Re-examining the tradeoff between lexicon size and average morphosyntactic complexity in recursive numeral systems

David M. Yang (davidmyang@berkeley.edu)
Terry Regier (terry.regier@berkeley.edu)
Department of Linguistics, UC Berkeley
Berkeley, CA 94720 USA

Abstract

Denić and Szymanik (2024, henceforth D&S) argued that recursive numeral systems optimize the tradeoff between lexicon size and average morphosyntactic complexity. In support of this claim, they showed that a broad range of attested numeral systems trade off these two quantities nearly as well as the best of a large set of artificial numeral systems. However, D&S' artificial systems were in some respects not entirely comparable to natural ones. Here, we address this issue by creating a grammar framework that can represent both natural and artificial numeral systems, and we derive both natural and artificial numeral systems from this single framework, which ensures the comparability of the two sorts of systems. We test D&S' original claim under these new conditions, and find support for it. We also explore the proposal that numeral systems might optimize the sum of lexicon size and average morphosyntactic complexity under a fixed weighting of the two terms, and the role of the prior distribution over numbers.

Keywords: numerals; numbers; semantic systems; grammar; lexicon; complexity; efficient communication; morphosyntax

Introduction

It has been argued that pressure for efficient communication shapes languages generally (e.g. Zipf, 1949; Piantadosi et al., 2011; Fedzechkina et al., 2012; Gibson et al., 2019; Hahn et al., 2020; Haspelmath, 2021), and semantic systems in particular (e.g. Kemp et al., 2018; see also Rosch, 1999). In this context, efficiency is often cast as an optimal tradeoff between simplicity (having a compact cognitive representation) and informativeness (supporting precise communication). Each of these two characteristics is desirable in its own right, but they compete with each other: a maximally simple system might have only a single word to cover an entire semantic domain, which would not be informative. Conversely, a maximally informative system might have a separate word for each idea in a domain, which would not be simple. The proposal is that naturally occurring semantic systems reflect optimal tradeoffs between these two competing characteristics. Previous work has shown that this notion of efficiency explains cross-language semantic variation in the domains of color (Zaslavsky et al., 2018), kinship (Kemp & Regier, 2012), indefinite pronouns (Denić et al., 2022), personal pronouns (Zaslavsky et al., 2021), and Boolean connectives (Uegaki, 2024), among others.

Xu et al. (2020) argued that numeral systems across languages (e.g. Greenberg, 1990; Comrie, 2013) also reflect pressure for efficiency in this sense. However Denić and Szymanik (2024, henceforth D&S) highlighted a limitation of that argument. D&S noted correctly that while non-recursive numeral systems (e.g. approximate and restricted systems: Gordon, 2004; Pica et al., 2004; Hammarström, 2010; Comrie, 2013) may optimally trade off simplicity and informativeness, recursive numeral systems sometimes do not. The reason is that recursive numeral systems, such as the numeral systems of English and many other languages, are all equally informative over a large range because they all assign a unique numeral to each number within that range — but they vary in simplicity, sometimes substantially. It follows that all but the simplest recursive systems could in principle be simpler without sacrificing informativeness, and thus they do not optimally trade off these two quantities.

Given this, D&S proposed that in the case of recursive numeral systems, the relevant tradeoff is instead that between lexicon size and average morphosyntactic complexity. They tested their proposal by calculating these two quantities in a variety of attested recursive numeral systems, and in a large set of artificial systems. However, their artificial systems were in some respects not entirely comparable to natural ones. Here, we re-examine D&S' argument by addressing this issue, along with some other issues suggested by their article. In what follows, we first briefly recap D&S' argument, and then describe our own studies that probe D&S' central question in new ways. Ultimately, we find support for D&S' original claim, when re-assessed as we describe.

The argument of Denić and Szymanik (2024)

D&S focused on the number range 1-99, and they defined the *lexicon size* of a numeral system to be "the number of lexicalized meanings" (p. 6) that system uses in denoting numbers within that range — that is, the number of meanings for which a monomorphemic form is used in that system. For example, English lexicalizes the numbers 1-12 (*one, two, ... ten, eleven, twelve*) within this range, and thus the lexicon size of English is 12.[1] D&S defined the *morphosyntactic complexity* of a given numeral in a given system to be the total number of digits and arithmetic symbols in that numeral's underlying morphosyntactic form. Some examples from English are shown in Table 1. Here, the morphosyntactic complexity of

[1] D&S actually take *twelve* to be multimorphemic, in contrast with *eleven* which they take to be monomorphemic. We consider it simpler to consider *eleven* and *twelve* to each be monomorphemic. As D&S point out, their qualitative results are robust to this choice.

