of an FS-set are supposed to satisfy an appropriate "parallelism requirement." The core of their analysis is to articulate the "Parallelism Properties" associated with FS. They use this notion in two ways. First, parallelism refers to conditions such as 'referential similarity' and predicate-hood. In particular, for typical FS targets like NPs and VPs, referential similarity or predicate-hood is tied to features on the lexical head, and syntactic objects that share such similar properties qualify as targets for FS. Second, any operation targeting an FS-set must apply in a strictly parallel fashion to each of its members. FS thus has the character of free Merge, but whatever operation is applied to an FS-set is imposed uniformly on all of its members. When n[9] ≥ 2, FS yields a structure of the form {X, Y}, which Fong and Oishi explicitly distinguish from an ordinary set {X, Y}; although both are sets, the FS-set {X, Y} encodes the fact that X and Y share a common property and can be seen as an ordered pair.

[9] '*n*' refers not to the number of Separate Dimensions, but to the number of syntactic objects targeted by FS. If a Separate Dimension is construed as the combined syntactic object itself, then the two notions can be regarded as equivalent.

(11) a. the student who lives here who studies English who(m) I know
b. {who$_{Rel}$[10], {student, {v*, {lives here}}}}
{who$_{Rel}$, {student, {v*, {studies English}}}}
{who$_{Rel}$, {I, {v*, {know student}}}}
c. {{who$_{Rel}$, {student, {v*, {lives here}}}}, {who$_{Rel}$, {student, {v*, {studies English}}}},
{who$_{Rel}$, {I, {v*, {know student}}}}}
d. {student, {{who$_{Rel}$, {student, {v*, {lives here}}}}, {who$_{Rel}$, {student, {v*,
{studies English}}}}, {who$_{Rel}$, {I, {v*, {know student}}}}}} (Fong and Oishi 2025, p. 3)

In the examples above, three relative clauses are formed in parallel, and once the syntactic objects that share a referential property have been identified, Internal Merge of the structurally identical occurrence of 'student' yields the configuration shown. The referential property in their analysis can also be encoded in features on the head.

(12) a. Arabic: Nφ.DEF Aφ.DEF
al-rajul-u al-saʕid-u
DEF-man-$_{NOM.SG.M}$ DEF-happy-$_{NOM.SG.M}$
'the happy man'

b. Hebrew: Nφ.DEF DEFφ
ha-yeled ha-ze
DEF-child.φ.$_{SG.M}$ this.φ.SG.M
'this child' (Fong and Oishi 2025, p. 9)

As illustrated in (12a), in Arabic two syntactic objects bearing the affixal DEF feature combine to form a structure corresponding to 'the happy man'. In (12b), in Hebrew, the word ha, which carries a DEF property, is shared and contributes to the structure 'this child.' Fong and Oishi (2025) take such nominal head features to instantiate referential similarity. Although agreement in Case or in Ø-features can allow an adjective and the nominal head to share referential similarity, they argue that feature sharing or Ø-feature matching is not the operation for Merge or for FS. Rather, these features belong to language-particular externalization rules. This is precisely where further research on 'referential similarity' is needed

The important point in Fong and Oishi's (2025) analysis is that purely operational conveniences such as category labels or extra features must be excluded by ICC and NTC. Concretely, they argue that structures of the form {D, {N$_1$, …, N$_n$}} are not generated by the syntax. In this regard, consider the following data. If the structure {D, {N$_1$, …, N$_n$}} were generable, then the example in (13a) should be well-formed.

(13) a. *The and (a/this) book are on sale
b. {book$_{[the]}$, book$_{[a/this]}$}
c. {{~~book~~$_{[the]}$, ~~book~~$_{[a/this]}$}, book}
d. the book (you mentioned) and this/a book are on sale. (Fong and Oishi 2025, p. 13)

If D were the head that projects a DP, then a phrase like "the and a book" ought to be possible, contrary to fact.

[10] In who$_{Rel,}$ '*Rel*' indicates 'Relative pronouns'

Since the string in (13a) is ungrammatical, whereas the coordination in (13d) is perfectly acceptable, Fong and Oishi (2025) conclude that these nominals should not be analyzed as DPs headed by D, but rather as NPs, with the article D treated as a property of NP (see Footnote 7).

Notably, Fong and Oishi (2025) do not employ the term "Separate Dimension." That term presupposes a main workspace centered on the stem structure. By contrast, their analysis dispenses with any explicit distinction between a main workspace and SDs. Instead, they assume that the syntax freely generates UUS under the notion of parallelism; once the Merge system identifies referential similarity, the relevant members are grouped into an FS-set, thereby yielding ordered-pair-like behavior. If each freely generated UUS were reinterpreted as SDs, their view would converge with Chomsky's (2019b) proposal, except that FS requires at least two members. In other words, there is no counterpart to Chomsky's disjunction (n = 1) in their system. Furthermore, although Fong and Oishi's work provides a detailed account of FS and ensures ordered-pair effects, it leaves open the nature of the link that connects these FS-sets to the stem structure. This raises two issues: when exactly are freely generated UUS created, and are they compatible with the seven desiderata when they are linked back to the stem structure?

