# Simplicity of Underlying Representation as Motivation for Underspecification

David Odden

A fundamental assumption underlying research in generative phonology has been that grammatical descriptions should contain the minimum number of symbols. This view is enforced through three principles. First, it is held that the number of rules in a grammar should be minimized. Second, each rule should be formalized with as few feature specifications as possible. Third, underlying representations should contain the minimum number of feature specifications. This paper reports the results of computational experiments in underspecification which raise questions about the legitimacy of this last assumption. It is shown that the goal of reducing the number of underlying feature specifications leads to intractable problems in computing the optimal grammar for a language. For any phonemic inventory there are vast numbers of underspecified representations, and the principle of representational simplicity does not provide an effective method for selecting between competing analyses. This leads to postulating a principle restricting the use of underspecification to cases where there is direct evidence in a language for deleting certain feature values from underlying representations: the mere possibility of doing so does not per se justify underspecification. Furthermore, the fact that underspecification of feature values will result in a grammar using fewer symbols will not be taken to be sufficient motivation for underspecification.

The outline of the paper is as follows. Section 1 establishes the significance of simplicity as a motivating force in the literature of underspecification. Section 2 discusses certain technical issues in underspecification theory, focusing in particular on the consequences of the fact that there are many ways to underspecify a phonemic inventory. Section 3 presents results from computational experiments in generating underspecification systems from surface segmental inventories.

## 1. The role of simplicity arguments in underspecification

Arguments for underspecifying feature values which one encounters in the literature can be divided into four classes.

(1) Line crossing
Behavioral asymmetry
Linking and markedness
Simplicity

The first two types of arguments may motivate eliminating certain feature values from underlying representations; the last two do not, and the focus of this paper is on the very last of these arguments.

The first argument, the line crossing reason, is clearly the strongest. An exemplar of this argument is the underspecification analysis of neutral vowels in vowel harmony. The vowels [i] and [e] in Finnish are transparent to the spreading of the feature [back], despite being [-back] on the surface. Given the fundamental principle that association lines cannot cross, it is inescapable that these vowels must be underspecified for the feature [back].[1]

---

[1]A terminological distinction will be adopted here between underspecification and nonspecification of features. The term "nonspecification" will be employed when a segment lacks a given feature at every stage in a derivation. The term "underspecification" will refer

Asymmetry arguments, such as those given in Pulleyblank (1988) regarding the vowel [i] in Yoruba (where only [i] triggers certain rules, and all vowels but [i] seem to trigger other rules) or those in Pulleyblank (1986) for mid tone in Yoruba could also provide evidence for feature underspecification, especially if it is impossible to directly exclude the segment in question from the class of triggers or targets. Although Clements and Sonaiye (1989) argue that the Yoruba vowel argument is not as compelling as appears, the questions which they raise cast doubt on a specific analysis, and do not show that asymmetry is in principle an unacceptable argument.

The linking and markedness argument, also known as the epenthetic vowel argument, is often given to support the claim that certain features are unspecified in underlying representations. For example, if a language has a number of vowel epenthesis rules, all of which insert the same vowel, it is widely assumed that one only needs to insert a timing slot and allow redundancy rules to fill in the relevant features. However, Mohanan (1991) shows that a theory with structure preserving constraints (which may include something analogous to default rules) can account for such phenomena; underspecification in underlying representations is not necessary to explain why rules of vowel epenthesis generally insert a single vowel in a given languge. While studying the results of epenthesis rules may provide some information about the form of redundancy rules in a language, the existence of such rules does not per se show that a given segment must have an underlying blank in positions where the feature specification could be supplied by redundancy rule.

The weakest reason for underspecifying segments, the one to be investigated here, is that eliminating redundant features results in simpler underlying representations. This paper argues that considerations of simplicity have no validity. Yet the simplicity argument plays a central role in underspecification literature. For example, Halle (1959:29-30) sets forth as one of the fundamental axioms of phonological analysis the following condition:

> Condition (5): In phonological representations the number of specified features is consistently reduced to a minimum...

