Articles

On Hilbert's Epsilon Operator in FormSequence

Chenchen Song [1]

[1] *Department of Linguistics and Translation, Zhejiang University, Hangzhou, P.R. China.*

Biolinguistics, 2024, Vol. 18, Article e14061, https://doi.org/10.5964/bioling.14061

Received: 2024-02-28 • **Accepted:** 2024-05-27 • **Published (VoR):** 2024-07-23

Handling Editor: Kleanthes K. Grohmann, University of Cyprus, Nicosia, Cyprus

Corresponding Author: Chenchen Song, Department of Linguistics and Translation, School of International Studies, Zhejiang University, 866 Yuhangtang Road, Hangzhou 310058, P.R. China. E-mail: cjs021@zju.edu.cn

Abstract

This paper examines Chomsky's recently proposed and abandoned FormSequence operation and presents a middle-ground implementation of it in a way that conforms to the Strong Minimalist Thesis. Special attention is paid to the role of Hilbert's epsilon (ϵ) operator in this operation. I argue that while the ϵ-operator can give FormSequence its desired effect, the sequence-choosing mechanism should more adequately be attributed to the cognitive-computational context (mainly the interfaces) instead of Narrow Syntax. In other words, FormSequence is not entirely syntactic in nature but only partly so. I implement its syntactic part as repeated Pair Merge of a coordinator with a number of conjuncts, which yields a partially ordered set as output instead of a sequence. This implementation reconciles FormSequence with the Strong Minimalist Thesis and maintains a purely hierarchical syntactic module of human language. Furthermore, I compare the use of the ϵ-operator in FormSequence and its more established use in formal semantics and eventually promote a domain-general perspective on the fundamental cognitive procedure of sequence formation.

Keywords

Hilbert's epsilon operator, FormSequence, Pair Merge, Minimalist Program, third factor

1 Introduction

In two recent lectures, Chomsky (2019)[1] and Chomsky (2020), Chomsky proposed a new syntactic operation called FormSequence (FSQ). The operation is also discussed in Chomsky (2021), which is a revised and extended version of Chomsky (2020). Chomsky's view on FSQ has kept changing in the past few years. Following its initial introduction in Chomsky (2019), FSQ was acknowledged in Chomsky (2020) as an unavoidable departure from the Strong Minimalist Thesis (SMT). Chomsky (2021) further remarked that the operation might not be a departure from the SMT if it could be regarded as part of the "third factor" toolkit. Most recently, Chomsky (2023) eliminated FSQ due to its anti-SMT nature and replaced it with the similarly *n*-ary but order-free operation FormSet (FS). Simply put, FS takes a number of syntactic objects $X_1, \ldots, X_n$ as input and yields a set as output, as in (1a). Meanwhile, FSQ on Chomsky's (2021) conception further converts such a multimembered set into a sequence, as in (1b).[2]

(1) a. $\mathrm{FS}(X_1, \ldots, X_n) = \{X_1, \ldots, X_n\}$
 b. $\mathrm{FSQ}(\{X_1, \ldots, X_n\}) = \langle X_1, \ldots, X_n \rangle$

According to Chomsky (2023), FS is "a costless operation available freely for all inquiry" (p. 6) and is used in the domain of language to construct, among others, the Workspace and the Lexicon; FSQ, on the other hand, is the only operation in Chomsky (2021) that "departed from strict adherence to SMT" (p. 18) and therefore should be eliminated. While FSQ as defined above is indeed anti-SMT, this paper shows that it need not be totally abandoned but may be reformulated in an SMT-conforming way.

The FormSequence operation was introduced in Chomsky (2019) as an extension of Pair Merge (Chomsky, 2000), mainly to derive "unbounded unstructured coordination."[3] The phenomenon is illustrated in (2).

(2) I met someone [young, happy, eager to go to college, tired of wasting his time, ...] (Chomsky, 2019)

The bracketed coordination in (2) is "unbounded" in that it can go on and on without upper bound. And since no conjunct is in the scope of any other conjunct, the coordination is also "unstructured" in the technical sense that there is no asymmetrical c-command relation between the conjuncts.

To generate conjunct sequences like (2), Chomsky (2019) resorts to the Minimalist operation for adjunction: Pair Merge. Pair Merge takes two syntactic objects α and β as input and returns an ordered pair $\langle \alpha, \beta \rangle$ as output. It is different from Set Merge (Chomsky, 1995) in that sets are unordered while pairs are ordered, as in (3).

(3) a. $\mathrm{SetMerge}(\alpha, \beta) = \{\alpha, \beta\} = \{\beta, \alpha\}$
 b. $\mathrm{PairMerge}(\alpha, \beta) = \langle \alpha, \mathrm{\beta} \rangle \neq \langle \beta, \mathrm{\alpha} \rangle$

[1] The content from Chomsky (2019) in this paper is directly taken from Chomsky's 4th UCLA lecture (May 2, 2019). There is also an edited transcript of all of Chomsky's UCLA lectures by Robert Freidin available at https://lingbuzz.net/lingbuzz/005485.

[2] The description here has been simplified. See Section 3 for Chomsky's original description.

[3] Chomsky (2021) uses the alternative term "unbounded unstructured sequences."

For a coordination sequence S with conjuncts S_1, S_2, ..., S_n, Chomsky (2019) pairs up each conjunct S_i with a link element L_i.[4] He then places all the $\langle S_i, L_i \rangle$ pairs in a sequence, as in (4a). The first slot of the sequence is occupied by a conjunction denoted by "CONJ," and all the link elements are assumed to be identical. Chomsky (2020) presents the same idea in the slightly different form in (4b).[5]

(4) a. $\langle \mathrm{CONJ}, \langle S_1, L_1 \rangle, \ldots, \langle S_n, L_n \rangle \rangle$ (Chomsky, 2019)

b. $\langle (\&), X_1, \ldots, X_n \rangle$ (Chomsky, 2020)

(& is an optional conjunction and each X_i is a conjunct)

In Chomsky (2020), the operation that generates such a conjunct sequence is officially named FormSequence. Note that (4b) is just the instantiation of the general sequence-forming operation in (1b) in the case of coordination. In Chomsky (2021, p. 31), it is further specified that the sequence $\langle X_1, \ldots, X_n \rangle$ is formed from m members of the Workspace, possibly with $m < n$ since items in the sequence may repeat, as in *John, Mary, and John saw Tom, Jane, and Jill, respectively* (with the same John).[6]

Crucially, on Chomsky's (2019) conception, the formation of such a sequence involves the choice of a particular interconjunct ordering out of a set of alternatives:

> ... in order to generate these objects, you generate a set—[a] finite set. You pick out of—you form from that set a sequence, and it could be any sequence of elements, and there's in fact infinitely many possible sequences. You pick one out of those, and that sequence—S, call it—is the thing that you are then going to merge into the construction to proceed with the interpretation. This operation of **picking a particular element out of the set of sequences** is—there's formal ways of doing it which are familiar. Those of you who know some logic will recognize that this is **David Hilbert's epsilon operator**, which picks a single thing out of a set. It was part of his work on foundations of mathematics—[a] basic operation. So, it's

[4] See Manzini (2021) for potential externalizations of the link element. Thanks to an anonymous reviewer for this reference.

[5] In Chomsky (2021), the parentheses around "&" are again omitted.

[6] As an anonymous reviewer points out, sequences with repeated items are subject to certain special conditions, and their acceptability requires the aid of expressions like *respectively*. Thus, while Chomsky's sentence above is natural, the sentence #*John, Mary, and John are tired* (with the same John) is hardly acceptable. See Gawron and Kehler (2004) for further discussion. I will return to the issue of *respectively* in Section 3.2.

PsychOpen GOLD

> a straightforward operation, but it does have the property that it's indeterminate.
> (12:05–13:12, Chomsky, 2019, boldface mine)

Albeit only mentioned in passing, Hilbert's epsilon operator (henceforth ϵ-operator) evidently plays a special role in the above-quoted conception, because it is what ultimately fixes the sequence. Thus, even though the ϵ-operator is not explicitly mentioned in Chomsky (2020, 2021)—probably because the discussion of FSQ there is very brief and less technical—its role in a complete description of FSQ is nonnegligible. Therefore, to fully understand FSQ and its theoretical status, we need to first give the ϵ-operator a more careful examination.

The ϵ-operator was proposed in Hilbert and Bernays (1939) as a formal tool to create a term out of a formula,[7] as in (5).[8]

(5) $\epsilon x\,.\,\mathrm{F}(x)$

Here, x is an individual variable, F is a first-order predicate, and the entire ϵ-term means "an individual x such that F is true for x." For instance, if F is *apple*, then (5) denotes a particular apple. In Hilbert's original definition, ϵ-terms are nondeterministic, so there is no way to know precisely which individual (e.g., which apple) is picked. In this sense, (5) intuitively corresponds to the indefinite description "an F" (e.g., *an apple*).[9]

Chomsky does not clarify how exactly the ϵ-operator picks out a particular sequence. Equally unclear is the modular status of FSQ. The way it is introduced suggests that it is a syntactic operation. However, as I will show in Section 3, the sequence-generating procedure cannot be entirely within Narrow Syntax. It is best viewed as a hybrid operation instead, relying on both the syntax and the general cognitive-computational context (mainly the prederivational preparatory stage and the interfaces). In particular, I will propose that the narrow-syntactic part of FSQ does not involve sequences at all but only produces sets and pairs (more exactly sets of pairs), and that the sequence structure is only generated in syntax-to-interface (mainly but not exclusively syntax-to-PF) mapping. In this way, a reconciliation is reached between FSQ and SMT. Moreover, I will argue in Section 5 that the core mechanism of the reformulated FSQ, which still crucially involves the ϵ-operator, is in fact domain-general. This is in line with Chomsky's (2021, p. 35) suggestion that FSQ "can be regarded as part of the 'third factor' toolkit."

[7] This formula-to-term-converting role of ϵ more exactly makes it a *subnector* in Curry's (1963, pp. 32–33) terminology, but I will keep calling it an "operator" since that is the more common term in the literature.