In D. Barner, N.R. Bramley, A. Ruggeri and C.M. Walker (Eds.), *Proceedings of the 47th Annual Conference of the Cognitive Science Society*

Number	Numeral	Morphosyntax	Complexity
4	*four*	4	1
14	*fourteen*	$10+4$	3
47	*forty-seven*	$4 \cdot 10+7$	5

Table 1: Morphosyntactic form and morphosyntactic complexity of three sample numerals in English, in D&S' scheme.

the numeral *four* is 1, because it contains only the single morpheme *four*, representing the number 4. In contrast, the complexity of *fourteen* is 3 because it is analyzed morphosyntactically as $10+4$, and one unit of complexity is allocated for each of 10 and 4 and also one for the operation of addition, denoted $+$ in the morphosyntactic form.[2] Finally, the morphosyntactic complexity of *forty-seven* is 5 because one unit of complexity is allocated for each lexified number $(4, 10, 7)$ and one for each operation (multiplication: $\cdot$, addition: $+$).[3] Given this, D&S calculate the average morphosyntactic complexity $amsc(L)$ of a language L as:

$$amsc(L) = \sum_{n \in [1,99]} P(n) \cdot msc(n, L) \quad (1)$$

Here, $msc(n, L)$ is the morphosyntactic complexity of the numeral expressing number n in language L, as just defined, and

$$P(n) \propto n^{-2} \quad (2)$$

is a prior distribution over numbers, motivated by earlier work (Dehaene & Mehler, 1992; Piantadosi, 2016; Xu et al., 2020).

D&S wished to compare a set of numeral systems in natural languages to a set of artificial languages that had been evolved to optimize the tradeoff between lexicon size and average morphosyntactic complexity, to see whether the natural languages optimize this tradeoff nearly as well as possible. They obtained lexicon size and morphosyntactic complexity for numerals in individual natural languages from descriptive grammars and similar resources. They then generated artificial languages using the numeral grammar of Hurford (2007, 2011), shown in Equation 3:

$$\begin{aligned} \text{NUMBER} \rightarrow & D \,|\, \text{PHRASE} \,|\, \text{PHRASE} + \text{NUMBER} \,| \\ & \text{PHRASE} - \text{NUMBER} \\ \text{PHRASE} \rightarrow & M \,|\, \text{NUMBER} \cdot M \end{aligned} \quad (3)$$

Specifically, artificial languages were generated by assigning several numbers to the Digits (D) and Multipliers (M) categories of this grammar. Multipliers are lexified numerals that serve as bases — for example, English has $M = \{10\}$ — and digits are all other lexified (monomorphemic) numerals. Numerals for all other numbers in the range 1-99 were then generated using the rules of this grammar, with the shortest construction selected for each number; in cases of ties, the tie was broken by random selection among the shortest constructions. D&S then used an evolutionary algorithm to generate variants of these artificial languages that *optimized the tradeoff* between lexicon size ($|D| + |M|$) and average morphosyntactic complexity — that is, to find languages such that there exists no other possible language that is better (lower) on one of these two dimensions without being worse (higher) on the other. They took the resulting optimized artificial languages to define the *Pareto frontier* of these two quantities. Their central finding was that attested recursive numeral systems lay on or near this Pareto frontier, suggesting that these numeral systems near-optimize the tradeoff between lexicon size and average morphosyntactic complexity.

[2] This analysis makes three assumptions: (1) that *-teen* is an allomorph of *ten*, i.e. a different phonetic realization of the same morpheme, (2) that arithmetic operations such as $+$ can be represented by a covert morpheme, i.e. one with no phonetic realization, and (3) that elements can be reordered in moving from the underlying form to the surface form, such that $10+4$ yields *fourteen* rather than **teenfour*. Such reordering does not affect complexity.

[3] This analysis similarly assumes that *-ty* is an allomorph of *ten*.

Open issues

D&S' study broadens the discussion over efficient communication. At the same time, it also leaves several important points open, and we address three of them here: (1) the extent to which natural and artificial numeral systems are comparable; (2) why attested systems lie where they do along the Pareto frontier; and (3) the role of the prior distribution. We pursue these issues in the following sections.