4. Proposal: Head Can Have SDs

The original motivation for distinguishing Set-Merge from Pair-Merge lies in the existence of asymmetric syntactic objects in natural language whose components bear an intrinsic ordering. From the standpoint of computational and mathematical principles, however, the direct implementation of Pair-Merge in the Merge system is impossible. It is therefore crucial to identify a principle that is compatible with the Merge system and nevertheless yields structures that are interpreted as ordered pairs. The approaches surveyed above either handle such structures in a dimension distinct from the stem structure or generate UUS configurations that are subsequently grouped into FS-sets by parallelism. Building on these insights, the following sections analyze the step-by-step progression of the syntactic derivation related to SDs.

4.1 Head-Adjunction

Adjunction can be divided into Head-Adjunction and XP-adjunction. If adjunction takes place in an SD, we must first clarify the notion of Head. Chomsky (1970) argues that category-less Roots are inserted into syntactic structures and receive a category only within those structures. Chomsky's (2013, 2015) analysis of substantive elements in the Lexicon proceeds along the same lines. (The following discussion presents Oishi's (2015, p. 327) analysis of Chomsky's proposals.)

(14) i)" that substantive elements of the lexicon are Roots, unspecified as to category"
ii)" that category ... derives from merger with a functional element n, v, etc."

The Lexicon, likewise, is assumed to store Roots, while categories are treated as functional elements introduced by the Merge system. Borer (2014, p. 117) develops Chomsky's analysis and proposes the notion of "Contextual Categorization." She writes:

(15) "It is the structural context that is responsible for categorial properties, and not the inherent, presumably listed properties of a terminal."

In other words, categorial properties are determined within syntactic structure, not by inherent lexical listings.

On Borer's (2014) view, a category-neutral Root becomes a verb when it merges in a verbal context and a noun when it merges in a nominal context, with the categorial context driving the categorization. Merchant (2019) similarly argues that the nominalizer N both selects which Root it combines with and determines which preposition is compatible with that Root, treating the "categorizing head" as the selector of the Root. This paper takes the "Category" in Chomsky (2013, 2015) and Borer (2014) to correspond to a "categorial head". This is supported by Oishi's (2015, p. 323) discussion of substantive elements:

(16) "Specifically, a substantive element is regarded as a syntactic complex (or amalgam) of a categorizer (or categorial specification) K and a Root R: <K, R>. The Root ('undifferentiated Root' in Chomsky's (2007) terminology) is assumed to be common to any relevant categorial alternants, and responsible for the selection properties. ~ in assuming that a categorizer and a Root to be associated are introduced separately into syntax and end up with an amalgam through derivation, to put it more specifically, that a categorial character of Root is derivationally determined. ~ "

In other words, a substantive element is not a simple object in syntax, but has the composite form <K, R>, where a categorizer K combines with a Root R; in the syntax proper, this complex behaves indistinguishably from a syntactic object. The combination <K, R> is an amalgam, and its formation is assumed to be automatic. According to Oishi (2015), a syntactic object X is not composed of a single, undifferentiated X. Rather, the X accessed by the Merge system in the workspace has an internal structure of the form <K, R>, where K is a categorial head and R is a Root, such as <N, book>. The formation of this internal structure takes place at the moment the Root is introduced from the lexicon into the workspace. Oku (1998) refers to this process as "selection," analyzing it as the process that determines the categorial features of a lexical item and introduces that categorized item into the workspace.

To summarize, a substantive element (i.e. a lexical item) exists in the Lexicon as a Root, and when it is placed in the workspace it already bears the form <K, R> with a functional head attached. In this form, it is visible as an SO for the purposes of the Merge system. In the <K, R> configuration that underlies categorization, the author assumes that K is not a "Category" in the sense of a higher label, but a Head, and there is no categorial label above Head. As in Fong and Oishi's (2025) discussion of DP, the Merge system does not merge directly with separate categorial labels. A complex like <K, R>, which carries functional-head information, is therefore visible to Merge in exactly the same way as a SO.

If this analysis is correct, a further question arises. The configuration <K, R> is an amalgam of a head and a Root, <H, R>. While such an amalgam may be legitimate in the Lexicon, as in Shim's (2022) example 'un-do-able', its formation in syntax requires reconsideration. The reason is that the Lexicon is the domain of meaning formation, whereas syntax is the domain of structure building; in syntax, unlike in the Lexicon, CE is the primary guiding principle. This is precisely why seven desiderata are imposed on the operation of capital MERGE. The issue, then, is whether the initial combination <H, R> that appears in the workspace is compatible with these computational principles. What is needed is a way to maintain the visibility of <H, R> while remaining fully compatible with the Merge system.

The author seeks to apply the mathematical principles from Tourlakis (2003, p. 182-183) and Kuratowski (1921), as discussed in Omune (2019) and Shim (2020) with respect to Head-Adjunction, to the process by which syntactic objects are introduced from the Lexicon into the workspace. These principles are further applied to a detailed analysis of the link that connects Pair-Merge-generated SDs to the stem structure. Tourlakis (2003) and Kuratowski

(1921) establish that an ordered pair <x, y> is equivalent to the set {{x}[11], {x, y}}. Omune (2019) leverages the invisibility of certain elements to the Merge system together with Tourlakis's (2003) principle to license the structure {{R}, {R, v*}} as the output of Pair-Merge between a Root R and v*, in conformity with the principles of capital MERGE. Similarly, Shim (2020) uses Kuratowski (1921) to derive {{v*}, {v*, R}} as the structure underlying <v*, R> in Head-Adjunction.