Later, commenting on two competing methods for specifying the Russian phonemes /t,s,c,n/, it is noted (p. 36)

> The freedom in ordering feature-questions may result in several branching diagrams compatible with the above requirements. In such cases the choice may be dictated by Condition (5)... It is evident that the second ordering is the more economical since it yields a greater number of zeros.

Halle (1962: 60) states "In general, we must omit features in all dictionary representations, whenever these can be introduced by a rule that is less costly than the savings it effects". In a similar vein, Stanley (1967: 434) states:

---

more narrowly to the situation where a segment is specified for a feature in the output of the phonology, but is not so specified in some earlier stage. Feature geometry makes this distinction important. Rules of front ~ back vowel harmony are generally blind to intervening consonants, so for example labial consonants never intrinsically trigger or block backness harmony. In pre-geometric autosegmental theories, this would entail that [p] (which would be [–back]) is underspecified for the feature [back], since it neither causes fronting of vowels nor interferes with [back] spreading from other vowels. Adopting certain theories of feature geometry such as those proposed in Sagey (1986) or Steriade (1987), [back] is a feature under the dorsal (or velar) node, and labials have no dorsal node either in underlying or surface representations. In such a case, the lack of a [back] specification for labial consonants is not underspecification, as defined here.

> ...we have an evaluation measure which tells us what the best set of MS conditions is; it is, essentially, the shortest set of MS conditions that allows us to leave the greatest number of blanks in dictionary matrices...[2]

In the contemporary literature on underspecification, formal simplicity continues to play a major role. According to Archangeli (1984:36),

> In this theory, certain values for all features are supplied by redundancy rules, rather than being present in underlying representation. In this way, underlying representations are simplified.

Similarly, it is stated that (p. 41-2)

> The underlying representation of our hypothetical language is considerably streamlined if underspecification is assumed. This means that the language learner has less to learn and less to memorize.

Elsewhere, it is claimed that elimination of whole features or elimination of feature values, takes priority over considerations of rule simplicity (p. 48):

> There are various assumptions possible about what constitutes the most highly valued minimally specified matrix. These break into three categories, algorithms based on:
> (2.25)
> a. the rules necessary to supply the missing feature values.
> b. the number of feature values (i.e. the number of pluses and minuses) in underlying representation.
> c. the number of features in underlying representation.
> Consideration of these options ranks the third above the other two, and the second above the first.

The Feature Minimization Principle is established, to elevate these sentiments to the level of theoretical principle (p. 50):

> FEATURE MINIMIZATION PRINCIPLE:
> A grammar is most highly valued when underlying representations include the minimal number of features necessary to make different the phonemes of the language.

Finally, it is stated (p. 65) that

> The procedure includes principles like those noted above: Minimize the number of features and minimize the number of marks.

Archangeli (1988: 183) echos these sentiments.

> An evaluation metric in Universal Grammar provides a means of selecting between possible grammars for a particular language. The evaluation metric as conceived in Chomsky and Halle (1968; henceforth SPE) prefers the

[2] For the most part, Stanley (1967) presumes that "dictionary" entries contains zeros for all redundant features, but appears to reject all forms of underspecification on p. 435.

> grammar in which only the idiosyncratic properties are lexically listed and predictable properties are derived. The essence of underspecification theory is to supply such predictable distinctive features or feature specifications by rule.

More recently, Archangeli and Pulleyblank (1992) have reaffirmed the methodological principle of selecting the simplest underlying representation, setting forth the principle of Representational Simplicity which holds that the value of a representation is the inverse of the number of terminal F-elements (features) and associations to features.

Thus the question of simplicity, in terms of counting feature specifications and in terms of counting features having specifications, has been a significant motivating factor for underspecification. Virtually no attention has been paid to addressing the question of why it should be intrinsically desirable to require that the underlying representation be "simple" in this sense. One view, expressed in the earlier generative literature, was that on the assumption that storage space in the brain is limited, underlying representations containing fewer elements could be stored in less space than would be required for fully specified representations. This kind of reasoning cannot be given much credence: our understanding of the mechanisms of information storage in the human brain is not sufficiently refined that we can seriously compare the consequences of full specification versus underspecification for a psychologically realistic model of phonology. A related argument is that under-specification gives the child "less to learn" about a language. In fact, underspecification gives a child more to learn — the child must learn what the rules are in the language which fill in missing feature values. Moreover, given that the surface feature values are phonetic features, it is not necessary for the child to "learn" that, for example, [a] is [+voice]. That fact can be provided by simple observation. What does require learning is that in some language, certain [+voice] segments are not specified as [+voice] in underlying representations. Such an inference requires reasoning beyond simple observation of the phonetic segments in a language.