Comparability of natural and artificial systems

D&S' artificial systems are similar to natural ones, but there are differences, as D&S point out. The Hurford grammar that they use (Equation 3) is quite general – which is appropriate as it is intended as a universal grammar that accommodates cross-language variation – but it lacks certain constraints that natural systems exhibit. For example, D&S' application of Hurford's grammar often resulted in multiple equally short constructions for a given numeral, with a single construction chosen randomly, which yielded "unorderliness" (D&S, p. 12) in the artificial languages, including the optimal ones. An example of this is the rapid switching of bases back and forth in runs of consecutive numerals (e.g. $11 = 10+1; 12 = 3 \cdot 4; 13 = 10+3$; and later $17 = 4 \cdot 4 + 1; 18 = (10-1) \cdot 2; 19 = 10+10-1$; D&S, p. 13), something not seen in natural numeral systems. Another difference is that their set of artificial languages necessarily excludes aspects of certain natural ones, including English. For instance, English *eleven* is not an ordinary digit: one cannot express 21 as $10+11$ (**eleventeen*) and must instead use the more morphosyntactically complex form $2 \cdot 10+1$ (*twenty-one*). If we were to consider 11 to be a digit (rather than a multiplier, which it clearly is not) and followed D&S' scheme, then we would be forced to choose, as our representation for 21, the incorrect shorter form $10+11$ over the correct longer form $2 \cdot 10+1$. Thus D&S' use of the Hurford grammar, if extended to natural languages, would not correctly accommodate such suppletive forms, which occur elsewhere as well: e.g. Russian *sorok* is an irregular monomorphemic numeral for 40. These numerals affect average morphosyntactic complexity but cannot be appropriately represented in the artificial systems that D&S use. Thus, their natural and artificial systems, although similar, are not entirely comparable, which complicates the comparison between them.

To address this issue, we propose a new framework of grammar constraints placed on top of Hurford's grammar, an idea discussed by D&S as a possible direction for future work. Specifically, we (1) created a new category of numeral *S*, for suppletives, such as *eleven* in English, and added to Hurford's grammar a rule stating that NUMBER $\rightarrow S$, and (2) created four grammar constraints to be specified on a per-language basis. These four constraints are listed below:

- **Base constraint**: Defines the base (multiplier) used to construct numerals within a specific number range.
- **Number addition constraint**: Specifies the maximum number that can be added to a phrase within a specific range (e.g. 29 = 20 + 9).
- **Number subtraction constraint**: Specifies the maximum number that can be subtracted from a phrase within a specific range (e.g. 29 = 30 - 1).
- **Exceptions constraint**: Defines a specific construction (a non-canonical form other than suppletives) for a numeral within a given range.

Examples of the lexicon and of grammar constraints for English and Gola (iso: gol, Westermann, 1921) are shown in Table 2. English has a base-10 numeral system with digits for 1–9 and suppletive numerals for 11 and 12. Since it has one base, the base constraint specifies that for numerals in the range 10–99, the base to use is 10. The number addition constraint specifies that for numerals in the range 10-99, the maximum number that can be added to a phrase is 9. English has no subtraction, so that constraint is empty. The English exceptions constraint is empty because English has no exceptions other than the already-specified lexical suppletive forms for 11 and 12. Gola has a base-20 numeral system with smaller bases of 5 and 10. The base constraint specifies that base 5 is used to construct numerals between 5 and 9, base 10 is used to construct numerals between 10 and 19, and base 20 is used to construct numerals between 20 and 99. The number addition constraint specifies that for numerals between 5 and 9, the maximum number that can be added to a phrase is 4; for numerals between 10 and 19, the maximum number that can be added to a phrase is 9, and so on. Like English, Gola does not use subtraction so that constraint is empty. The exceptions constraint shows that 20, when it appears in the range $\{20\}$, is constructed as $1 \cdot 20$, even though 20 is lexicalized. We selected 40 natural languages of the 128 that D&S analyzed: the first language listed in each subtype of numeral system listed in their online appendix of language types (`https://osf.io/gw2x3/`). We specified language-specific constraints for each of these 40 languages, and verified that Hurford's grammar plus our language-specific constraints produced the correct numeral system for each.