The paper generalizes the structure {X, {X, Y}} to {H, {H, √Root}}, where H is visible to the Merge system that builds the stem structure, while {H, √Root} resides in an SD linked to H. In other words, H functions as both the categorizer and the link. Within the SD, {H, {H, √Root}} is a standard Set-Merge configuration, and the same principles of the Merge system apply there as in the main workspace. At the interfaces, this structure is interpreted as <H, √Root>, without violating ICC or NTC. Some scholars define the attachment of a head to √Root as categorization, in which case Head and Category are effectively identical, with Category serving as categorizer and link. The author, however, avoids the terms "Categorizer" or "Category" in favor of "Head" because traditional notions of Category do not guarantee combination at the terminal node, as in Head. As noted by Fong and Oishi (2025), the Merge system does not target category labels directly, and simultaneously the form √Root itself is ill-suited to Merge. The required format must therefore be compatible with Merge without being a category label or a bare √Root. Oku's (1998) analysis of "feature decomposition"[12] and "selection" precisely captures the process by which √Root becomes a syntactic object through combination with a functional head. The structure {H, {H, √Root}} is thus legible to the Merge system as <H, R>. If the process by which a √Root-form lexical item from the Lexicon is placed in the first workspace or pre-Merge stage (see Footnote 13) is termed 'selection', then Head-Adjunction takes place at this stage, yielding {H, {H, √Root}} = <H, √Root> in the first stage of workspace.

This Head-Adjunction process corresponds to Chomsky's (2019b) notion of disjunction. In other words, the traditional concept of amalgamation refers exactly to this operation. On the judgement of this paper, the Pair-Merge involved in Chomsky's (2019b) disjunction and conjunction is analyzed as a unidirectional "attachment" mode. The problem with this mode is that it is mathematically ill-defined and incompatible with the minimalist Merge system, as it fails to conform to computational principles. By contrast, Form Set (FS) avoids these issues: Merge proceeds in its basic set-forming mode, and if the linking problem can be resolved, no resource constraints are violated and computation can continue unimpeded. The author claims that forming the structure {H, {H, √Root}} is precisely the mechanism that is legitimate within the Merge system, guarantees ordered-pair interpretations, and ensures CE. The author's claim can be summarized as follows.

(17) **Premises**

(i) The initial Head-Root combination is realized as a set-theoretic structure {H, {H, √Root}}, derived by standard Set-Merge in conformity with computational principles.

(ii) In this configuration, H is visible to subsequent applications of Merge, while the structure is interpreted at the interfaces as <H, √Root>.

(iii) No independent categorial label distinct from H is introduced in the syntax.

---

[11] In mathematics, {x} denotes the singleton set containing x. Therefore, I assume that x inside a set and x itself are identical in language.

[12] Oku (1998) argues that, at LF, movement is possible by decomposing and recombining only certain features, a view that underlies the notion of feature decomposition. According to this account, lexical items enter the workspace through the combination with a head via feature decomposition. Oku's (1998) analysis can be summarized as follows: the selection properties of a head are syntactic features that must be checked in the course of the derivation, and some of these selection features are weak enough to be checked in the covert component of syntax.

The analysis aims to show that the structure of an SD is guaranteed as a set and can be linked to the stem structure computation without violating Resource Restriction or determinacy. Chomsky's (2019b) distinction between disjunction and conjunction is based on the cardinality of n-ary structures, corresponding to the number of SDs that Pair-Merge can generate. In Fong and Oishi's (2015) analysis, n-ary refers to the number of syntactic objects targeted by FS, which therefore requires $n \geq 2$. If SDs are not construed as distinct spaces (and they should not be), but rather as structures of independently merged syntactic objects (SOs), then the notions coincide; however, Fong and Oishi's (2025) FS analysis excludes the case $n = 1$. This makes it difficult to apply their approach to modification relations, such as cases in which a verb phrase is modified by an adverb phrase, where there is no referential similarity, as in example (9). Accordingly, this paper analyzes the process of SO formation at the Lexicon-to-workspace (i.e., Head-Adjunction) and then divides cases into disjunction (one SD) and conjunction (two or more SDs), following Chomsky (2019b).

4.2 Disjunction

As noted earlier, Chomsky (2019b) defines disjunction as the case where Pair-Merge applies in a distinct dimension, forming exactly one SD. The author takes the Head-Adjunction process analyzed above to also instantiate disjunction. Head-Adjunction is classified separately only because it is essentially identical to the process of placing items in the workspace and thus does not require the notion of SD. From the author's perspective, the case where there is a single SD ($n = 1$) corresponds to configurations such as a verb phrase receiving modification by an adverb phrase, or a noun phrase receiving modification by an adjective.

4.2.1 Modification by an adjective: 'the young man'

As discussed earlier, following Oishi (2015), the present analysis does not posit a distinct DP but treats the configuration as an NP. At the pre-merge[13] stage, where lexical items become syntactic objects, these are introduced into the workspace as {A, {A, √young}} and {N*, {N*, √man}}[14]. Suppose that FS applies at the same time as the Merge system searches for a target to merge with V. These two syntactic objects lack the referential similarity required by Fong and Oishi (2025). For them to exhibit referential similarity, {A, {A, √young}} must first combine with {N, {N, √man}} to form 'young man', enabling FS between 'the man' and 'young man' on the basis of the shared 'man.' This is how FS applies in Fong and Oishi (2025), but the present author argues that Pair-Merge in the case of $n = 1$ disjunction should instead be analyzed as a modification relation in the sense of Pan and Du (2024). This modification relation is connected to the hierarchical phrase structure rules in Chomsky (1957).