[8] Alternative notations include $\epsilon x\mathrm{F}x$, $\epsilon x\mathrm{F}$, and $\epsilon_x\mathrm{F}(x)$. I stick to $\epsilon x\,.\,\mathrm{F}(x)$ in this paper.

[9] However, I will show in Section 3 that the use of the ϵ-operator in FSQ is not always nondeterministic (pace Chomsky, 2019).

Against the above background, I have two main objectives in this paper. First, I explore the exact role Hilbert's ϵ-operator plays in FSQ. Note that the sequence-fixing function of the ϵ-operator does not change whether FSQ is a syntactic operation or not. Since there is not yet any dedicated discussion of this issue in the literature to my knowledge, I will take Chomsky's original remarks as my point of departure and try to first make sense of the elements therein before presenting my own conception of FSQ. Second, I present a concrete implementation of my hybrid, middle-ground conception of FSQ, including both its syntactic and its nonsyntactic part. The effort to find a middle ground here is worthwhile because the core idea behind FSQ is more general than its particular use in coordination. Thus, Chomsky (2020) states that "wherever there is an XP, there would be a sequence," and simple phrases like *John saw Bill* and *John ran* are "just limiting cases of sequences." Again, this general idea remains insightful regardless of the modular status of FSQ—that is, whether it is syntactic or not.

The remainder of this paper is organized as follows. In Section 2, I introduce the mathematical-logical basics of the ϵ-operator in more detail. This section can be skipped by readers who are already familiar with this formal tool. In Section 3, I examine Chomsky's sketch of FSQ more closely and argue for a hybrid view of the operation. In particular, I highlight the relevance of the cognitive-computational context that the syntactic derivation is embedded in. In Section 4, I present my concrete implementation of the hybrid FSQ in Minimalist terms. In Section 5, I further show that the ϵ-based, reformulated FSQ rule has more applicability beyond syntax and even beyond the domain of language, which leads me to view it as a "third factor" strategy. Section 6 concludes.

2 The Epsilon Operator

Before examining the role of the ϵ-operator in FormSequence, I first introduce this mathematical-logical tool in more detail. This section is mainly based on Avigad and Zach (2020), Chatzikyriakidis et al. (2017), Leisenring (1969), and Slater's (n.d.) entry in the *Internet Encyclopedia of Philosophy*. There is controversy among logicians about certain aspects of the ϵ-operator (e.g., about its semantic model), but the general picture presented below suffices for current purposes.

Partly inspired by Russell's iota (ι) operator for definite descriptions (Whitehead & Russell, 1910), Hilbert proposed two generic element symbols—tau (τ) and epsilon (ϵ)—in the 1920s, as in (6).

(6) a. $\iota x \,.\, \mathrm{F}(x)$: the unique x that satisfies F
b. $\tau x \,.\, \mathrm{F}(x)$: an x that satisfies F when every individual does so
c. $\epsilon x \,.\, \mathrm{F}(x)$: an x that satisfies F when some individual does so

Unlike ι, which just says *the*, τ and ϵ return generic elements, and in principle there is no way to know exactly which individual is chosen. Thus, τ and ϵ are said to be nondeterministic or indeterminate. Of course, to define these symbols in a more complete manner, we also need to consider what happens when their preconditions are not met.[10] I will return to this issue shortly below.

The two symbols τ and ϵ are closely related to the two quantifiers $\forall$ and $\exists$ in predicate logic. Indeed, (6b) and (6c) are respectively a universal and an existential generic object with regard to F (Chatzikyriakidis et al., 2017), as in (7).

(7) a. $\mathrm{F}(\tau x \,.\, \mathrm{F}(x)) \equiv \forall x \,.\, \mathrm{F}(x)$
b. $\mathrm{F}(\epsilon x \,.\, \mathrm{F}(x)) \equiv \exists x \,.\, \mathrm{F}(x)$

While τ and ϵ had started their lives as different symbols, they are in fact mutually definable (see, e.g., Retoré, 2014 and Abrusci, 2017), so in the end Hilbert only kept ϵ.

Hilbert's original purpose was to find a consistent and complete axiom set for mathematics, first and foremost for arithmetic. The ϵ-operator was convenient for this purpose in that it eliminated the two quantifiers and could also replace the Axiom of Choice (see, e.g., Bernays, 1991/1958). Hilbert's program eventually failed due to Gödel's (1931) incompleteness theorems, according to which there is no consistent axiomatic system for arithmetic, and no sufficiently expressive system can prove its own consistency. But Hilbert's endeavor left us with a number of working tools, including ϵ.

The ϵ-operator as a logical symbol is equipped with an ambient syntax and a corresponding semantics. Its syntax is known as the ϵ-calculus, which is a minimal extension of predicate calculus, with ϵ being the only new symbol. The ϵ-calculus defines an ϵ-term for any formula, and for each ϵ-term there is a corresponding axiom as in (8).

(8) Axiom ϵ: $\mathrm{F}(t) \to \mathrm{F}(\epsilon x \,.\, \mathrm{F}(x))$
"If the object denoted by any term t has the property F at all, then the object denoted by $\epsilon x \,.\, \mathrm{F}(x)$ has it."

What (8) says is essentially that for any nonempty subset of the domain of discourse, we can choose a representative element from it, but that is just an instance of the Axiom of Choice. Indeed, the ϵ-operator is also known as the choice operator.

Hilbert did not give ϵ any semantics but merely used it as a syntactic tool to facilitate proof construction. Asser (1957) interpreted ϵ as a choice function in the model in (9).

(9) $\mathscr{M} := \langle \mathbf{J}, \mathbf{I} \rangle$ (based on Asser, 1957, pp. 33–34)

[10] Specifically, the preconditions for τ and ϵ are not met whenever something is not F for τ, which is common, and whenever nothing is F for ϵ, which is less common. In such cases, τ and ϵ typically return something arbitrary that does not satisfy F. Thanks to an anonymous reviewer for this clarification.

In (9), $\mathscr{M}$ is the model, **J** is its domain, and **I** is its constant-interpreting function. Asser sets $\mathbf{I}(\epsilon)$ to be a choice function Φ, which chooses an arbitrary element from each nonempty subset of **J**. Asser also took into account the empty set—the case where the *if*-condition in (8) is false (i.e., when the precondition in (6c) is not met). He suggested two solutions, one with a total choice function and the other with a partial one. On the total function solution, $\Phi(\varnothing)$ returns an arbitrary element ξ_0 of **J**—namely, an arbitrarily chosen individual in the whole world—and on the partial function solution it is undefined. Thus, assuming $[\![F]\!] = A \subseteq \mathbf{J}$, we have

(10) $$[\![\epsilon x\,.\,\mathrm{F}(x)]\!] = \Phi([\![\mathrm{F}]\!]) = \Phi(A) = \begin{cases} a \in A, & \text{if } A \neq \varnothing \\ \xi_0 \in \mathbf{J} \text{ or undefined}, & \text{if } A = \varnothing \end{cases}$$

As Leisenring (1969) points out, the total function solution suits Hilbert's original purpose better. This is also the sentiment in some later works on the philosophy of language. For instance, Slater (2017, p. 278) explicates that if there is no such x that satisfies $F(x)$, then the denotation of $\epsilon x\,.\,F(x)$ "is a fiction, which means it is simply a pragmatically chosen individual in the whole world at large."

In the above, my introduction of the ϵ-operator has been limited to the first order, where the ϵ-bound variable is of the individual type. But ϵ can also bind higher-order objects. See Ackermann (1925) and Hilbert and Bernays (1939, Supplement IV.A) for second-order ϵ-terms of the form $\epsilon f\,.\,F(f)$, where f is a function variable. As I will show in Section 3.2, the ϵ-term in FormSequence is also of the second order, with the ϵ-bound variable being of the sequence type. Furthermore, in the Konstanz School's use of ϵ in formal semantics to be reviewed in Section 5, the choice function itself becomes a matter of choice too, for which purpose context-indexed ϵ-terms of the form $\epsilon_i x\,.\,F(x)$ are used, where i stands for the particular context in which the ϵ-choice is made. See Mints and Sarenac (2003) and Leiß (2017) for the semantics of indexed ϵ-calculus.

In sum, the ϵ-operator had originally been proposed as part of a program to completely axiomatize mathematics. Despite the failure of that program, this formal tool has survived. The formal system ϵ lives in is the ϵ-calculus, and it is semantically interpreted as a choice function. In addition, the ϵ-operator can be flexibly applied to objects of varied types.

3 Dissecting FormSequence

3.1 Basics and Puzzles

As mentioned in Section 1, the FormSequence operation is an extension of Pair Merge, which takes two syntactic objects as input and returns an ordered pair of them as output. Chomsky (2004, pp. 117–118) further likens this to higher-dimensional structure building, suggesting that "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." This feature of Pair Merge is inherited in its FSQ extension:

> … we need an operation Pair Merge, which will also apply to the simple adjunct case like *young man*. *Young* will be adjoined to—will be attached to—*man*, but you don't see it in the labeling, okay, 'cause it's off in some other dimension. And the unbounded unstructured cases show you in effect that there are **unboundedly many dimensions** to what's going on up there [in the mind]. It's not two-dimensional like a blackboard. You can add any number of adjuncts at any point.
> (10:09–10:47, Chomsky, 2019, boldface mine)

I illustrate the dimension-expanding capacity of Pair Merge in (11), where n adjuncts $\alpha_1, \alpha_2, \ldots, \alpha_n$ are attached to a single host β in n different dimensions.

(11) PairMerge(α_1, β) = $\langle \alpha_1, \beta \rangle$
PairMerge(α_2, β) = $\langle \alpha_2, \beta \rangle$
…
PairMerge(α_n, β) = $\langle \alpha_n, \beta \rangle$

Recall from Section 1 that Chomsky (2019) uses the notation $\langle S_i, L_i \rangle$ for conjunct-link pairs. The angle brackets seem to suggest that each conjunct-link unit is a pair-merged object, especially in the representation in (4a). Chomsky further specifies that all links in the same coordination sequence "have to be identical" because we are "adjoining everything to the same point" (14:26–14:38, Chomsky, 2019).