## 2. Technical questions

There are two widely accepted theories of underspecification, often referred to as Contrastive Underspecification (CU) and Radical Underspecification (RU), which differ on whether contrastive feature specifications may be eliminated from underlying representations. The problem considered here regarding simplicity is largely independent of the CU/RU debate. We will start from the simpler set of assumptions made within CU, discover the consequences of pursuing simplicity considerations, and then extend the investigation to the RU framework.

### 2.1 Eliminating redundant features in Contrastive Underspecification

In order to investigate the interaction between underspecification and simplicity, it is necessary to state explicitly how an underspecified representation is arrived at. With prior knowledge of the underspecified representation and the rules which fill in redundant values, one can mechanically apply the rules to the representation and verify that correct surface segments result. Inferring a set of rules and an underlying representation from a surface inventory, as a child learning a language might do, is much more difficult. The first step is therefore to consider techniques for removing all and only non-contrastive feature values from a surface inventory and arriving at an underlying inventory.[3]

---

[3]The term "contrastive" is defined as follows (this assumes the standard definition of feature value distinctness which is that only the values + and – are distinct):

It is important to recognize that feature values must be arranged into a hierarchy in order to resolve conflicting redundancy relations. A consideration of the 3 vowel system [iau] shows why this is so. There is a well-known redundancy relation between the features [back], [low] and [round] in such vowels systems. Since all [+back,–low] vowels are [+round], all [+low] vowels are [–round], and all [–back] vowels are [–round] it is possible to predict all values of [round] on the basis of values of [back] and [low], and no segment needs to be specified for a value of [round]. But it is also possible to predict all values of [back] on the basis of [round] and [low]: all [+low] vowels are [+back], all [–round,-low] vowels are [–back] and all [+round] vowels are [+back].

It is not possible to simultaneously eliminate specifications of both [round] and [back], since doing so renders the contrasting vowel pairs {i,u} indistinct.[4] There are at least two ways to underspecify the system [iau], one in which [back] is eliminated and [round] is retained, and one in which [round] is eliminated and [back] is retained. Each such hierarchy of features represents a potentially distinct underspecified underlying representation. Stanley (1967: 400) recognizes this problem, and in commenting on the method of feature underspecification in Halle (1959) observes that

> there may be considerable freedom in the way this branching diagram is constructed for a given set of systematic phonemes; a different choice of redundant feature values in this set will lead to a different branching diagram and thus to a different hierarchy of features.

A simple algorithm can be devised which deletes features from a surface inventory of a language and arrives at an underspecified representation as follows:

---

> DEFINITION: The value of feature $F_i$ is contrastive in segment X iff there exists a segment Y such that the value for $F_i$ of X is not identical to the value for $F_i$ of Y, and there is no other feature $F_j$ where the value for $F_j$ of X is distinct from the value for $F_j$ of Y.

[4]Archangeli (1988: 192) presents the following algorithm for establishing a contrastively underspecified representation:

a. fully specify all segments
b. isolate all pair of segments
c. determine which segment pairs differ by a single feature specification
d. designate such feature specifications as 'contrastive' on the members of that pair
e. once all pairs have been examined and appropriate feature specifications have been marked 'contrastive', delete all unmarked feature specifications on each segment.

From this she concludes that only a single underlying representation is possible for a given system. It is then argued (p. 201-2) that CU cannot distinguish the vowels in the system [iaɔæʊ], on the grounds that following the above algorithm, [i] and [æ] have the same representation, namely [–back], since the features [low] and [high] will be unspecified for both vowels on the grounds that they are not (surface) contrastive.