Phrase structure rules are too complex to form the basis of Universal Grammar from an evolutionary perspective, but they provide crucial principles for understanding sentence structure. The specific realization of modification relations varies across languages, but their governing principles are believed to stem from the second of Chomsky's

[13] The pre-merge stage is a term used by Epstein et al. (2016). They assume that, due to the need for phase cancellation in 'passive' or 'unaccusative' constructions, R and v undergo pre-merge prior to the onset of the derivation. In their analysis, pre-merge refers to 'external Pair-Merge' between lexical items in the workspace. By contrast, the pre-merge concept employed here denotes the combination of √Root with a Head. In other words, the analyses differ in how R is construed. If R = V, then Epstein et al.'s (2016) pre-merge corresponds to Head-Head combination of V and v. If $R \neq V$, then R = √Root, which exists in the Lexicon and is not effectively visible to the workspace. Consequently, this paper takes R = V, such that the pre-merge stage refers to the process whereby √Root combines with a Head to form a syntactic object that enters the workspace.

[14] N* uses the superscript '*' to distinguish definiteness from indefiniteness in nouns. Naturally, N denotes indefiniteness.

(2005, 2007) three factors of language, namely Experience—i.e., the linguistic input and parameter settings generate a hierarchy of modification relations. Pan and Du (2024) also analyze noun phrases modified by adjectives as generating both AP and NP structures, with the CI interface filtering them. After processing linguistic input and parameter settings, the present author assumes the structure 'the young man' is formed as follows:

(18) {A, {A, √young}}
{N*, {N*, √man}}

{N*, {N*, √man}} = $_{NP}$man$_{[the]}$ and {A, {A, √young}} = $_{AP}$young are simultaneously placed in the workspace. If 'the young man' is the object in the sentence, $\sum_{Merge}$ searches for V and 'the young man,' forming an SD as in (18), where V merges with 'the young man.' The formation of this stage can be examined within (19)

(19) Mary loves the young man
Lexicon: √man, √young, √love, √Mary ….
Pre-Merge: {N*, {N*, √man}}, {V, {V, √love}}, {A, {A, √young}}…..
WS: $_{V}$love, $_{N*}$man, $_{A}$young …
Merge ∑: ($_{V}$love, ?) / SD simultaneously form the IA 'the young man' in (17)[15]
WS$_1$: {$_{V}$love, $_{N*}$man}
{A, {A, √young}}

4.2.2 Modification by an adverb: 'fix the car carefully'

Consider the SD created by an adverbial phrase in this example:

(20) fix the car carefully
Lexicon: √fix, √car, √carefully ...
pre-merge: {V, {V, √fix}}, {N*, {N*, √car}}, {Ad, {Ad, √carefully}}
WS: $_{V}$fix, $_{N*}$car, $_{Ad}$carefully, v*, ...[16]
WS$_1$: {$_{V}$fix, $_{N*}$car}
WS$_2$: {$_{N*}$car, {$_{V}$fix, $_{N*}$car}}[17]
WS$_3$: {v*, {$_{N*}$car, {$_{V}$fix, $_{N*}$car}}}
WS$_4$: {$_{v*}$-$_{V}$fix, {$_{N*}$car, {$_{V}$fix, $_{N*}$car}}}[18]

[15] The author argues that the combination of $_{N*}$man and $_{A}$young occurs simultaneously with the Search operation of Merge, rather than prior to it. This claim is supported by two considerations: first, both possibilities yield the same output in the workspace; second, it is unwarranted to posit the existence of a pre-merged structure such as the young man in the initial workspace or at any pre-merge stage.

[16] All uses of indices such as $_{V}$*fix*, $_{N*}$*car*… are employed for convenience of expression.

[17] In the structure {$_{N*}$car, {$_{V}$fix, $_{N*}$car}}, N is not indicated as the label of the entire syntactic object. Rather, N and V indicate how the respective syntactic objects are shown to the Merge system. Whether Object Shift (OS) applies obligatorily remains a matter of ongoing debate and falls outside the scope of the present paper. Here, the author simply includes cases in which OS is potentially available.

[18] The author assumes the combination of v*-V as a Head-Head Merge, as mentioned in Footnote 13. It requires further in-depth study to determine whether it should be viewed as v*-V (v*-R) or V-v*(R-v*), as well as to analyze data showing cross-

WS_5: {Ad, {Ad, √carefully}
{v*-V fix, {N* car, {V fix, N* car}}}

After the Lexicon and pre-merge stages, syntactic objects are placed in the workspace, and then $\sum_{Merge}$ begins. The Merge proceeds from WS_1 to WS_5. The SD {AdvP, {AdvP, √carefully}} connects to v*. The dotted linkage represents {v*, <AdvP, √carefully>}. Here, v* functions as a link, and the Merge system sees {v*, <AdvP, √carefully>}, which at the interfaces is interpreted as the WS_5 structure {v*, <AdvP, √carefully>}. This configuration satisfies Resource Restriction and Determinacy conditions required by the Merge system. The author analyzes the link formed by <AdvP, √carefully> in the SD as attaching not to the VP (= RP) but rather to v* as a link. In syntax, connecting a syntactic object in an SD to a V that requires Merge with v* is akin to carrying a heavy load. The operation of $\sum_{Merge}$, however, proceeds naturally, automatically, and freely, optimizing CE.

4.3 Conjunction

Conjunction involves cases where at least two Separate Dimensions are generated, including configurations in which FS operates to form parallel ordered pairs, as well as cases in which two or more SDs arise from modification relations. Conjunction is examined through three types: (i) multiple adjectives forming SDs on the same link, combined via FS as ordered sequences; (ii) relative clauses generating SDs through modification relations; and (iii) the integration of those relative clauses with the main clause.