Both Set Merge and Pair Merge are simple, basic operations in the Minimalist Program. By contrast, FSQ is not that simple. The quote from Chomsky (2019) in Section 1 specifies three steps for it: (i) generate a finite set; (ii) form from that set a set of sequences; and (iii) choose a particular sequence. Chomsky (2021), on the other hand, gives the following description:

> Generation of these [coordination] structures first selects $X_1, \ldots, X_m$ from WS [i.e., the Workspace], forming $Y = \{X_1, \ldots, X_m\}$, freely using the core operation of set-formation already discussed. Merging of & and FSQ yields $\langle \&, X_1, \ldots, X_n \rangle$, where the X_i's exhaust the elements of Y.
> (Chomsky, 2021, pp. 31–32)

This is a more complete description of the procedure in (1b), and the "core operation of set-formation" is just FS in (1a). Taking both Chomsky (2019) and Chomsky (2021) into account, I illustrate the FSQ operation with the coordinate phrase *young, tall, and happy* in (12). I temporarily ignore the "&" but will return to it shortly below.

(12) a. WS = {..., young, tall, happy, ...} (Workspace)

b. Y = {young, tall, happy} (a set of conjuncts)

c. {⟨young, tall, happy⟩, ⟨tall, young, happy⟩, ⟨happy, tall, young⟩, ...}
(a set of possible sequences)

d. ⟨young, tall, happy⟩ (a particular sequence)

As I mentioned in Section 1, since there is not yet any detailed discussion of FSQ in the literature, I take Chomsky's original remarks as my point of departure. However, this does not mean that I will confine myself to Chomsky's remarks. Below I highlight four puzzles in Chomsky's sketch of FSQ above.

First, the set Y in (12b) is multimembered and cannot be formed by Set Merge, which is binary by definition. The quote from Chomsky (2021) above suggests that the "core operation of set-formation" used to form Y is not Merge but FS, but then the question becomes whether it is desirable to admit FS as a standard narrow-syntactic operation on a par with Merge. Note that the two typical uses of FS mentioned in Chomsky (2023), in Workspace and Lexicon construction, are not really Narrow Syntax-internal but more exactly part of the preparatory stage of narrow-syntactic processes—namely, they belong to the computational context, an important notion that I will return to in Section 3.2.

Second, the set in (12c) contains "infinitely many possible sequences" on Chomsky's conception, but syntactic derivations are finitely defined, so it is unclear how this critical step can ever be made part of a derivation. There are ways to bypass this problem of infinitude, which deviate from Chomsky's original conception but nevertheless lead to viable alternative formulations of FSQ. I will return to these in Section 3.2 as well.

Third, the timing and fashion of the merger of the conjunction & is unclear, nor is it clear how it eventually becomes part of the sequence in (12d). Chomsky's (2021) wording "[m]erging of & and FSQ" seems to imply that the merger of & precedes the application of FSQ. But what is & merged with? The answer implied by Chomsky's sketch seems to be Y. If so, which type of Merge is applied to & and Y? If it is Set Merge, we get (13a); if it is Pair Merge, we get (13b).

(13) a. {&, {young, tall, happy}} (Set Merge of & and Y)

b. ⟨&, {young, tall, happy}⟩ (Pair Merge of & and Y)

Yet neither merged product can be readily targeted by FSQ to yield the desired final product ⟨&, young, tall, happy⟩.

Fourth, the status of the "link" mentioned in Chomsky (2019) is unclear. Recall from Section 1 that the product of FSQ specified in Chomsky (2019) is:

(14) $\langle \text{CONJ}, \langle S_1, L_1 \rangle, \ldots, \langle S_n, L_n \rangle \rangle$ (=4a)

The "CONJ" in Chomsky (2019) is equivalent to the "&" in Chomsky (2020, 2021), and on Chomsky's (2019) view, all the link elements L_1, L_2, …, L_n are identical. While these link elements are not included in Chomsky (2020, 2021), as in (15),[11] there is no indication of their abolition either. Thus, a question arises as to what their theoretical status is.

(15) $\langle (\&), X_1, \ldots, X_n \rangle$ (=4b)

Much of the discussion on FSQ in Chomsky (2020, 2021) is dedicated to "matching conditions among the X_i's," some of which involve "semantic/pragmatic properties" (Chomsky, 2021, p. 32). However, it is unclear to what extent such matching conditions are meant to replace the link elements.[12] On the one hand, the matching conditions are apparently a broader notion, in that they may be syntactic, semantic, or pragmatic,[13] while the link elements are unambiguously formal syntactic in nature. To illustrate, consider (16).

(16) a. John arrived at the hospital [in an ambulance] and [in a coma].
b. *John arrived at the hospital in [an ambulance and a coma].
c. ?John arrived at the hospital in [an ambulance and his street clothes].
(Chomsky, 2021, p. 31)

All three conjunct pairs in (16) match in syntactic categories, so the observed grammaticality difference can only be due to mismatches in semantic/pragmatic properties, if it is due to matching failure at all.[14] On the other hand, the matching conditions are more descriptive and empirically oriented than the link elements. The latter, as described

[11] Recall from (4) that the parentheses around "&" in Chomsky's notation indicate the optionality of the conjunction.

[12] An anonymous reviewer comments that in their perception Chomsky did essentially abandon the link elements around the time when he was writing Chomsky (2021), even if he had not rejected the idea behind them. But as the same reviewer points out, the task of fleshing out the link element–based analysis in the current paper is valuable no matter what Chomsky's personal thought process happens to have been.

[13] The broad nature of the matching conditions means that there may well be category mismatches among conjuncts, as long as the conjuncts match in the dimension that matters to the particular construction. In slightly different terms, Patejuk and Przepiórkowski (2023, p. 347) conclude that while "all conjuncts must satisfy any external restrictions on the syntactic position they occupy," the relevant restrictions may be either strict (resulting in category sameness) or underspecified/disjunctive (resulting in category mismatches). Thanks to an anonymous reviewer for this reference.

[14] An anonymous reviewer suggests that the reduced acceptability of (16b–c) may be due to the inability of a single semantic reading of *in* to appropriately apply to both conjuncts, which is particularly clear when one of the readings is idiomatic. In this case, *in* must directly select its co-idiomatic complement. This is a reasonable analysis, which as far as I am concerned implicitly references the semantic matching condition of being associated with the same reading of the shared governor.

in Chomsky (2019), are more like part of the abstract technicalities of the Minimalist machine.

Thus, the introduction of the matching conditions in Chomsky (2020, 2021) does not necessarily force us to give up the link elements in Chomsky (2019), and the theoretical relationship between the two concepts remains a puzzle on our way toward a full understanding of FSQ. To further clarify the issue, let's also consider the example in (17a) together with its syntactic representation in (17b).

(17) a. John arrived and met Bill.
b. C, {John$_3$, {INFL, ⟨&, {$_1$ *v*, {arrive, John$_1$}}, {$_2$ John$_2$, {*v**, {meet, Bill}}}⟩}}
(Chomsky, 2021, p. 33)

The sentence in (17a) underlyingly involves the coordination of *John arrived* and *John met Bill*. In (17b), Chomsky treats this as coordination at the *v*P/*v**P level, below the level of INFL. The part in (17b) that is relevant to our concern is the sequence demarcated by the angle brackets. We can make the connection between this concrete sequence and the abstract pattern in (15) more obvious by the explicit matching in (18).

(18) a. ⟨&, {$_1$ *v*, {arrive, John$_1$}}, {$_2$ John$_2$, {*v**, {meet, Bill}}}⟩ (from 17b)
b. ⟨&, X$_1$, …, X$_n$⟩ (=15)
c. X$_1$ = {$_1$ *v*, {arrive, John$_1$}}, X$_2$ = {$_2$ John$_2$, {*v**, {meet, Bill}}} (n = 2)

Chomsky (2019) takes the link elements to be *v* and *n*. This means that each conjunct and *v*/*n* together constitute a pair ⟨S$_i$, L$_i$⟩. However, neither X$_1$ nor X$_2$ in (18c) is in this format—both are just sets instead—nor is it fully clear what S$_1$ and S$_2$ correspond to in this newer presentation.

In sum, while the extension from Pair Merge to FSQ may be conceptually appealing, before the latter can be integrated into the working syntactician's toolkit, the above puzzles must be resolved, and the relevant technical details must be filled in. I will address these puzzles in the following sections and fill in the technical details based on a middle-ground conception of FSQ.

3.2 Sequence Generation in the Computational Context

Chomsky describes FSQ as an unavoidable departure from the Strong Minimalist Thesis, which "holds that I-language, the system that generates thought, keeps to Merge and language-independent principles, such as computational efficiency" (38:00–38:15, Chomsky, 2020). Chomsky (2021, p. 35) adds that FSQ "may not be a departure at all" if it "can be regarded as part of the 'third factor' toolkit." My middle-ground conception of FSQ is along similar lines. I do not treat FSQ as an entirely syntactic operation but view it as a hybrid operation instead—partly in the syntax and partly in the cognitive-computational context. The latter refers to the environment that syntactic derivation is embedded in—a fairly broad notion encompassing the prederivational preparatory stage (where Lexical Arrays, Workspaces, and the like are formed), the interfaces, the discourse, the encyclopedia of world knowledge, and even general memory. My rethinking of FSQ below mainly involves the interfaces, but since the other components are also occasionally referenced, I will keep using the umbrella term "cognitive-computational context" or just "computational context" for short.

The relevance of the cognitive-computational context to syntactic derivation is most evident at the prederivational preparatory stage. To derive a sentence like *Lily eats cookies*, the lexical items *Lily*, *eats*, and *cookies* as well as necessary abstract elements must first be selected from the Lexicon into a Lexical Array. This selection procedure does not take place in Narrow Syntax proper. Similarly, in the derivation of a sentence like *LILY eats cookies, not Tom*, the focused *LILY* is assumed to bear a [+FOCUS] feature, which is not intrinsic to the lexical item *Lily* but must be added before the derivation starts.[15] In fact, the relevance of the computational context in syntactic theory has become clearer in Chomsky's recent work, where the significance of the derivational Workspace is officially recognized.