The problem which Archangeli points to is not strictly a consequence of the theory of CU. Rather, it is the result of a specific algorithm for underspecifying an inventory in CU theory. As will be seen here — and as hinted by Archangeli — deletion of redundant features cannot be performed simultaneously on all features, but rather must be performed with respect to an assumed preference among features for deletability.

(2) Select some feature value of some segment. If that value is noncontrastive, delete it.
Repeat until all feature values of all segments have been evaluated.

This procedure avoids the paradox of incorrectly eliminating values of both [round] and [back]. One may start with the value of [round] for [i]. That value is noncontrastive ([i] differs from [u] and [a] in [back]), so the value of [round] for [i] is deleted. Proceding to [round] in [u], the value is noncontrastive ([u] differs from [i] in [back], and from [a] in [high]), so [round] is deleted. Finally coming to [round] in [a], that value is also noncontrastive ([a] differs from [i] and [u] in [high]), so it too is deleted. This gives the following partial result.

(3)

| | high | low | back | round |
|---|---|---|---|---|
| i | + | – | – | 0 |
| u | + | – | + | 0 |
| a | – | + | + | 0 |

Now, coming to the value of [back] for [i], [i] and [u] contrast only in the feature [back], so [back] cannot be eliminated from [i] (nor can it be eliminated from [u]).

On the other hand, one might instead start with the value of [back] for [i], which is noncontrastive, and eliminate it. By a similar procedure one arrives at a system where all values of [back] are eliminated, predicted primarily on the basis of the value of [round]. Other hierarchies are possible: one could start with the value of [round] for [i], proceed to the value of [back] for [u], and so on, and arrive at a different system. The number of distinct hierarchies to consider is K! (i.e. $1 \times 2 \times 3 ... \times k$ ) where $k$ is the number of feature values in a given inventory (which is the number of segments multiplied by the number of features). To specify the three vowel system [iua] using the four relevant vowel features, there are 12 feature values, so there are 479,001,600 (=12!) different ways in which to approach the underspecification task. For the 5 vowel system [ieaou] which requires 20 feature specifications, there are 2,432,902,008,176,640,000 (=20!) arrangments of the four relevant features, and for the phonemic inventory of English which uses 528 feature specifications (based on 33 segments and 16 surface features to represent those segments) there are $1.5 \times 10^{2146}$ (=528!) feature arrangements.[5]

Given the requirement that redundant features must be removed from underlying representations, there are many competing hypotheses which must be considered regarding which system of underspecification is selected over all of its competitors. Simplicity plays a crucial role in grammar selection (cf. especially Halle (1959), Stanley (1967)): it acts as the final arbitrar in picking an underlying representation for a phonemic inventory, since one will select the system which gives an underlying representation with the fewest feature values remaining specified.

[5]There are obviously many more paths for getting at underspecified systems under this algorithm than there are underspecified systems. If any feature value were freely deleted independent of any other value and with no regard for whether the resulting system is a well formed underspecification system, then given a total of $k$ feature values there are at most $2^k$ underlying representations. So for a 5 vowel system there could be no more than 1,048,576 distinct underlying representations.

## 2.2 Redundant features in Radical Underspecification

The difference between RU and CU is the additional principle of RU that no feature may be specified with both minus and plus values. RU requires context sensitive rules filling in values of features on the basis of values for other features (as does CU), and unlike CU it also requires context free rules which insert plus or minus values of each feature without reference to other feature specifications. It is clearly the practice in RU to include context sensitive rules of segment redundancy as well as context free rules. For instance, in de Haas's (1988:238) analysis of the vowel system [ieaou] in Kasem, [+low] is eliminated from /a/ by a rule filling in [+low] on all [+back,–round] vowels, and yet there is also a context free rule assigning the value [-low] to all vowels. Similarly, in Hyman (1988: 261) [+low] vowels in Esimbi are specified [–hi], but there is also a context free rule assigning [+high] to all remaining vowels. In Ringen (1988: 332) the context free value of [back] in Hungarian is [+back], but [– rd,–low] vowels are by a context sensitive rule given the value [–back], allowing [i,e] to be unspecified for backness. Finally, Vago (1988: 354) assigns [–back] vowels in Pasiego the value [–low], and [+back,–low] vowels receive the value [+round]; in addition, [-round] and [+back] are assigned by context free rules.