4.3.1 Parallelism in FS

Consider the formation of multiple adjective phrases connected to a noun phrase, example (17), analyzed following Fong and Oishi (2025) and restated here in stages:

(21) a long narrow and dark hallway
Lexicon: √long, √narrow, √dark, √hallway, …
Selection: √long, √narrow, √dark, √hallway, N, A, …
Pre-merge: {A, {A, √long}}, {A, {A, √narrow}}, {A, {A, √dark}}, {N, {N, √hallway}}, …
WS: $_A$long, $_A$narrow, $_A$dark, $_N$hallway
WS_1: {$_A$long, $_N$hallway}
WS_2: {$_A$narrow, $_N$hallway}
WS_3: {$_A$dark, $_N$hallway}
WS_4: {{$_A$long, $_N$hallway}, {$_A$narrow, $_N$hallway}, {$_A$dark, $_N$hallway}}
WS_5: {hallway, {{$_A$long, $_N$~~hallway~~}, {$_A$narrow, $_N$~~hallway~~}, {$_A$dark, $_N$~~hallway~~}}}

In example (18), the process proceeds within SDs. The lexical items and selection put SOs in the workspace, and the noun phrases proceed through the same steps as with the n=1 example, forming {$_A$long, $_N$hallway} in parallel. The author considers WS_1-WS_3[19] as a simultaneous generation rather than sequential. A potential question

linguistic variation such as v-T-C and T-C. This paper excludes such in-depth analysis. (See Blümel, 2024; Blümel et al., 2024)

[19] In other examples, the subscript n in WS_n indicates an update to the workspace, but WS_1-WS_3 in example (18) do not signify sequential application; they are used merely for convenience.

here concerns the issue of *and*-insertion and how this compares with a method in which FS first applies to multiple adjectives, which are then combined with the linking noun.The author adheres to Fong and Oishi's (2025)[20] view on the insertion of 'and.' Furthermore, unlike (18), UUS (Unbounded Unstructured Sequence) structures might be generated in other ways, such as first forming {$_A$long, $_A$narrow, $_A$dark} and then combining with $_N$hallway as {$_N$hallway, {$_A$long, $_A$narrow, $_A$dark}}. Fong and Oishi (2025) assume multiple methods exist for generating UUS. Chomsky (2019b) invokes Hilbert's (1923) epsilon operator[21] in this kind of context. After all possible data configurations are generated, the epsilon operator selects one and eliminates the rest. The present author analyzes this process as ensuring that FS operates in the direction of maximal CE. The reason for this analysis, paralleling Fong and Oishi (2025), is that adjectives, being predicative, require a θ-marked target for the derivation of SDs to be legitimate. The author assumes that the operation of FS and its integration into the stem structure at WS_5 takes place at the stage of NP-Merge.

4.3.2 Modification relation

The following illustrates the formation of a relative clause and the overall sentence as an example of conjunction ($n \geq 2$). Example (19) represents the process within SDs, and the author assumes that this process links to the stem structure after the external argument (EA) undergoes external Merge (EM) followed by internal Merge (IM) to Spec-Infl. As noted earlier, linking of SDs occurs automatically after the necessary copies are formed. If linking takes place after EA's EM, it would require carrying a heavy SD during IM of EA, which is incompatible with CE. Accordingly, one further premise is added.

(22) **Additional Premise.**
The linking relation between the main workspace structure and subordinate structural domains (SDs) is established only after the copy formation of the syntactic object in the main workspace has been completed.

(23) who fixed the car carefully
lexicon: √who, √fix, √car, √carefully, ...
Selection: √who, √fix, √car, √carefully, V, N, Ad, v*...
Pre-merge: {N, {N, √who}}(= $_N$who), {V, {V, √fix}}, {N, {N, √car}}, {Ad, {Ad, √carefully}}
WS: <N, √who>, <V, √fix>, <N*, √car>, <Ad, √carefully>
: $_N$who, $_V$fix, $_{N*}$car, $_{Ad}$carefully, …
WS_1: {$_V$fix, $_{N*}$car} EM of IA, θ assign
WS_2: { $_{N*}$car {$_V$fix, $_{N*}$car }} Object Shift/ IM
WS_3: {v*, { $_{N*}$car {$_V$fix, $_{N*}$car }}} EM v*
WS_4: {<v*-V, √fix>, { $_{N*}$car {$_V$fix, $_{N*}$car }}} Head Adjunction
WS_5: {$_{v*\text{-}V}$fix, { $_{N*}$car {$_V$fix, $_{N*}$car }}} EPM of 'carefully'
$_{Ad}$carefully
WS_6: {$_N$who, {$_{v*\text{-}V}$fix, { $_{N*}$car {$_V$fix, $_{N*}$car }}}} EM of EA, θ assign
$_{Ad}$ carefully

[20] Fong and Oishi (2025, p. 6) assume *and* is not part of Merge syntax, but instead a language-particular spell-out option.
[21] In summary, the element selected by the epsilon operator applies identically to all other elements bundled in the set. This corresponds to 'referential similarity' or 'structural identical inscription' in Fong and Oishi (2025).

(24) The mechanic who fixed the car carefully packed his tools

lexicon: √mechanic, √pack, √tool, √who, √fix, √car, √carefully, ...