Now let's consider the three steps of FSQ: conjunct set formation, sequence set formation, and sequence-choosing. None of these steps is narrow-syntactic in the canonical sense: the conjunct set is multimembered and therefore cannot be generated by Merge (and FS is not really Narrow Syntax–internal as mentioned above), the sequence set is similarly multimembered and additionally infinite, and the sequence-choosing step (i.e., the ϵ-operation) neither builds up syntactic objects nor manipulates them, unlike familiar syntactic rules (e.g., Merge, Agree). I assume that these three steps all take place in the computational context.

Beginning with conjunct set formation, I follow Chomsky's selection-plus-set-formation procedure (i.e., FS) quoted in Section 3.1 but additionally clarify that it takes place before the relevant subderivation (of the coordinate phrase) starts—namely, at the pre(sub)derivational preparatory stage. This stage works by different rules (e.g., FS) from those in Narrow Syntax proper (e.g., Merge). Moving on to sequence set formation, this step does not need to take place at the prederivational preparatory stage since the sequence set is not needed by syntactic derivation. Thus, I assume that this step simply takes place when it has to—namely, right before the sequence-choosing step. I will return to the timing of this latter step below. For now, let's first note that the set of all possible sequences generated from a given set A (assuming these are all finite) is formally just the free monoid A^* on A, whose identity element is the empty sequence $\langle\,\rangle$ and whose

[15] [+FOCUS], as well as other discourse features like [+TOPIC], falls in Chomsky's (1995, p. 227) category of *optional features*, which are "added arbitrarily as [a lexical item] LI enters the numeration."

monoid operation is concatenation (notated by ++ below). See (19) for the definitions of monoid and free monoid.

(19) a. A monoid $\langle M, \cdot, e\rangle$ is a set M equipped with an associative binary operation $\cdot$ and an identity element e such that $\forall m \in M$, $e \cdot m = m \cdot e = m$.

b. The free monoid on a set has as elements all finite sequences generated from zero or more elements of that set by concatenation.

Thus, the free monoid on $\{a, b\}$ is $\{\langle\,\rangle, \langle a\rangle, \langle b\rangle, \langle a, b\rangle, \langle a, a\rangle, \langle a, a, b\rangle, \ldots\}$, where singleton entries like $\langle a\rangle$ are generated by vacuous concatenations like $\langle a\rangle{+}{+}\langle\,\rangle$. The ϵ-operator then picks an element out of this free monoid, which is a particular sequence. I specify the ϵ-term in (20), where seq_A stands for “a sequence generated from A.”

(20) $\epsilon X\,.\,\mathrm{seq}_A(X)$

This ϵ-term chooses a sequence from the free monoid on A. I use the uppercase X as the ϵ-bound variable here because strictly speaking this is not a first-order ϵ-term, as we are selecting a sequence of items instead of an individual item. Regardless of that, the ϵ-operator works in the same way as in the first-order scenario.

While the above difference in variable type is less significant, there does exist a more significant difference between the ϵ-term in (20) and Hilbert’s original ϵ in mathematics. Recall from Section 2 that in Hilbert’s original work, the ϵ-choice is nondeterministic. That is, a term like $\epsilon x\,.\,F(x)$ can pick out any member of $[\![F]\!]$ without preference. This is how the ϵ-operator should work if we strictly follow its mathematical definition. Nevertheless, in the case of (20), some members of A^* are in practice never chosen under normal circumstances. These include, among others, sequences with many random repetitions and sequences with fewer components than members of the initial set. Thus, for the initial set $B = \{\mathrm{young}, \mathrm{tall}, \mathrm{happy}\}$, the sequences in (21) are examples of bad candidates (I temporarily ignore the link elements and the conjunction).

(21) $\langle$young, tall, young, young, tall, happy, happy$\rangle$, $\langle$young$\rangle$, $\langle\,\rangle$

In fact, under normal circumstances, the viable candidates of $\epsilon X\,.\,\mathrm{seq}_B(X)$ (henceforth its “normally viable candidates” for short) are just those in (22), which are the simple permutations of the initial set.

(22) $\langle$young, tall, happy$\rangle$, $\langle$young, happy, tall$\rangle$, $\langle$tall, young, happy$\rangle$, $\langle$tall, happy, young$\rangle$, $\langle$happy, tall, young$\rangle$, $\langle$happy, young, tall$\rangle$

Thus, we are forced to conclude that Chomsky’s intended use of the ϵ-operator in FSQ is not entirely consistent with the mathematical design of the symbol, for it is somewhat deterministic (pace Chomsky, 2019). But that, as far as I am concerned, is a legitimate departure, which may reflect a fundamental difference between mathematics and linguistics (as a cognitive science), for the kind of choice in the latter is less abstract and more susceptible to influence of the physical world. As I will show in Section 5, another more established use of the ϵ-operator in linguistics is even more deterministic and involves an explicit adaptation of Hilbert's original definition as well. Such adaptation is often necessary for tools borrowed from one discipline to another.

Given the normally viable candidates in (22), one can choose to deviate from Chomsky's original conception of the sequence set and define it just as the set of permutations of the initial set instead. This may be a more desirable definition, because it bypasses the problem of overgeneration inherent in Chomsky's definition of the sequence set as an infinite set. As for sequences with repeated items, they can be attributed to repetitions that already exist in the Workspace.[16] Another way to bypass the problem of infinitude is to fuse sequence set formation and sequence-choosing as one step by recursively applying the ϵ-operator. We can first select an element from the initial set, then select another element that is different from the one already selected, and do this repeatedly until we have selected as many elements as there are in the initial set.[17] This essentially also reduces the possible sequences to just permutations of the initial set, though it sounds more complicated than the first alternative definition above.

A major advantage of these alternative definitions is that they bypass the overgeneration problem. Nevertheless, they may induce new problems along the way, such as the problem of undergeneration. On the permutation-based definitions, the normally viable candidates of $\epsilon X \,.\, \mathrm{seq}_B(X)$ are hard-coded into the definition of FSQ. But it is uncertain to what extent such hard-coding is desirable, because there are still less normal circumstances after all. For instance, nothing strictly prohibits a speaker from starting with the initial set B above in mind but ending up only uttering *young* (e.g., waiting for the interlocutor to continue). In this situation, the output of syntactic derivation in the speaker's mind still contains all three conjuncts from the initial set, and they are still all sent to the LF interface for (unordered) semantic interpretation; it is just that an incomplete sequence is externalized at PF. Also note that this situation is different from PF deletion or ellipsis. In these latter situations, what is left out is part of the sentence string, which can usually be filled back in a determinate word order; by contrast, the unuttered conjuncts discussed here are not yet linearized (due to the way the ϵ-choice works) and so cannot be filled back in a determinate word order. See (23) for an illustration.

(23) a. Ellipsis: – Do you like candy? – No, I don't (like candy /*candy like).
b. Unuttered conjuncts: John is young (tall and happy / happy and tall).

[16] Thanks to an anonymous reviewer for suggesting this permutation-based alternative definition.

[17] Thanks to another anonymous reviewer for this alternative suggestion.

Thus, it seems that the speaker in principle does have more freedom in terms of the ϵ-choice, and the normally viable candidates above are probably better attributed to pragmatic factors such as Grice's (1975) maxims of conversation.[18] The undergeneration problem may well be avoidable via some modification of the permutation-based approaches above, but a proper evaluation of the various alternative definitions would take us too far afield. Therefore, I leave exploration in that direction to future research and still follow Chomsky's (2019) original conception of an infinite sequence set in the rest of this paper, assuming that the much smaller set of normally viable candidates in (22) is indeed due to pragmatic constraints.

There are also cases where the ϵ-choice is so severely constrained that the number of viable sequences is reduced to a minimum. Consider the sentence in (24).

(24) [John and Bill] saw [Tom and Mary] respectively. (Chomsky, 2019)

In (24), the adverb *respectively* imposes an interpretative interdependence on the two coordinate phrases, and the number of viable sequences is reduced to just two: ⟨John, Bill⟩/⟨Mary, Tom⟩ and ⟨Bill, John⟩/⟨Tom, Mary⟩. Chomsky mentions a similar effect associated with the expression *in that order*, which I illustrate in (25).[19]

(25) As for fruit, I like [apples, bananas, oranges, and strawberries], in that order.

This time, the number of viable sequences is reduced to just one. Note that the "viable sequences" above all refer to word order sequences. Assuming that the output of syntactic derivation has no linear order yet, such viability constraints on the ϵ-choice are essentially imposed on PF linearization. This means that the timing of the two sequence-generating steps (sequence set formation and sequence-choosing) is just that of syntax-to-PF mapping. That said, the sequence-generating procedure is not a PF-specific tool. I will further discuss this point in Section 5.

In fact, not only is the sequence-generating procedure not PF-specific, but the relevant mechanism is not purely PF-based either even when the ϵ-choice does take place at PF (as in the case of coordination). Let's still consider the viability constraints above. Such constraints are clearly not phonological in nature and cannot be intrinsic to the PF interface. Where do they come from then? In both (24) and (25), the relevant viability constraints seem to be from extralinguistic knowledge, which could be in the

[18] The discussion of concrete choices of sequences inevitably takes us out of the realm of competence and into that of use, but this is unproblematic because on the middle-ground conception of FSQ being developed here, the ϵ-choice of sequence is not made in Narrow Syntax.

[19] An anonymous reviewer mentions that the linear order of coordination may furthermore affect the possibility of anaphoric reference, as in the acceptability contrast between *John saw Mary and her brother* and *%John saw her brother and Mary.*

discourse or the speaker's general memory. In short, while the ϵ-choice of sequence in our discussion here takes place at the PF interface, the way the choice is made is crucially influenced by information that is not PF-internal but available in the more general cognitive-computational context. This is why I have not narrowly identified the sequence-generating part of FSQ as a PF rule in this section.

Also note that the constraints illustrated above are tied to the constraint-introducing expressions. Once we remove such expressions, the constraints are gone, as in (26).[20]

(26) a. [John and Bill] saw [Tom and Mary].
b. As for fruit, I like [apples, bananas, oranges, and strawberries].

The coordination ordering in neither sentence in (26) has interpretative significance this time, except for potential conversational implicatures. But implicature-induced sequential interpretation is defeasible, as in (27).

(27) a. I like [apples, bananas, oranges, and strawberries], but not in that order.
b. I went to [the post office, the market, and the bookstore], but not in that order.