The addition of context free redundancy rules gives RU a greater degree of freedom in establishing an underspecified inventory: the possibilities for underspecification in RU are a product of the context sensitive rules plus the context free rules. Systems of radical underspecification can be generated on the basis of systems of CU underspecification in the following manner: beginning with any CU system, create two new systems, one where all plus values of the first feature are deleted, and one where all minus values of the first feature are deleted. Repeat this procedure on the resulting systems with the second feature and so on until no feature has both plus and minus values. Given any CU system of specification with *k* features having both plus and minus values, there are $2^k$ corresponding RU systems.

## 3. Computing underspecification systems

This section describes the results of a series of computational experiments in underspecification, whose goal is determining how the simplest underlying representation can be found. One algorithm for underspecification, (2) above, has been considered, where feature values may be considered in any order. This procedure was implemented as a computer program which reads in a phonemic system and eliminates feature specifications from that phonemic system according to the algorithm, given an order in which features are eliminated. With this procedure one could in principle search all possibilities looking for the simplest underlying representation. However since there are 479,001,600 ways to delete features in the vowel system [iau], an exhaustive search is not feasible.

### 3.1 Improving the algorithm

One way to reduce the size of the task is to adopt a more restrictive algorithm for underspecification. The search procedure was therefore constrained so that all values of a particular feature are underspecified simultaneously. This assumes a linear ordering of features, such that all possible eliminations of values of $F_1$ will take place, followed by elimination of values of $F_2$, and so on. Schematically, the procedure operates as follows:

(4) Start with the first feature $F_1$ and the first segment A. Locate the next segment B which has a distinct value of $F_1$.
Scan the features of segments A and B: if there is no other feature such that A and B have distinct values of that feature, then the values which A and B have for feature $F_1$ is contrastive.
If the values which A and B have for $F_1$ is not contrastive, continue the comparison by determining whether A and the next segment C have contrastive values for $F_1$ (and so on until A has been compared with all segments in the language).
If there is no segment which has a contrastive value for feature $F_1$, then the value which A has for $F_1$ is noncontrastive. When all segments have been scanned for $F_1$, noncontrastive values of $F_1$ are deleted. Proceed to the next feature.

This algorithm still requires an ordering among features, but the number of orderings is reduced to K!, where *k* is the number of features present. Consequently the number of cases to search does not grow as a function of the number of segments in the language. Under this approach, there are 24 feature orders to consider in an exhaustive search of all underspecification systems for a vowel system using the features [high], [back], [low] and [round]. For 16 features, there are 20,922,789,888,000 orders. In comparison to the $1.5\times10^{2146}$ possibilities for the vowel system [ieaou] under the "any feature in any segment" algorithm, this is a considerable reduction in the size of the problem. It must be kept in mind, though, that the restricted algorithm is incapable of discovering underspecification systems which the unrestricted algorithm can uncover.[6] While this gives fewer orders to consider, it remains impossible on practical grounds to evaluate the set of underlying representations arising from each of these orders for anything but the simplest tasks. An exhaustive search of an entire phonemic inventory, or of a consonantal system, is still out of the question.

An alternative is to select a large random sample of orders of feature evaluation, and generate the underspecification systems resulting from those orders. Given a sufficiently large sample, one can reasonably extrapolate from the properties of the sample to the properties of the whole set. This task can be performed by a computer program which eliminates feature specifications following the revised algorithm.

A search was conducted for representations of the vowel system [ieaou] using the features [high], [low], [back] and [round], employing 900,000 randomly selected orders. This search resulted in 50 distinct ways of underspecifying this vowel system. These same 50 systems can also be reached by a less extensive search of 500 orders. It is therefore quite likely that an exhaustive search would not yield many more ways of underspecifying the vowel system [ieaou].