Selection: √mechanic, √pack, √tool, √who, √fix, √car, √carefully, V, N, Ad, v*, C, Infl...

Pre-merge: {N*, {N*, √mechanic}}, {V, {V, √pack}}, {N*, {N*, √tools}}, {N, {N, √who}},
{V, {V, √fix}}, {N, {N, √car}}, {Ad, {Ad, √carefully}}, v*, C, Infl,…

WS: $_{N*}$mechanic, $_{V}$pack, $_{N*}$tools, $_{N}$who, $_{V}$fix, $_{N}$car, $_{Ad}$carefully, v*, C, Infl, …

WS_1: {$_{V}$pack, $_{N*}$tools}

WS_2: {$_{N*}$tools, {$_{V}$pack, $_{N*}$tools}}

WS_3: {v*, {$_{N*}$tools, {$_{V}$pack, $_{N*}$tools}}}

WS_4: {<v*-V, √pack>, {$_{N*}$tools, {$_{V}$pack, $_{N*}$tools}}}

WS_5: {$_{N*}$mechanic, {$_{v*\text{-}V}$pack, {$_{N*}$tools, {$_{V}$pack, $_{N*}$tools}}}}

WS_6: {Infl, {$_{N*}$mechanic, {$_{v*\text{-}V}$pack, {$_{N*}$tools, {$_{V}$pack, $_{N*}$tools}}}}}

WS_7: {$_{N*}$mechanic, {Infl, {$_{N*}$mechanic, {$_{v*\text{-}V}$pack, {$_{N*}$tools, {$_{V}$pack, $_{N*}$tools}}}}}}

$_{Ad}$ carefully

{$_{N}$who, {$_{v*\text{-}V}$fix, {$_{N*}$car {$_{V}$fix, $_{N*}$car }}}}

WS_8: {$_{N*}$mechanic, {Infl, {$_{N*}$mechanic, {$_{v*\text{-}V}$pack, {$_{N*}$tools, {$_{V}$pack, $_{N*}$tools}}}}}}

$_{Ad}$ carefully

{$_{N}$who, {$_{v*\text{-}V}$fix, {$_{N*}$car {$_{V}$fix, $_{N*}$car }}}}

WS_9: {C, {$_{N*}$mechanic, {Infl, {$_{N*}$mechanic, {$_{v*\text{-}V}$pack, {$_{N*}$tools, {$_{V}$pack, $_{N*}$tools}}}}}}}

As described above, lexical items acquire functional heads during the selection process from the Lexicon to the workspace; this stage constitutes pre-merge. Once elements are placed in the workspace, they qualify as syntactic objects. The present author takes the verb Root to become readable by the Merge system in the form V, which is identical to the Root but legible in syntax proper. External Merge proceeds from W_1, with "raising-to-object" occurring at WS_2. In WS3, the phase head v*[22] is introduced by external Merge. Sugimoto (2022) analyzes v*'s unvalued features as undergoing feature inheritance to the Root (R). Once the NP's unvalued Ø-feature is valued, the RP or VP receives the label <Ø, Ø>. In WS_4, the combination <v*-V, √Root> causes v* to lose its phasehood. Sugimoto (2022) argues that in some languages, where v* only plays a role as a suffix, v* adjoins to R and becomes invisible. The question then arises as to how v*P is labelled According to Epstein, Kitahara, and Seely (2016), labeling need not be determined exclusively by one of the heads involved in a Merge operation. Rather, labels may be established on the basis of features that are interpretable at the CI interface. From this perspective, labeling remains possible even when one of the merged elements becomes invisible due to adjunction or deletion, provided that the structure as a whole shares interpretable features. Consequently, even if v* adjoins to the Root and ceases to function as an independent head, the categorial and semantic features already associated with the VP can serve as the basis for labeling. At WS_5, the EA is externally merged and receives θ-marking. Infl is externally merged at WS_6. At WS_7, the EA undergoes Internal Merge to Spec-Infl, and once copy formation of the EA is completed, the

[22] One of the reviewers raised the question of what status functional heads such as v*, C, and Infl have at the pre-merge stage. The author assumes that, in the case of such functional heads, they are treated as having the same form across the Lexicon, the pre-Merge stage, and the workspace. However, since there are languages in which these heads are realized as overt morphemes, a detailed investigation of this issue is left for future research.

SD (defined in (23)) is linked in workspace WS_8. At WS_9, C is introduced by external Merge and feature inheritance takes place. Valuation of the NP's unvalued features enables <Ø, Ø> labeling at WS_8, and WS_9 is finally labelled as CP.

Bode (2020) analyzes the properties of SDs, noting that adjunction neither alters the elements selected by Merge nor involves self-selection by the adjunct. Adjunction creates asymmetric structures subject to island constraints (no extraction from one of the elements), and it does not project or label. It is invisible and inaccessible to MERGE and thus does not participate in labeling during the above derivation.

5. Conclusion

This study aims to elucidate how asymmetric structures in natural language—namely those arising from Pair-Merge—can be derived without compromising CE. To summarize the dimensions in which Merge applies: external and internal Set-Merge proceed in the primary workspace, with external Merge forming arguments and internal Merge encoding discourse-related content; external and internal Pair-Merge, by contrast, are handled by generating SDs. Analyses of SD creation and integration in Pair-Merge fall into four main categories: (i) the SIMPL mechanism in Chomsky (2008) and Bode (2020); (ii) Chomsky's (2019b) link-based analysis centered on the categorizer; (iii) Pan and Du's (2024) modification-relation approach; and (iv) Fong and Oishi's (2025) analysis of FS properties via referential similarity or predicatehood.