We then constructed artificial numeral systems using the same grammar-plus-constraints framework just described. Artificial languages were generated by randomly selecting (1) a maximum value (between 1 and 20) for digits, and then defining digits sequentially from 1 up to and including that maximum number,[4] (2) bases (up to 5 numbers), and (3) suppletives (up to 5 numbers), each drawn from a uniform distribution. The smallest base was selected in a manner that ensured that all numerals from 1-99 could be constructed[5] — all other bases were randomly selected. Suppletives were randomly selected from numerals in the range 1-99 that were not included in the digits or bases sets. The base constraint was generated from the bases set by assigning each base in the set a range that extended from the base itself to the next base in the set (or to 99 if it was the last base). Next, it was randomly determined whether the language includes subtraction, with $P(\text{yes}) = 0.3$. If subtraction was included, the digits used for subtraction and addition were specified; otherwise, the number addition constraint was based on the base constraint, and the number subtraction constraint remained empty. Similarly, it was randomly determined whether the language includes exceptions, again with $P(\text{yes}) = 0.3$. If exceptions were included, they were generated using the language's base(s); otherwise, the exceptions constraint remained empty. The 0.3 probability was deliberately chosen to be higher than the rate of occurrence of subtraction and exceptions in natural languages in the dataset. These grammar constraints enforce a single construction per numeral; using these constraints together with Hurford's grammar, we generated constructions for the numerals 1–99, and took the result to be an artificial numeral system. Because of the constraints, these artificial systems exhibited systematicity of a sort that D&S' artificial systems did not — e.g. here, there was no rapid switching back and forth of bases in runs of consecutive numerals.

We then created optimal artificial numeral systems using an evolutionary algorithm similar to D&S', adjusted to also allow mutations (additions, deletions, or modifications) of suppletive forms in a manner analogous to D&S' mutation of bases, and to allow addition or deletion of rules from the exceptions constraint. An exception rule could be added if the number of exceptions was fewer than the number of bases, while a deletion could only occur if there was at least one exception rule. The first generation of the evolutionary algorithm generated 300 languages, and each subsequent generation added 50 new languages while mutating the optimal languages from the previous generation. This process was repeated for 100 generations. Finally, the last generation was combined with natural languages, and the Pareto frontier was identified by selecting Pareto-optimal languages from this

[4]Sequential values were chosen for the digits because in all 128 natural languages in D&S' dataset, digits followed a sequential pattern starting from 1.

[5]For artificial languages that used only addition, the smallest base was set as the number immediately following the largest digit. For those that used subtraction, the smallest base was restricted to a range that guaranteed all numerals 1-99 could be constructed: e.g. if the largest digit was 5, the largest permissible value for the smallest base was 11, to allow construction of the next numeral 6 = 11 - 5.

Lexicon/Grammar	English	Gola
Digits	$\{1,2,...,9\}$	$\{1,2,3,4\}$
Bases	$\{10\}$	$\{5,10,20\}$
Suppletives	$\{11,12\}$	$\{\}$
Base constraint	$[[\{10,\ldots,99\},10]]$	$[[\{5,\ldots,9\},5]$, $[\{10,\ldots,19\},10]$, $[\{20,\ldots,99\},20]]$
Number addition constr.	$[[\{10,\ldots,99\},9]]$	$[[\{5,\ldots,9\},4]$, $[\{10,\ldots,19\},9]$, $[\{20,\ldots,99\},19]]$
Number subtraction constr.	[]	[]
Exceptions constraint	[]	$[[20,\{20\},\text{‘}(1\cdot 20)\text{’}]]$

Table 2: Lexicon and grammar constraints for English and Gola. If a language does not need a particular constraint, it is represented as []. Following D&S , we restrict attention to the numerals 1-99.

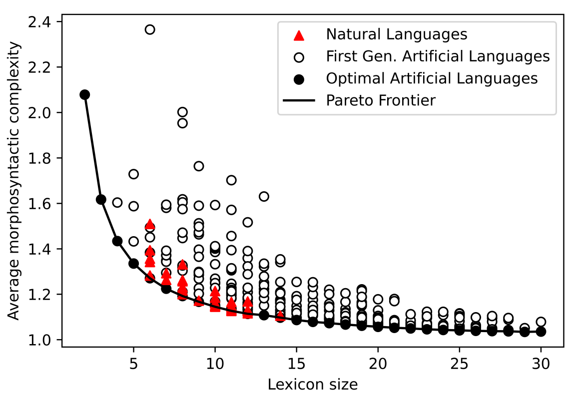


Figure 1: Tradeoff between lexicon size and average morphosyntactic complexity of natural languages and artificial languages with grammar constraints. Natural languages lie close to the Pareto frontier.

combined set.

The results of this analysis are plotted in Figure 1. It can be seen that the 40 natural languages we consider (red triangles) lie on or close to the Pareto frontier (black line), when compared to artificial languages with grammar constraints. Thus, natural languages continue to near-optimally balance lexicon size and average morphosyntactic complexity, even when artificial languages are structured similarly to natural ones.

Why do attested systems lie where they do?