Apart from expressions like *respectively* and *in that order*, world knowledge may also serve to guide the ϵ-choice (of word order sequence). Thus, when an initial set of conjuncts is associated with a conventionalized order, the most natural sequence is usually the one obeying that order, as in (28).

(28) a. The rainbow colors are [red, orange, yellow, green, blue, indigo, and violet].
b. The twelve months are [January, February, March, April, May, June, July, August, September, October, November, and December].
c. The Hogwarts houses are [Gryffindor, Hufflepuff, Ravenclaw, and Slytherin].

In all these examples, the ϵ-choice is made under the guidance of common knowledge, which as a type of encyclopedic information is readily available in the cognitive-computational context. Overall, as I have mentioned above, the ϵ-choice in FSQ is not always indeterminate (pace Chomsky, 2019) but may sometimes be (semi)deterministic.

In sum, the three steps in Chomsky's (2019) conception of FSQ all take place in the cognitive-computational context instead of Narrow Syntax. The conjunct set is formed at the prederivational preparatory stage, while the sequence set is formed at the syntax-to-interface mapping stage, right before the sequence-choosing step. The discussion above

[20] Chomsky (2023, p. 17) also acknowledges that "the ordering imposed by such phrases as *respectively* is a discourse property" (contra Chomsky, 2021). Thanks to an anonymous reviewer for pointing this out.

reveals that the ϵ-choice of sequence is mainly needed at the PF interface, which is the expected state of affairs considering the sequential nature of word order.

4 A Hierarchical Implementation

In the above, I have proposed a hybrid view of FSQ and discussed its nonsyntactic part in detail. In this section, I turn to its syntactic part and derive the actual coordinate phrase. Recall from Section 3 that both Pair Merge and FSQ create higher-dimensional objects. Whatever the exact definition of "higher-dimensional" is in the context of syntactic derivation, such objects clearly have properties that set-merged objects do not possess. Prior to Chomsky (2019), a multidimensional structure was already suggested for coordinate phrases in de Vries (2004, 2005) to answer the following question: "[H]ow can we represent the intuitive symmetry of coordination, and in particular, how can we prevent the first conjunct from c-commanding the second?" (de Vries, 2005, p. 92) De Vries proposed a "behindance" relation between nodes on the syntactic tree based on the idea that "conjuncts are behind each other in a three-dimensional structure" (ibid.). Accordingly, he proposed a "b-Merge" operation (i.e., Merge by behindance). Aside from technical details, de Vries's and Chomsky's ideas on coordination are basically the same.

In Chomsky's sketch of FSQ quoted in Section 3.1, each conjunct in a coordination is adjoined from a different dimension to the same point. I call this point the *pivot* of coordination.[21] The pivot is arguably not the link element, because it is pair-merged with each conjunct, whereas the link, on Chomsky's *v*/*n* conception, is subject to Set Merge, as in (17b). I repeat the pattern in (29).

(29) C, {John$_3$, {INFL, ⟨&, {$_1$ *v*, {arrive, John$_1$}}, {$_2$ John$_2$, {v^*, {meet, Bill}}}⟩}}

Since the pivot serves to hold the conjuncts together, it must lie at the intersection of all the dimensions in a coordination. And since there can be an unbounded number of conjuncts, the pivot must have flexible arity (i.e., it accepts any number of arguments). Based on these criteria, the obvious candidates for the pivot are just the logical connectives AND/OR, which I notate by the umbrella label Co.

As for Chomsky's (2019) link elements, recall that all link elements in a coordination are identified as one and the same. I take this to be a well-formedness condition (presumably an interface condition) and perhaps part of Chomsky's (2021) matching conditions.

[21] Note that this pivot point of multidimensional adjunction is *internal* to the coordination structure. In fact, it is just the (potentially abstract) coordinator itself on my conception, as specified below (29). Importantly, I am not suggesting that each conjunct is adjoined to some position in the derivational spine; that is not the right way to look at things as an anonymous reviewer rightly points out, especially when the conjuncts are not modifiers in the ambient sentence. Instead, the whole (non-modifier) coordinate phrase is normally set-merged into the external syntactic structure on my conception.

Satisfaction of this constraint may be what makes a multidimensional coordination labelable. If the link element is *v*, then the coordinate phrase's real label is *v*P rather than CoP—for clarity's sake I will notate such a phrase as Co*v*P. This scenario constitutes a special instance of the XP-YP case in Chomsky's (2013, 2015) Labeling Algorithm, where the label is provided by some shared feature(s).[22]

I make a further distinction between the notions "link" and "host." Recall that in a pair-merged object $\langle \alpha, \beta \rangle$, α is the adjunct and β is the host, and the category of the entire object is the same as that of β. Thus, the category of *young man* is the same as that of *man*. However, there is a crucial difference between this classical scenario of Pair Merge and the multidimensional Pair Merge involved in FSQ, which precludes an identification of the pivot (i.e., Co) with the host. The difference is that while the host in classical Pair Merge (e.g., *man* in *young man*) can be used on its own, Co cannot (i.e., it is syncategorematic); nor are the conjuncts modifiers of Co. Intuitively, if anything in a coordinate phrase projects at all, it should be the conjuncts' shared feature—namely, the link—rather than Co. Therefore, if we reserve the term "host" for the projecting component as in standard Pair Merge, then the host of each Co-XP pair should be XP instead of Co. This brings us to the somewhat peculiar conclusion that the multidimensional coordinate phrase is *multihosted*, with as many hosts as its dimensions. That said, these hosts are still not the same as those in classical Pair Merge, for Co is not a modifier of XP either, just as XP is not a modifier of Co. Below, I will use $\langle \mathrm{Co}, \mathrm{XP} \rangle$ to notate the Co-XP pair and call XP the host, though this designation is more expository than substantive. See Figure 1 for an illustration of the internal makeup of a multidimensional syntactic object. I use the superscript "L" to indicate an XP-internal element that serves as the link element and use dotted lines to indicate Pair Merge.

On this view of multidimensional coordination, I view the FSQ structure in Chomsky (2019), repeated in (30), as a high-level declaration of what FSQ is rather than an actual syntactic object.

(30) $\langle \mathrm{CONJ}, \langle \mathrm{S}_1, \mathrm{L}_1 \rangle, \ldots, \langle \mathrm{S}_n, \mathrm{L}_n \rangle \rangle$ (=4a)

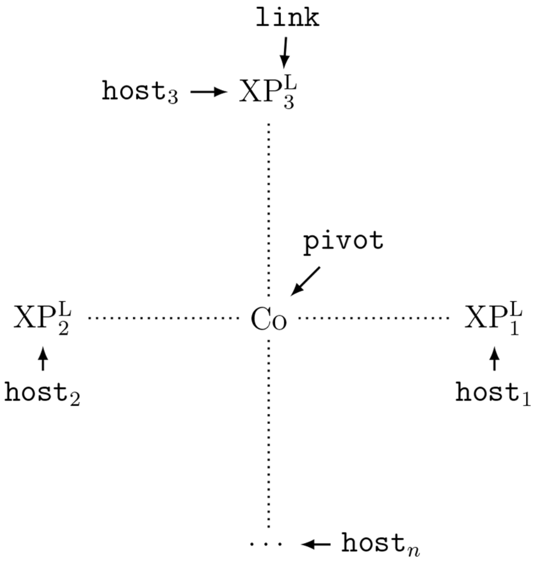
This notation declares that a link element can be identified for each conjunct. This formally amounts to defining a function $\lambda \mathrm{S}_i .\ \mathrm{L}_i$ that assigns to each conjunct term one of its subterms,[23] which in set talk is exactly a set of pairs $\{\langle \mathrm{S}_1, \mathrm{L}_1 \rangle, \langle \mathrm{S}_2, \mathrm{L}_2 \rangle, \ldots, \langle \mathrm{S}_n, \mathrm{L}_n \rangle\}$.

[22] On this conception of the link elements, matching requirements on them do not amount to the claim that conjuncts need to match in category, which is in line with Patejuk and Przepiórkowski's (2023) conclusion that matching conditions on coordination are imposed externally (see Note 13 above). Specifically, two constituents $\{_\mathrm{X} \ldots \mathrm{L} \ldots\}$ and $\{_\mathrm{Y} \ldots \mathrm{L} \ldots\}$ of different categories can be conjoined as long as L can be identified as their shared link element. Thanks to an anonymous reviewer for noting this.

[23] An anonymous reviewer suggests that this link-assigning function could be understood as a search procedure over the conjunct set, with the search target $\tau \in \{v, n\}$ if we adopt Chomsky's conception of link elements. This is

Figure 1

Internal Makeup of a Multidimensional CoP



Crucially, the pairs $\langle \mathrm{S}_i, \mathrm{L}_i \rangle$ here are not products of Pair Merge but just a metatheoretical notation. The alternative notation $\mathrm{S}_i^{\mathrm{L}_i}$ is less ambiguous.

Now that we have inspected the makeup of the multidimensional coordination structure, we can put everything together and give the syntactic part of FSQ an implementation. Recall from Section 3 that FSQ, on Chomsky's conception, comprises three steps: conjunct set formation, sequence set formation, and sequence-choosing. With the discussion above, we must add in a fourth step: coordinate phrase derivation.

In Section 3.1, I argued that the initial set of conjuncts required by FSQ could not be formed by Set Merge due to its multimembered nature. That said, it is perfectly viable as a set of initial ingredients for the derivation of a coordinate phrase, generated by FS as suggested by Chomsky. In this sense, the role of the initial set resembles that of a Lexical Subarray in Chomsky (2000), except that Lexical Subarrays contain items selected from the Lexicon, whereas the initial conjunct set contains syntactically derived conjunct phrases. The idea is that the conjuncts are prederived and reselected into the

indeed what the function is, especially on my above identification of the link-assigning procedure as a special instance of the Labeling Algorithm. The same reviewer further suggests that the procedure could probably be stated more precisely with Ke's (2023) formalism for Search. This is a promising future research direction.

quasi Lexical Subarray, which then serves as the starting point of the derivation of the coordinate phrase. This is well compatible with Chomsky's sketch. When analyzing the sentence *John arrived and met Bill* in (17), Chomsky (2020) remarked that "there are two parallel things generated separately. One of them is *arrive John*; the other is *John meet Bill*."