Chomsky (2008) and Bode (2020) address the integration of Pair-Merge-generated SDs into the stem structure via the SIMPL mechanism. Chomsky (2008) treats adjoined syntactic objects as just converted via SIMPL for syntactic computation, while Bode (2020), building on this, invokes transfer: after transfer from the SD, Pair-Merge structures undergo SIMPL(ify) to become Set-Merge configurations interpretable at the interfaces without issue. The present author excludes SIMPL from detailed analysis, deeming it incompatible and impossible with $\sum_{\text{Merge}}$.

Chomsky's (2019b) analysis posits that Pair-Merge generates SDs, with links to the stem structure provided by the categorizer; single SDs are disjunctions, while n-ary ($n \geq 2$) cases are conjunctions.

Pan and Du (2024) insist on distinguishing whether syntactic operations instantiate an abstraction of language or language itself; they treat Merge as a powerful abstract mechanism of FL. Connections between stem structure and SDs are modification relations if categories differ, or coordination if they match; SD generation is intrinsic to language and requires no dedicated link since Pan and Du assume that creating SDs is naturally and freely proceeded

Among recent analyses of adjunction, Fong and Oishi's (2025) account of FS operation stands out. They argue that ordered-pair structures like those from Pair-Merge are formed by FS, whose basic property is parallelism. Parallelism has two aspects: (i) target syntactic objects should have 'referential similarity', and (ii) any operation applies uniformly to each. FS thus applies only for $n \geq 2$ in their analysis. On the author's understanding, Parallelism connects to the mathematical notion of Hilbert's epsilon operator, as noted by Chomsky (2019b). Fong and Oishi's diverse examples demonstrate the most legitimate computational analysis of Pair-Merge without additional mechanisms, though they require extensions for $n = 1$ cases and adjunctions lacking referential similarity.

Building on these analyses and using Chomsky's (2019b) disjunction and conjunction, the present study treats disjunction as encompassing Head-Adjunction (Lexicon-to-Workspace SO placement), NP formation via adjectival modification, and v*p formation via adverbial modification in a single SD. For conjunction ($n \geq 2$), it analyzes multiple adjectives modifying a noun (with referential similarity) and the generation and adjunction of

relative clauses linking back to the stem structure, from Lexicon to sentence formation. The criteria are: (i) legitimacy within the computational system; (ii) no violation of NTC or ICC via added category labels; (iii) validity of SD generation for adjunction; and (iv) specificity of stem-structure linkage.

For Head-Adjunction, the stage where lexical items become SOs (i.e., √Root acquiring a head) is analyzed as {H, {H, √Root}} = <H, √Root>, yielding a workspace-visible syntactic object.(SO) Relations where an SO generates one SD and links to a host (Head-to-Head) are treated as modification relations (e.g., an adjectival Head linking to a nominal Head). Modification hierarchies develop via the second language factor, experience and parameter setting (Chomsky 2005, 2007). For conjunction ($n \geq 2$), cases with shared referential similarity (e.g., multiple adjectives modifying one noun) are distinguished from those generating two or more SDs without it. The generation of multiple SOs that are target to FS and share referential similarity may proceed in different ways, depending on the properties of SOs. For example, in analyzing the generation of multiple adjectival phrases, rather than deriving FS among the adjectives themselves, each adjective is first combined with the noun, and the nouns that bear referential similarity are then connected via a Link; this is because adjectives have a predicative nature and therefore require a theta(θ)-bearing target. This analysis follows the proposal of Fong and Oishi (2025).

Finally, cases that do not involve referential similarity but create more than one SD—specifically, the generation of relative clauses and their connection to the stem structure—are also examined. In such cases, for reasons of derivational efficiency, it is assumed that the head of SDs is linked to the head of a syntactic object in the stem structure whose Copy-generation has been completed.

This paper attempts to envisage how sentence formation can be traced from the Lexicon to its combination with the C-phase head, including the generation of Separate Dimensions (SDs), while acknowledging that several issues remain unsolved or require further clarification in future research.

Reference

Bode, S. 2020. *Casting a Minimalist Eye on Adjuncts*. New York: Routledge.

Borer, H. 2014. The category of Roots. In A. Alexiadou, H. Borer and F. Schäfer, eds., *The Syntax of Roots and the Roots of Syntax*, 112-148. Oxford: Oxford University Press.

Blümel, A. 2024. A case study in underspecification of UG: External pair merge of v and T. *Syntax* 27(1), 1-29.

Blümel, A., N. Goto and Y. Sugimoto. 2024. When the grammar doesn't mind which Merge it chooses. *Proceedings of the 39th West Coast Conference on Formal Linguistics*, 61-70.

Chomsky, N. 1957. *Syntactic Structures*. The Hague: Mouton.

Chomsky, N. 1970. Remarks on nominalization. In R. A. Jacobs and P. S. Rosenbaum, eds., *Readings in English Transformational Grammar*, 184-221. Waltham, MA: Ginn.

Chomsky, N. 1995. *The Minimalist Program*. Cambridge, MA: MIT Press.

Chomsky, N. 2000. Minimalist inquiries: The framework. In R. Martin, D. Michaels and J. Uriagereka, eds., *Step by Step: Essays on Minimalist Syntax in Honor of Howard Lasnik*, 89-155. Cambridge, MA: MIT Press.