D&S noted (p. 11) that in their analysis natural languages tended to lie not only along the Pareto frontier, but more specifically along a particular region of the frontier for which both lexicon size and average morphosyntactic complexity had relatively low values. This is also the case in our analysis, as can be seen in Figure 1. Why is this? D&S suggested that one possible reason for this could be that natural languages are actually minimizing the *sum* of lexicon size and average morphosyntactic complexity, with some fixed weighting factor for the two components of the sum. We wished to test this idea. To that end, we defined:

$$S(L) = amsc(L) + \lambda \cdot ls(L) \tag{4}$$

$S(L)$ is a weighted sum of the average morphosyntactic complexity $amsc(L)$ and the lexicon size $ls(L)$ for a given language L, with both quantities defined as above. λ is a free parameter controlling the relative weights of the two quantities. Minimizing this weighted sum for different values of λ yields the entire Pareto frontier (cf. e.g. Zaslavsky et al., 2018; Yu & Xu, 2025).[6] We have already seen that attested numeral systems lie near the Pareto frontier, and therefore we know that they near-minimize this sum for different values of λ. The remaining question is whether there exists a single value of λ at which all naturally occurring numeral systems near-minimize this sum.

To test this, we optimized λ through grid search. For each value of λ, we computed the sum $S(L)$ in Equation 4, for (a) all natural languages in our dataset, (b) all first-generation artificial languages, and (c) all optimized artificial languages that were produced as the output of the evolutionary algorithm. We then noted the language $L = L^*$ (whether natural or artificial) that yielded the lowest value for $S(L)$ at that value of λ. Finally, we selected the value of λ that minimized the difference between the optimal $S(L^*)$ and $S(L)$ for natural languages L; that value was $\lambda = 0.02199$. The distribution of $S(L)$ values at that value of λ is shown in Figure 2. Natural languages lie at the lower end of this distribution, which to some extent reflects our optimization of λ. Note however that most natural languages cluster more specifically near the minimal value for $S(L)$. This outcome is consistent with the proposal that naturally occurring numeral systems optimize the sum of lexicon size and average morphosyntactic complexity under a fixed tradeoff value between the two quantities. At the same time, this analysis leaves open the question of why that tradeoff parameter has the value that it does.

The role of the prior

The study by Xu et al. (2020), to which D&S responded, explored the tradeoff between informativeness and simplicity in

[6]This can be seen by noting that if we obtain the optimal L^* that minimizes the sum in Equation 4 for a given value of λ, we know that no other L exists that would yield a lower value for $amsc(L)$ at the same value of $ls(L) = ls(L^*)$, because if such a lower value for $amsc(L)$ did exist, it would yield an even lower value for the sum, at that value of λ. The same argument holds analogously for the other dimension. Adjusting λ sweeps up and down the frontier.

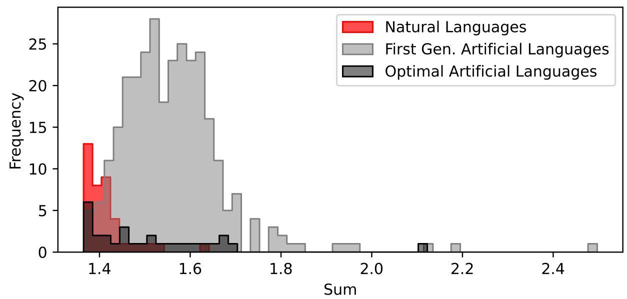


Figure 2: Attested numeral systems near-minimize the sum of lexicon size and average morphosyntactic complexity, with a fixed weighting factor.

numeral systems. Xu et al. (2020) found that (non-recursive) systems traded off these two quantities efficiently — that is, they found that attested systems lay on or near the Pareto frontier defined by those two quantities. They then manipulated the prior distribution over numbers (recall Equation 2 above), and found that their efficiency results were sensitive to the prior. Their main analysis used a prior based on the frequency with which numbers are named in language use (e.g. Dehaene & Mehler, 1992), which is left-skewed, with mass decreasing as number increases, as in the power law in Equation 2. Under this prior, they found that attested numeral systems exhibited efficiency. In contrast, when they used an unrealistic uniform prior, they found that attested numeral systems were farther from the Pareto frontier. They concluded from this that "the efficiency of attested systems relies on the tendency for smaller numeric values to be used more often" (p. 66). To our knowledge, no such manipulation of the prior distribution has been attempted for the tradeoff between lexicon size and average morphosyntactic complexity in recursive numeral systems, and so we conducted such a manipulation.