To implement parallel derivation, I adopt Zwart's (2007, 2009, 2011) theory of layered derivation and assume that each conjunct is derived in a separate layer. On Zwart's theory, complex noncomplements like subjects are constructed in separate layers before they join the main layer, and one layer's output can be included in another layer's input. Johnson (2003) calls this mechanism "renumeration." Besides complex subjects, Zwart suggests that several other constructions can be given a layered-derivational analysis, including coordination. Viewed from the current layer, elements derived in previous layers "have a dual nature" since they are "complex in the sense that they have been derived in a previous derivation" but "single items in that they are listed as atoms in the numeration for a subsequent derivation" (Zwart, 2009, p. 173). I illustrate the layered-derivational implementation of multidimensional coordination with the sentence in (31).

(31) The man [kicked the ball, slipped, and fell].

First, we derive the three conjuncts in three separate layers, as in (32). I follow Chomsky's presentation in (17b) and treat coordinate verbal predicates as full-fledged *v*Ps or v^*Ps.

(32) a. **Layer 1**: {$_1$ the man$_1$, {v^*, {kick, the ball}}}
b. **Layer 2**: {$_2$ v, {slip, the man$_2$}}
c. **Layer 3**: {$_3$ v, {fall, the man$_3$}}

Next, we renumerate the three conjuncts into the Lexical Subarray of a new layer (via FS), which also contains Co (normally inherited from the overall Lexical Array), and do multidimensional adjunction.

(33) **Layer 4**

a. Lexical Subarray: {Co, {$_1$ …}, {$_2$ …}, {$_3$ …}}
b. Multidimensional adjunction: ⟨Co, {$_1$ …}⟩, ⟨Co, {$_2$ …}⟩, ⟨Co, {$_3$ …}⟩

Finally, we renumerate the coordinate phrase into the Lexical Subarray of the main layer (again via FS), which also contains INFL and C. And the derivation continues for the rest of the sentence.

(34) **Layer 5** (main layer)

a. Lexical Subarray: {Co*v*P, INFL, C, …}
b. Further derivation: C, {the man$_4$, {INFL, ⟨Co, $\{_1 \ldots\}$, $\{_2 \ldots\}$, $\{_3 \ldots\}$⟩}}

On Chomsky's (2020, 2021) conception, the sentential subject *the man*$_4$ can be raised to its surface position (via Internal Merge) from any of the three conjuncts indexed 1, 2, and 3. Its lower occurrences that do not raise are indistinguishable from copies of the raised occurrence and therefore get deleted across the board. Chomsky does not specify how exactly this cross-dimensional raising takes place. Here I tentatively suggest that the pivot Co, being the connection between each of the dimensions in Co*v*P and the main dimension of the derivation, may serve as a bridge or "edge" for cross-dimensional movement. This, plus the fact that the derivation of the coordinate phrase has its own Lexical Subarray, points to the possibility that a coordinate phrase is a phase in the sense of Chomsky (2001 et seq.).

The overall derivational procedure above is summarized in (35), where I respectively use $\otimes$ and $\Rrightarrow$ to indicate parallel and sequential interlayer relations.

(35) **(Layer 1 $\otimes$ Layer 2 $\otimes$ Layer 3) $\Rrightarrow$ Layer 4 $\Rrightarrow$ Layer 5**

The above derivation is hard to illustrate with conventional tree diagrams due to its multilayeredness and multidimensionality, but it can be easily illustrated by a proof tree, as in Figure 2.[24] The final line of the proof resembles Chomsky's (2020) representation in (17b), except that I do not treat the multidimensional Co*v*P as a sequence in Narrow Syntax but treat it as a set of pairs with a shared component Co. This structure meets the definition of a *partial order*, more exactly one where one element (i.e., Co) is ranked above everything else. Thus, we can view multidimensional Pair Merge as an operation that takes a certain kind of numeration—one with a pivot and some syntactic objects with a common link—as input and yields a partially ordered set as output. This is exactly what happens in the step labeled "pp." In particular, since the three conjuncts themselves are not related to one another by the partial order, the partially ordered set in (36a) can be more compactly written as (36b), with Co being ranked above a plain set.

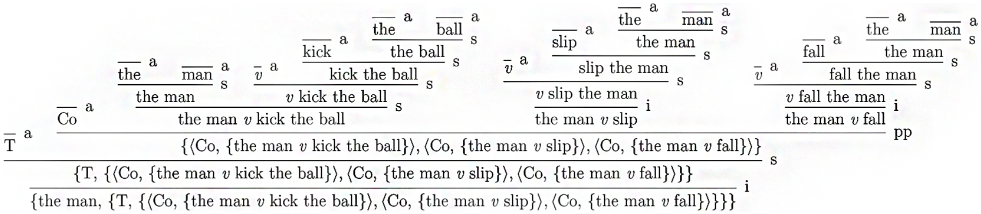
(36) a. {⟨Co, {the man *v* kick the ball}⟩, ⟨Co, {the man *v* slip}⟩, ⟨Co, {the man *v* fall}⟩}
b. ⟨Co, {{the man *v* kick the ball}, {the man *v* slip}, {the man *v* fall}}⟩

This more compact way of notation in (36b) is also that used in (13b). Thus, at a certain level of abstraction, pair-merging a whole conjunct set to Co is essentially the same as pair-merging each conjunct to Co individually, as they both yield the same partially

[24] The set-theoretic representations in Figure 2 have been simplified for expository convenience.

Figure 2

An Example Proof Tree for Multilayered, Multidimensional Derivation


the man
kick the ball
v kick the ball
the man v kick the ball
slip the man
v slip the man
the man v slip
fall the man
v fall the man
the man v fall
Co
T
{⟨Co, {the man v kick the ball}⟩, ⟨Co, {the man v slip}⟩, ⟨Co, {the man v fall}⟩}
{T, {⟨Co, {the man v kick the ball}⟩, ⟨Co, {the man v slip}⟩, ⟨Co, {the man v fall}⟩}}
{the man, {T, {⟨Co, {the man v kick the ball}⟩, ⟨Co, {the man v slip}⟩, ⟨Co, {the man v fall}⟩}}}
a
s
i
pp

Note. a = axiom; s = Set Merge; i = Internal Merge; pp = multidimensional Pair Merge.

ordered set. Intuitively, Chomsky’s higher-dimensional talk itself may be a metaphor for a non-plain-set (yet still SMT-conforming) structure supported by natural language syntax. Whether there still exist other such structures apart from the partially ordered set is an interesting question. Finally, note that independently of whether or not the result of “multidimensional Pair Merge” (pp) is really multidimensional, c-command does not obtain between the conjuncts, as they are not (contained in) sisters of each other.

The proof tree in Figure 2 illustrates the two steps conjunct set formation and coordinate phrase derivation. Now I connect these to the two other steps of FSQ: sequence set formation and sequence-choosing. As mentioned in Section 3.2, since the coordinate phrase in the syntactic output is not a sequence, the sequence-generating procedure can only take place at the interfaces. At the PF interface, this is just normal word order linearization; at the LF interface, this is interpreting the coordinate phrase in a particular order. Obviously, the sequence-generating procedure is more imperative at PF than at LF, because a coordinate phrase, like any other type of phrase, must always be uttered as a string, but it is not necessarily tied to a specific interpretative order (unless there is some special linguistic device like *respectively*). This suggests that the sequence-generating procedure may be separately executed (when needed) at the PF/LF interfaces. In most cases, an arbitrary sequence is generated at PF, while no sequence is generated at LF. But sometimes the two interfaces may even generate opposite sequences, as illustrated in (37).[25]

(37) a. I went to [the post office, the market, and the bookstore], in reverse order.
b. PF: ⟨the post office, the market, the bookstore⟩
c. LF: ⟨the bookstore, the market, the post office⟩

[25] I am abstracting away from many formal details here for expository convenience.

Assuming that the generation of every sequence involves an ϵ-choice, we can further conclude that ϵ-choices at the PF and the LF interface can in principle be influenced by different factors, though the two interfaces still both have access to the generally available contextual information (e.g., common knowledge) discussed in Section 3.2.

In sum, my implementation of FSQ has two parallel processes: a coordinate phrase is derived in Narrow Syntax as a partially ordered set, and a sequence is chosen at the PF/LF interface via the ϵ-operator. In this way, we can both obtain the sequence and stick to a purely hierarchical, SMT-conforming syntactic module. Before concluding this section, I want to highlight the fact that due to the c-command-free nature of the syntactically derived coordinate phrase, it cannot be linearized at PF by the usual means (such as Kayne's 1994 Linear Correspondence Axiom and its variants). Therefore, the ϵ-choice of sequence is not just a feasible solution but perhaps *the* solution to the linearization of "unbounded unstructured coordination." Accordingly, a rule like FSQ is still useful in the Minimalist Program, if not in Narrow Syntax per se, and it is not an ideal move to abandon it altogether.

5 FormSequence Beyond Syntax

My discussion of FSQ so far has focused on its original application in Chomsky's lectures. In this section, I show that it has wider application both within and outside of the domain of language. I first review a more established use of the ϵ-operator in formal semantics, arguing that the core idea in FSQ is involved there too, and then present an application of the same sequence-generating strategy in the cognitive process of prioritization. Such domain-general applicability of FSQ (in a generalized sense) is in line with Chomsky's (2021) suggestion that it could be a "third factor" strategy.

The formal semantic case I review is the Konstanz School's semantic theory of (in)definite NPs and intersentential anaphora. These empirical phenomena are illustrated in (38).

(38) a. (In)definite NPs: the man, a man, ...
b. Intersentential anaphora: A man comes. The man / He smokes.