Chomsky, N. 2001. Derivation by phase. In M. Kenstowicz and K. Hale, eds., *A Life in Language*, 1-52. Cambridge, MA: MIT Press.

Chomsky, N. 2004. Beyond explanatory adequacy. In A. Belletti, ed., *Structures and Beyond: The Cartography of Syntactic Structures, Volume 3*, 104-131. New York: Oxford University Press.

Chomsky, N. 2005. Three factors in language design. *Linguistic Inquiry* 36(1), 1-22.

Chomsky, N. 2007. Approaching UG from below. In U. Sauerland and H.-M. Gärtner, eds., *Interfaces + Recursion*

= *Language?*, 1-29. Berlin: Mouton de Gruyter.
Chomsky, N. 2008. On phases. In R. Freidin, C. P. Otero and M. L. Zubizarreta, eds., *Foundational Issues in Linguistic Theory: Essays in Honor of Jean-Roger Vergnaud*, 133-166. Cambridge, MA: MIT Press.
Chomsky, N. 2013. Problems of projection. *Lingua* 130, 33-49.
Chomsky, N. 2015. Problems of projection: Extensions. In E. Di Domenico, C. Hamann and S. Matteini, eds., *Structures, Strategies and Beyond: Studies in Honour of Adriana Belletti*, 1-16. Amsterdam: John Benjamins.
Chomsky, N. 2019a. Some puzzling foundational issues: The reading program. *Catalan Journal of Linguistics* Special Issue, 263-285.
Chomsky, N. 2019b. *UCLA Lecture: Invited Lectures at the Department of Linguistics, University of California, Los Angeles, 29-30 April and 1-2 May 2019*. Unpublished manuscript by R. Freidin. <lingbuzz/005485>.
Chomsky, N. 2021. Minimalism: Where are we now, and where can we hope to go. *Gengo Kenkyu* 160, 1-41.
Chomsky, N. 2024. The miracle creed and SMT. In G. Bocci, D. Botteri, C. Manetti and V. Moscati, eds., *Rich Descriptions and Simple Explanations in Morphosyntax and Language*, 1-41. Berlin/Boston: De Gruyter Mouton.
Chomsky, N., A. J. Gallego and D. Ott. 2019. *Generative Grammar and the Faculty of Language: Insights, Questions, and Challenges* (3rd version). Unpublished manuscript. <lingbuzz/003507>.
Epstein, S. D., H. Kitahara and D. Seely. 2016. Phase cancellation by external pair-merge of heads. *The Linguistic Review* 33(1), 87-102.
Fong, S. and M. Oishi. 2025. On the nature of FormSet. *Linguistic Variation,* 1-21. Advance online publication.
Fong, S., R. Berwick and J. Ginsburg. 2019. The combinations of Merge and workspace right-sizing. *Workshop paper*, Evolinguistics Workshop 2019. Page information unavailable.
Gallego, Á. J. and N. Chomsky. 2019. *The Faculty of Language: A Biological Object, a Window into the Mind, and a Bridge across Disciplines*. Unpublished manuscript. <lingbuzz/004901>.
Hilbert, D. 1923. *Über das Unendliche*. *Mathematische Annalen* 95, 161-190.
Kitahara, H. 2024. A review and reanalysis of some recent developments from GK to MC, and beyond. *Reports of the Keio Institute of Culture and Linguistic Studies* 55, 199-214.
Komachi, M., H. Kitahara, A. Uchibori and K. Takita. 2019. Generative procedure revisited. *Reports of the Keio Institute of Cultural and Linguistic Studies* 50, 269-283.
Kuratowski, K. 1921. Sur la notion de l'ordre dans la théorie des ensembles. *Fundamenta Mathematicae* 2(1), 161-171.
Marcolli, M., N. Chomsky and R. C. Berwick. 2025. *Mathematical Structure of Syntactic Merge: An Algebraic Model for Generative Linguistics*. Cambridge, MA: MIT Press.
Merchant, J. 2019. Roots don't select, categorial heads do: Lexical selection of PPs may vary by category. *The Linguistic Review* 36(3), 325-341.
Oishi, M. 2015. The hunt for a label. In H. Egashira, H. Kitahara, K. Nakazawa, T. Nomura, M. Oishi, A. Saizen and M. Suzuki, eds., *Untiring Pursuit of Better Alternatives*, 322-334. Tokyo: Kaitakusha.
Oku, S. 1998. *A Theory of Selection and Reconstruction in the Minimalist Perspective*. Doctoral dissertation, University of Connecticut, Storrs, CT, USA.
Omune, J. 2019. A Merge-based approach to head adjunction. *Journal of Inquiry and Research* 109, 167-185.
Pan, V. J. and Y. Du. 2024. A multi-dimensional derivation model under the free-MERGE system: Labor division between syntax and the C-I interface. *The Linguistic Review* 41(1), 85-117.
Shim, J. 2020. *The Minimalist Program: Unfinished Business*. Seoul: Bookk.

Shim, J. 2022. *Maximizing Simplicity: A Pursuit of Genuine Explanation in the Minimalist Program*. Seoul: Youkrack.

Sugimoto, Y. 2022. *Underspecification and (Im)possible Derivations: Toward a Restrictive Theory of Grammar*. Doctoral dissertation, University of Michigan, Ann Arbor, MI, USA.

Tourlakis, G. 2003. *Lectures in Logic and Set Theory,* Volume 2. New York: Cambridge University Press.

Examples in: English
Applicable Languages: English
Applicable Level: Tertiary