We considered three priors: (a) the power law of Equation 2, which is very similar to the original prior used by Xu et al. (2020), (b) a uniform distribution, shown in Equation 5, and (c) a reverse power law which assigns higher mass to higher numbers, shown in Equation 6.

$$\text{Uniform prior:} \quad P(n) = \frac{1}{99} \tag{5}$$

$$\text{Reverse power law:} \quad P(n) \propto (100-n)^{-2} \tag{6}$$

Figure 3 shows the tradeoff between lexicon size and average morphosyntactic complexity under these two alternate priors, for comparison with the same tradeoff obtained under the original power-law prior, shown above in Figure 1. Considering first the uniform prior, in Figure 3(a), we can see that natural languages appear to lie reasonably close to the Pareto frontier, but apparently farther than under the original power-law prior (cf. Figure 1). To test this impression, we defined the *deviation from optimality* for a given language L under a given prior to be $amsc(L)$ minus the minimum $amsc(\cdot)$ score obtained for that lexicon size by any natural or artificial language under that prior: this measures how far above the Pareto frontier language L lies.[7] Figure 4 shows deviation from optimality for all 40 natural languages we considered, under the power-law, uniform, and reverse power-law priors. Note that Mandarin, shown in red, has no deviation from optimality under the power-law prior; thus it is an optimal numeral system under that prior.[8] We can also see that the Mandarin system is far from optimal under the two alternate priors. More generally it appears that across languages, deviation is lowest under the power-law prior. A sign test was conducted to compare the deviation from optimality under the power-law prior vs. the uniform prior, for all natural languages we considered. All 40 languages showed lower deviation from optimality under the power-law prior than under the uniform prior (40/40 positive signs, $p \ll 0.0001$). Turning next to the reverse power-law prior, in Figure 3(b), we can observe that in this case natural languages lie quite far from the Pareto frontier. Another sign test was conducted, this time to compare the deviation from optimality under the power-law prior vs. the reverse power-law prior. All 40 languages showed lower deviation from optimality under the power-law prior than under the reverse power-law prior (40/40 positive signs, $p \ll 0.0001$). Bonferroni corrections do not alter the qualitative outcome.

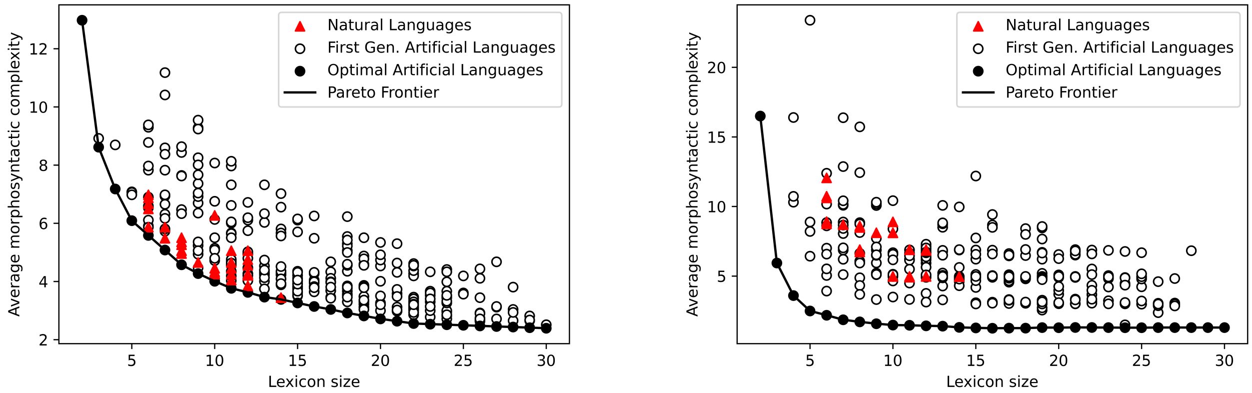
Why do we obtain this outcome? It is known that the bases for attested numeral systems tend to be relatively small numbers, such as 10 or 20 (e.g. Comrie, 2013). This allows simpler morphosyntactic forms for lower numbers generally (e.g. $15 = 10+5$) compared with higher numbers (e.g. $85 = 8 \cdot 10 + 5$). This pattern fits the power-law prior well because under this prior, lower numbers receive more weight — and average morphosyntactic complexity will in general be reduced by having shorter forms for more probable numbers, as is the case here. The power-law prior is motivated by the frequencies with which humans name different numbers (Dehaene & Mehler, 1992). It appears then that attested numeral systems are well-suited for communication under the distribution of numeral usage frequencies actually found in human language, but are not as well-suited for communication under some alternative distributions. This is compatible with a possible adaptation of numeral systems to the frequencies with which humans need to name specific numbers.