Dissatisfied with the iota-based approach to *the*, the quantificational approach to *a*, and the E-type pronoun approach to intersentential anaphora (see Egli & von Heusinger, 1995; von Heusinger, 1997a; and Retoré, 2014, for details), researchers in the Konstanz School (most representatively Klaus von Heusinger) developed a unified theory for all three phenomena above based on an extension of the ϵ-operator. Specifically, they equipped ϵ with a context index, thus assigning a dedicated choice function to each context. Given a context c, the classical ϵ-term $\epsilon x\,.\,\mathrm{F}(x)$ becomes $\epsilon_c x\,.\,\mathrm{F}(x)$, which picks out the most salient element in $[\![\mathrm{F}]\!]$ in c. On the semantic side, ϵ_c is interpreted by an indexed choice function Φ_c. As a result, there is "not *one single* choice function but *a whole family* of them indexed with situations" (Egli & von Heusinger, 1995, p. 134).

I will not go into the technical details of the Konstanz School's analysis. Interested readers can consult Egli and von Heusinger (1995) and von Heusinger (1997a, 1997b, 2000, 2002, 2004, 2013). My focus here is just on the notion *salience* their theory relies on (originally from Lewis, 1979). With this notion, the descriptive material in a definite NP (e.g., *man* in *the man*) denotes a set as usual, but this set is furthermore equipped with a salience-based ranking of its members, which is essentially a discourse-determined sequence. Note that unlike in the cases considered in previous sections, in this case the ϵ-choice is tied to the LF interface instead of the PF interface. This clearly demonstrates that the ϵ-choice of sequence is not a PF-specific tool.

As Egli and von Heusinger (1995, p. 134) point out, the context-indexed ϵ-operator (i.e., their "global" ϵ-operator) does two jobs at once: ranking $[\![F]\!]$ and choosing its most salient element. This is reminiscent of the ϵ-operation in FSQ. There, too, we must first prepare the ambient set (i.e., the sequence set) and then make the choice. The similarity between the Konstanz School's and Chomsky's use of ϵ goes beyond the level of the basic procedure. Importantly, the ϵ-choice in neither use is consistently nondeterministic. In fact, the Konstanz School's use of ϵ is always deterministic under the influence of the salience ranking; Chomsky's use, on the other hand, may be nondeterministic or (semi)deterministic depending on whether or not there are lexical/pragmatic constraints at work (and how strong the constraints are), as we have seen in Section 3.2. By contrast, in Hilbert's original definition, the ϵ-operator is strictly nondeterministic.

Another similarity between the Konstanz School's theory and Chomsky's theory I want to highlight concerns the way the former's salience ranking is formed (which they do not specify). A ranking is essentially a sequence. Hence, it in principle can be generated by a generalized FSQ rule, via the ϵ-term in (39).

(39) $\epsilon_{\langle \text{salience},\, c\rangle} X\,.\ \text{seq}_{[\![\text{man}]\!]^*}(X)$

In (39), $[\![\text{man}]\!]$ is the set to be ranked, and the tuple $\langle \text{salience}, c\rangle$ specifies the conditions that together influence the ϵ-choice. In this case, the choice is influenced by the salience parameter and the context c. As in Section 3.2, I use $[\![\text{man}]\!]^*$ to denote the free monoid on $[\![\text{man}]\!]$, which contains various sequences of men. The second-order indexed ϵ-term in (39) precisely picks out the sequence ordered by salience in context c.[26]

To take full advantage of the cross-theoretic similarity, we can reformulate the Konstanz School's analysis of definite NPs with two steps of ϵ-operation, as in (40).

[26] An anonymous reviewer points out that the Konstanz School's (de facto) application of FSQ has the same problem of overgeneration that Chomsky's application in coordination has, in that the former also needs to rule out sequences that include superfluous repetitions. Again, the permutation-based alternative conception can help bypass the problem.

(40) *the F*, context *c*

a. Choose sequence: $\epsilon_{\langle \text{salience},\ c\rangle} X\,.\ \text{seq}_{[\![\text{F}]\!]^*}(X)$

b. Choose element: $\epsilon_c x\,.\ [\![\text{F}]\!]^{\langle \text{salience},\ c\rangle}(x)$

Both steps in (40) involve a deterministic ϵ, and they furthermore share the context parameter *c*. Here, too, the sequence-generating procedure in (40a) takes place in the cognitive-computational context (more exactly in the discourse), which is then readily accessible to the element-choosing procedure (i.e., the interpretation of *the F*) in the semantic module of language.

The cross-theoretic comparison above potentially reveals a more general principle—that is, whenever a sequence is needed in the cognitive-computational context, whether by phonological linearization, semantic interpretation, or something else, it can be generated by a generalized FSQ rule. In fact, this generalized rule is applicable outside of the domain of language as well. Consider the cognitive process of prioritization for example, which is an important ability in real-world multitasking, especially when there is time pressure (Bai, 2017). There are both multitasking scenarios that require optimal routines and scenarios that require spontaneous prioritization. These respectively correspond to sequences with conventionalized ordering (e.g., *The twelve months are January, ...*) and sequences with more arbitrary ordering (e.g., *I like apples, bananas, and strawberries*) in the linguistic domain. An example of routinized prioritization is the ABC (Airway, Breathing, Circulation) protocol in first aid, and an example of spontaneous prioritization is that in household chores. I present their "initial conjunct sets" in (41).

(41) a. $P = \{\text{Airway}, \text{Breathing}, \text{Circulation}\}$

b. $Q = \{\text{cleaning floor}, \text{washing dishes}, \text{taking out trash}, \text{cooking}, \text{feeding pets}\}$

The items in (41a) are associated with a conventional order, while those in (41b) are not. Thus, the ranking of *Q* is more context-dependent than that of *P*, and accordingly, the ϵ-choice of sequence for *P* is more deterministic than that for *Q*, though the ϵ-choice in Q is not completely indeterminate either. I give the two ϵ-terms in (42).

(42) a. $\epsilon_{\langle \text{convention},\ c\rangle} X\,.\ \text{seq}_{[\![\text{P}]\!]^*}(X)$

b. $\epsilon_{\langle c\rangle} X\,.\ \text{seq}_{[\![\text{Q}]\!]^*}(X)$

As before, I use a subscript on ϵ to indicate parameters influencing the ϵ-choice. The ϵ-term in (42a) chooses a sequence of first-aid steps based on both convention and the context (the latter is included because there are occasions where the conventional protocol must be altered), while the ϵ-term in (42b) chooses a sequence of chores solely based on the context, which covers the hygienic state of the house, the time, the pets' level of hunger, and so on. In both situations, the sequence-choosing process takes place in the agent's mind, and the chosen sequences are turned into action instead of language.

The domain-general applicability of the generalized FSQ rule points to the possibility of its being identified as a "third factor" strategy (Chomsky, 2005). The (semi)deterministic choice of sequence is essentially a matter of decision-making, and the domain-general nature of sequence construction and decision-making is uncontroversial. The former is "ubiquitous in our lives" and "important in intact cognitive processing" (Jaswal, 2017, pp. 5–6), and the latter is a high-level process that "builds on more basic cognitive processes such as perception, memory, and attention" and "is uniquely identified by … the process of *choice*" (Gonzalez, 2017, p. 249).

6 Conclusion

In this paper, I examined Chomsky's recently proposed and abandoned operation FormSequence, which he mainly used to derive "unbounded unstructured coordination." I have paid special attention to the role of Hilbert's ϵ-operator in the operation, which is mentioned by Chomsky as a way to formally implement FSQ. Specifically, the ϵ-operator is what fixes the sequence, by choosing a particular sequence from a set of possible alternatives. Given this important function, we need to first have a proper understanding of the ϵ-operator before we can fully grasp the theoretical status of FSQ. I reviewed the historical background and mathematical-logical basics of this formal tool in Section 2.

Chomsky's various remarks on FSQ are not always clear or consistent (Section 3.1). Therefore, while I have taken Chomsky's original remarks as my point of departure, much of the content in this paper is my own development. In particular, I have departed from Chomsky's conception of FSQ as an ordinary syntactic structure-building operation (which is what has led him to abandon it) and proposed a new, middle-ground conception, viewing FSQ as a hybrid operation that is only partly syntactic (Section 3.2). Crucially, the syntactic part of FSQ on my conception does not involve sequence generation at all but is solely based on Set Merge and Pair Merge, while its sequence-generating part, including the ϵ-operator, is relocated to the cognitive-computational context, mainly to the interfaces (and especially to the PF interface). This middle-ground conception of FSQ reconciles it with the Strong Minimalist Thesis and makes a total elimination of it from the Minimalist Program unnecessary.

Building on the middle-ground conception, I presented a concrete Minimalist implementation of FSQ (Sections 3.2–4). In its original application scenario of coordination, FSQ consists of four steps: (i) conjunct set formation, (ii) sequence set formation, (iii) sequence-choosing, and (iv) coordinate phrase derivation. On my conception, step (i) takes place at the prederivational preparatory stage; steps (ii)–(iii), at the interfaces; and step (iv), in Narrow Syntax. I implemented step (i) as Lexical Subarray formation with renumeration (via FormSet), step (ii) as free monoid formation, step (iii) as the

ϵ-operation, and step (iv) as multidimensional Pair Merge (of a coordinator and a number of conjuncts).

Furthermore, I discussed the wider application of FSQ beyond Chomsky's immediate concern (Section 5). To illustrate, I demonstrated that the same strategy could be used to generate the salience ranking in the Konstanz School's formal semantic theory of (in)definite NPs, which involves a more established linguistic use of the ϵ-operator. Interestingly, neither the Konstanz School's nor Chomsky's use of the ϵ-operator is always nondeterministic—in fact, the Konstanz School's use of it is always deterministic—unlike in Hilbert's original definition, where ϵ is strictly nondeterministic. Finally, I showed that FSQ in a generalized sense could be applied beyond the domain of language, such as in the cognitive process of prioritization. This indicates the possibility that the operation may be a "third factor" strategy as suggested in Chomsky (2021) and further weakens the desirability of a total elimination of it from the Minimalist toolkit.

Funding: The author has no funding to report.

Acknowledgments: I am grateful to Kleanthes Grohmann for his patient editorial help and to the two *Biolinguistics* reviewers for their constructive comments. Thanks to Matthew Reeve for his timely advice during my search for a home for this article. I would also like to thank all previous reviewers who commented on earlier versions of the article and helped improve its quality. All remaining errors are my own.

Competing Interests: The author has declared that no competing interests exist.