Discussion

D&S argued that recursive numeral systems near-optimally trade off lexicon size and average morphosyntactic complexity. We pursued three issues left open by D&S' study, and showed: (1) that their central result continues to hold when natural and artificial numeral systems are made en-

[7] The prior affects only $amsc(\cdot)$ and not lexicon size, so we focus here on $amsc(\cdot)$ only.

[8] Mandarin is a base-10 system similar to English, with the exception that it lacks suppletives such as *eleven* and *twelve* for the numbers 11 and 12, and instead uses compositional forms that express the notions $10+1$ and $10+2$ respectively.



(a)

(b)

Figure 3: Tradeoff between lexicon size and average morphosyntactic complexity of natural languages and artificial languages with grammar constraints, under alternate priors. (a) Uniform prior. Natural languages lie fairly close to the Pareto frontier, but not as close as with the power-law prior. (b) Reverse power-law prior. Natural languages lie far from the Pareto frontier.

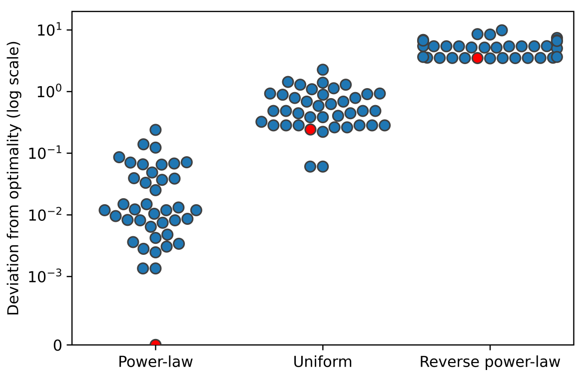


Figure 4: Deviation from optimality (log scale) for the 40 natural languages we consider, under each of the three priors. Overall, deviation is lowest under the independently motivated power-law prior. Mandarin is highlighted in red, and is an optimal numeral system under the power-law prior.

tirely comparable, in that they are generated by the same constrained grammatical framework; (2) that natural numeral systems near-minimize the sum of lexicon size and average morphosyntactic complexity for a particular fixed tradeoff value, which may help in understanding why languages appear where they do along the Pareto frontier; and (3) that natural numeral systems exhibit the near-optimal tradeoff noted above in (1) when assessed under a prior that reflects the frequencies with which humans name different numbers, but they emerge as less optimal under other, hypothetical, priors. Taken together, our results elaborate, extend, and ultimately support D&S' central argument.

Our results also leave a number of issues open, suggesting avenues for future work. The study by Xu et al. (2020), to which D&S responded, measured the size of the grammar and lexicon taken together, whereas D&S measured only lexicon size, assuming a universal numeral grammar of constant size. We have added language-specific grammar constraints to that universal grammar, which can result in grammars of different sizes for different languages — but we have here followed D&S in focusing only on lexicon size, rather than the size of the lexicon and grammar taken together. This choice seems natural in that intuitively, lexicon size and morphosyntactic complexity are opposed: e.g. a unary system has the smallest possible lexicon and very high morphosyntactic complexity, whereas a system with separate unrelated lexical items for each number in the range 1-99 has exactly the opposite configuration for these two quantities. Still, measuring the complexity of our language-specific grammar constraints, and exploring the role of that complexity in the tradeoff with average morphosyntactic complexity, seem relevant to obtaining a more complete picture; we leave this question for future work. Another issue left open is a fuller explanation of why languages lie where they do along the Pareto frontier. We have explored a partial explanation, in terms of the sum of lexicon size and average morphosyntactic complexity under a fixed weighting factor (cf. Xu et al., 2020, Supplementary Materials, p. 10). But our approach leaves the specific weighting factor unexplained, and there is a natural alternative possible explanation. Numeral systems may exhibit the range of lexicon sizes they do because of reliance on external cognitive reference points such as the number of body parts like fingers and/or toes, and the restriction of lexicon size to values easily grounded in such external reference points. Again, we leave exploration of these competing proposals, or some synthesis of them, for future work. We hope that our findings will help to motivate further research exploring numeral systems, and other semantic systems, in light of pressure for communicative efficiency.

Acknowledgments

We thank Milica Denić and Jakub Szymanik for their paper to which ours responds, and for making their code available. We also thank Bernard Comrie for his helpful input, and 3 anonymous reviewers for helpful comments on an earlier version of this paper. Any errors are our own. Author contributions: DY and TR designed the research; DY performed the research; DY analyzed the data; and DY and TR wrote the paper. Code and data can be found at `https://github.com/davidmyang/numerals-public`

References

Comrie, B. (2013). Numeral bases (v2020.4). In M. S. Dryer & M. Haspelmath (Eds.), *The World Atlas of Language Structures online.*

Dehaene, S., & Mehler, J. (1992). Cross-linguistic regularities in the frequency of number words. *Cognition*, *43*(1), 1–29.

Denić, M., & Szymanik, J. (2024). Recursive numeral systems optimize the trade-off between lexicon size and average morphosyntactic complexity. *Cognitive Science*, *48*(3), e13424.

Denić, M., Steinert-Threlkeld, S., & Szymanik, J. (2022). Indefinite pronouns optimize the simplicity/informativeness trade-off. *Cognitive Science*, *46*(5), e13142.

Fedzechkina, M., Jaeger, T. F., & Newport, E. L. (2012). Language learners restructure their input to facilitate efficient communication. *Proceedings of the National Academy of Sciences*, *109*(44), 17897-17902.

Gibson, E., Futrell, R., Piantadosi, S. P., Dautriche, I., Mahowald, K., Bergen, L., & Levy, R. (2019). How efficiency shapes human language. *Trends in Cognitive Sciences*, *23*(5), 389–407.

Gordon, P. (2004). Numerical cognition without words: Evidence from Amazonia. *Science*, *306*(5695), 496–499.

Greenberg, J. H. (1990). Generalizations about numeral systems. In K. Denning & S. Kemmer (Eds.), *On language: Selected writings of Joseph H. Greenberg* (pp. 271–309). Redwood City, CA: Stanford University Press.

Hahn, M., Jurafsky, D., & Futrell, R. (2020). Universals of word order reflect optimization of grammars for efficient communication. *Proceedings of the National Academy of Sciences*, *117*, 2347-2353.

Hammarström, H. (2010). Rarities in numeral systems. In J. Wohlgemuth & M. Cysouw (Eds.), *Rethinking universals: How rarities affect linguistic theory* (pp. 11–60). De Gruyter Mouton.

Haspelmath, M. (2021). Explaining grammatical coding asymmetries: Form–frequency correspondences and predictability. *Journal of Linguistics*, *57*(3), 605–633.

Hurford, J. R. (2007). A performed practice explains a linguistic universal: Counting gives the packing strategy. *Lingua*, *117*(5), 773–783.

Hurford, J. R. (2011). *The linguistic theory of numerals* (Vol. 16). Cambridge University Press.

Kemp, C., & Regier, T. (2012). Kinship categories across languages reflect general communicative principles. *Science*, *336*(6084), 1049–1054.

Kemp, C., Xu, Y., & Regier, T. (2018). Semantic typology and efficient communication. *Annual Review of Linguistics*, *4*, 109–128.

Piantadosi, S. T. (2016). A rational analysis of the approximate number system. *Psychonomic Bulletin and Review*, *23*(3), 877–886.

Piantadosi, S. T., Tily, H., & Gibson, E. (2011). Word lengths are optimized for efficient communication. *Proceedings of the National Academy of Sciences*, *108*(9), 3526–3529.

Pica, P., Lemer, C., Izard, V., & Dehaene, S. (2004). Exact and approximate arithmetic in an Amazonian indigene group. *Science*, *306*(5695), 499–503.

Rosch, E. (1999). Principles of categorization. In E. Margolis & S. Laurence (Eds.), *Concepts: Core readings* (pp. 189–206). Hillsdale, NJ: Erlbaum.

Uegaki, W. (2024). The informativeness/complexity trade-off in the domain of Boolean connectives. *Linguistic Inquiry*, *55*(1), 174–196.

Westermann, D. (1921). *Die Gola-Sprache in Liberia: Grammatik, Texte und Wörterbuch* (Vol. 4). Hamburg: L. Friederichsen Co.

Xu, Y., Liu, E., & Regier, T. (2020). Numeral systems across languages support efficient communication: From approximate numerosity to recursion. *Open Mind*, *4*, 57–70.

Yu, L., & Xu, Y. (2025). Infinite mixture chaining: An efficiency-based framework for the dynamic construction of word meaning. *Open Mind*, *9*, 1–24.

Zaslavsky, N., Kemp, C., Regier, T., & Tishby, N. (2018). Efficient compression in color naming and its evolution. *Proceedings of the National Academy of Sciences*, *115*(31), 7937–7942.

Zaslavsky, N., Maldonado, M., & Culbertson, J. (2021). Let's talk (efficiently) about us: Person systems achieve near-optimal compression. In *Proceedings of the 43rd Annual Meeting of the Cognitive Science Society.*

Zipf, G. K. (1949). *Human behavior and the principle of least effort*. Cambridge, MA: Addison-Wesley.