References

Abrusci, V. M. (2017). Hilbert's τ and ϵ in proof theoretical foundations of mathematics: An introduction. *The IfCoLog Journal of Logics and Their Applications, 4*(2), 257–274. http://www.collegepublications.co.uk/downloads/ifcolog00011.pdf

Ackermann, W. (1925). Begründung des 'tertium non datur' mittels der Hilbertschen Theorie der Widerspruchsfreiheit. *Mathematische Annalen, 93*(1), 1–36. http://eudml.org/doc/159075

Asser, G. (1957). Theorie der logischen Auswahlfunktionen. *Mathematical Logic Quarterly, 3*(1–5), 30–68. https://doi.org/10.1002/malq.19570030104

Avigad, J., & Zach, R. (2020). The epsilon calculus. In E. N. Zalta (Ed.), *The Stanford encyclopedia of philosophy* (Fall 2020 ed.). Metaphysics Research Lab. https://plato.stanford.edu/archives/fall2020/entries/epsilon-calculus/

Bai, H. (2017). *Cognitive processes of prioritization in multitasking* [Doctoral dissertation, Mississippi State University]. https://www.proquest.com/openview/093fc856283e26407071a065d7a636ed

Bernays, P. (1991/1958). *Axiomatic set theory.* Dover Publications.

Chatzikyriakidis, S., Pasquali, F., & Retoré, C. (2017). From logical and linguistic generics to Hilbert's tau and epsilon quantifiers. *The IfCoLog Journal of Logics and Their Applications, 4*(2), 231–256. http://www.collegepublications.co.uk/downloads/ifcolog00011.pdf

Chomsky, N. (1995). *The minimalist program.* MIT Press.

Chomsky, N. (2000). Minimalist inquiries: The framework. In R. Martin, D. Michaels, & H. Uriagereka (Eds.), *Step by step: Essays on minimalist syntax in honor of Howard Lasnik* (pp. 89–156). MIT Press.

Chomsky, N. (2001). Derivation by phase. In M. Kenstowicz (Ed.), *Ken Hale: A life in language* (pp. 1–52). MIT Press. https://doi.org/10.7551/mitpress/4056.001.0001

Chomsky, N. (2004). Beyond explanatory adequacy. In A. Belletti (Ed.), *Structures and beyond* (pp. 104–131). Oxford University Press. https://doi.org/10.1093/oso/9780195171976.003.0004

Chomsky, N. (2005). Three factors in language design. *Linguistic Inquiry, 36*(1), 1–22. https://doi.org/10.1162/0024389052993655

Chomsky, N. (2013). Problems of projection. *Lingua, 130*, 33–49. https://doi.org/10.1016/j.lingua.2012.12.003

Chomsky, N. (2015). Problems of projection: Extensions. In E. Di Domenico, C. Hamann, & S. Matteini (Eds.), *Structures, strategies and beyond: Studies in honour of Adriana Belletti* (pp. 1–16). John Benjamins. https://doi.org/10.1075/la.223.01cho

Chomsky, N. (2019, May 2). *Lecture 4* [UCLA lecture series]. https://linguistics.ucla.edu/noam-chomsky

Chomsky, N. (2020, November 22). *Lecture at the 161st Meeting of the Linguistic Society of Japan.* https://youtu.be/X4F9NSVVVuw

Chomsky, N. (2021). Minimalism: Where are we now, and where can we hope to go [Revised and extended version of Chomsky (2020)]. *Gengo Kenkyu, 160*, 1–41. https://doi.org/10.11435/gengo.160.0_1

Chomsky, N. (2023). The Miracle Creed and SMT. To appear in M. Greco & D. Mocci (Eds.), *A Cartesian dream: A geometrical account of syntax in honor of Andrea Moro.* http://www.icl.keio.ac.jp/news/2023/Miracle%20Creed-SMT%20FINAL%20%2831%29%201-23.pdf

Curry, H. B. (1963). *Foundations of mathematical logic.* McGraw-Hill.

de Vries, M. (2004). Parataxis as a different type of asymmetric Merge. *Proceedings of the Interfaces Conference hosted by the International Conference on Advances in the Internet, Processing, Systems, and Interdisciplinary Research (IPSI).* https://www.academia.edu/73795026/Parataxis_as_a_different_type_of_asymmetric_Merge

de Vries, M. (2005). Coordination and syntactic hierarchy. *Studia Linguistica, 59*, 83–105. https://doi.org/10.1111/j.1467-9582.2005.00121.x

Egli, U., & von Heusinger, K. (1995). The epsilon operator and E-type pronouns. In U. Egli, P. E. Pause, C. Schwarze, A. von Stechow, & G. Wienold (Eds.), *Lexical knowledge in the organization of language* (pp. 121–141). John Benjamins. https://doi.org/10.1075/cilt.114.07egl

Gawron, J. M., & Kehler, A. (2004). The semantics of respective readings, conjunction, and filler-gap dependencies. *Linguistics and Philosophy, 27*, 169–207. https://doi.org/10.1023/B:LING.0000016452.63443.3d

Gödel, K. (1931). Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I. *Monatshefte für Mathematik und Physik, 38*, 173–198. https://doi.org/10.1007/BF01700692

Gonzalez, C. (2017). Decision-making: A cognitive science perspective. In S. F. Chipman (Ed.), *The Oxford handbook of cognitive science* (pp. 249–264). Oxford University Press. https://doi.org/10.1093/oxfordhb/9780199842193.013.6

Grice, H. P. (1975). Logic and conversation. In P. Cole & J. L. Morgan (Eds.), *Speech acts* (pp. 41–58). Academic Press.

Hilbert, D., & Bernays, P. (1939). *Grundlagen der Mathematik* (Vol. 2). Springer. https://doi.org/10.1007/978-3-642-86896-2

Jaswal, S. (2017). Editorial: What next—The cognition of sequences. *Frontiers in Psychology, 8*, Article 2160. https://doi.org/10.3389/fpsyg.2017.02160

Johnson, K. (2003). Towards an etiology of adjunct islands. *Nordlyd, 31*(1), 187–215. https://doi.org/10.7557/12.25

Kayne, R. (1994). *The antisymmetry of syntax*. MIT Press.

Ke, A. H. (2023). Can Agree and Labeling be reduced to Minimal Search? *Linguistic Inquiry*. Advance online publication. https://doi.org/10.1162/ling_a_00481

Leisenring, A. C. (1969). *Mathematical logic and Hilbert's ϵ symbol*. Macdonald Technical & Scientific.

Leiß, H. (2017). On equality of contexts and completeness of the indexed ϵ-calculus. *The IfCoLog Journal of Logics and Their Applications, 4*(2), 347–366. http://www.collegepublications.co.uk/downloads/ifcolog00011.pdf

Lewis, D. (1979). Scorekeeping in a language game. In R. Bäuerle, U. Egli, & A. von Stechow (Eds.), *Semantics from different points of view* (pp. 172–187). Springer. https://doi.org/10.1007/978-3-642-67458-7_12

Manzini, R. (2021). Chomsky's (2020) Links and linker phenomena. *Qulso, 7*, 89–102. https://doi.org/10.13128/QULSO-2421-7220-12004

Mints, G., & Sarenac, D. (2003). Completeness of indexed ϵ-calculus. *Archive for Mathematical Logic, 42*(7), 617–625. https://doi.org/10.1007/s00153-003-0170-6

Patejuk, A., & Przepiórkowski, A. (2023). Category mismatches in coordination vindicated. *Linguistic Inquiry, 54*(2), 326–349. https://doi.org/10.1162/ling_a_00438

Retoré, C. (2014). Typed Hilbert epsilon operators and the semantics of determiner phrases. In G. Morrill, R. Muskens, R. Osswald, & F. Richter (Eds.), *Proceedings for the 19th international conference on formal grammar* (pp. 15–33). Springer. https://doi.org/10.1007/978-3-662-44121-3_2

Slater, B. H. (n.d.). *Epsilon calculi*. The Internet Encyclopedia of Philosophy. https://iep.utm.edu/ep-calc/

Slater, B. H. (2017). $(\exists y)(y = \epsilon xFx)$. *The IfCoLog Journal of Logics and Their Applications, 4*(2), 276–286. http://www.collegepublications.co.uk/downloads/ifcolog00011.pdf

von Heusinger, K. (1997a). Definite descriptions and choice functions. In S. Akama (Ed.), *Logic, language, and computation* (pp. 61–91). Springer. https://doi.org/10.1007/978-94-011-5638-7_4

von Heusinger, K. (1997b). *Salienz und Referenz: Der Epsilonoperator in der Semantik der Nominalphrase und anaphorischer Pronomen.* Akademie Verlag. https://doi.org/10.1515/9783050073934

von Heusinger, K. (2000). The reference of indefinites. In K. von Heusinger & U. Egli (Eds.), *Reference and anaphoric relations* (pp. 247–265). Springer. https://doi.org/10.1007/978-94-011-3947-2_13

von Heusinger, K. (2002). Specificity and definiteness in sentence and discourse structure. *Journal of Semantics, 19*(3), 245–274. https://doi.org/10.1093/jos/19.3.245

von Heusinger, K. (2004). Choice functions and the anaphoric semantics of definite NPs. *Research on Language and Computation, 2*(3), 309–329. https://doi.org/10.1007/s11168-004-0904-6

von Heusinger, K. (2013). The salience theory of definiteness. In A. Capone, F. Lo Piparo, & M. Carapezza (Eds.), *Perspectives on linguistic pragmatics* (pp. 349–374). Springer. https://doi.org/10.1007/978-3-319-01014-4_14

Whitehead, A. N., & Russell, B. (1910). *Principia mathematica* (Vol. 1). Cambridge University Press.

Zwart, J.-W. (2007, March 9). *Layered derivations* [Advanced core training in linguistics lecture, London]. https://www.let.rug.nl/zwart/docs/ho07actl.pdf

Zwart, J.-W. (2009). Prospects for top-down derivation. *Catalan Journal of Linguistics, 8*, 161–187. https://raco.cat/index.php/CatalanJournal/article/view/168909/221178

Zwart, J.-W. (2011). Recursion in language: A layered-derivation approach. *Biolinguistics, 5*(1-2), 43–56. https://doi.org/10.5964/bioling.8829