### 3.2 Underspecifying English

The language selected for this part of the study was a dialect of English.[7] This inventory was specified with standard SPE-style surface feature assignments, using 16

---

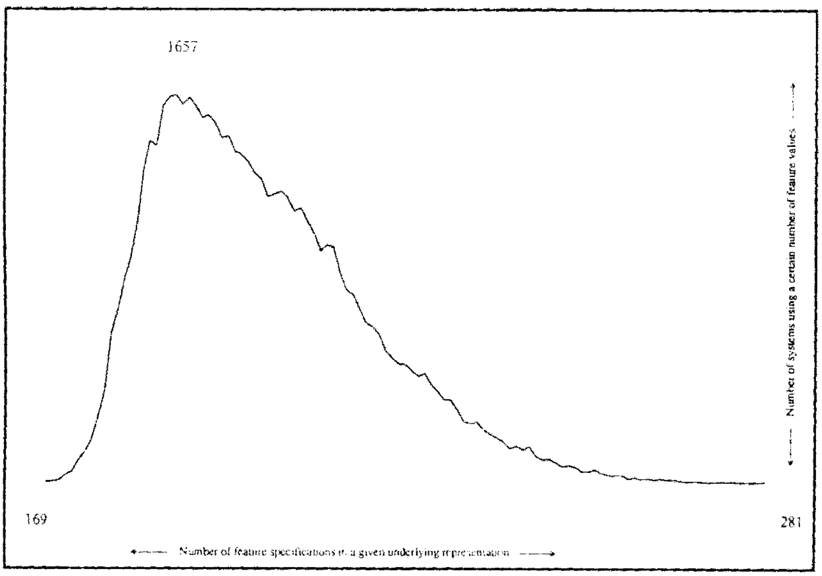
[6]Since this algorithm allows 24 ways of ordering the features [high], [low], [back] and [round] there could never be more than 24 ways of underspecifying a vowel system using just those features. In fact, there are 12 ways of underspecifying the system [ieaou] under this algorithm. The previously discussed algorithm allows at least 50 ways of specifying this system: the 12 allowed by the current algorithm, plus 38 others.

[7]In this dialect — that of the author — there is no contrast between [a] and [ɔ]. Furthermore, as a simplifying assumption, syllabic sonorants are phonemicized as

features. Certain features such as [constricted glottis] and [spread] were not entered into the database, given that all segments of English have the same value for those features. A sample of underlying systems based on 60,000 distinct random orders was generated; obviously this is a miniscule fraction of the entire population of underspecification possibilities. This sampling resulted in 59,958 distinct underlying representations. In short, only 0.07% of the systems in the sample turned out to be identical to other systems in the sample.

Different underlying systems may have different numbers of underlying feature specifications. (5) provides a plot of the feature elimination function for this sample. The vertical axis indicates the number of underlying systems which exhibit a particular degree of feature elimination, ranging from a low of 1 system to a high of 1657 systems. The horizontal axis indicates the total number of surface plus and minus values which remain in a given underspecified representation, ranging from a low of 169 to a high of 281. Expressed as a percentage of feature values remaining specified, there is a range from 32% to 53%; the mean number of specified features is 201, or 38%. However, the peak of the graph actually lies at 188 features, or 35%.

(5) ELIMINATION OF FEATURES IN ENGLISH — SPE FEATURE REPRESENTATIONS




As inspection of the graph reveals, this is a result of a relatively larger number of less efficient systems, in contrast to the relatively smaller number of more efficient systems. The mean will therefore be disregarded in favor of reporting where the peak of the function appears. 33% of the cases in the sample have between 34% and 36% of their features specified, and 46% of the sample specifies between 33% and 37% of the features. Only 1.9% are in the efficient range of less than 34% specification.

In this sample, there are two systems which result in the minimum number of feature specifications. Given the assumption that the least costly underlying representation is to be selected, one of these two systems would be the underlying system which a child acquiring English strives to uncover. But in order to arrive at one of these two solutions in the first place, a huge number of hypotheses must be considered and rejected — there is a 0.003% chance of finding one of these two underspecification systems. Moreover, if the drive to find a maximally underspecified inventory is so strong, there is no reason to end the search at this point, since it is possible that continuing the search will result in a system which has an even greater degree of underspecification, and is therefore even more desirable.

It is also interesting to consider two competing metatheories of what constitutes a "simple" underlying representation. On the one hand, there is the classical notion that the system requiring the fewest underlying feature values is simplest. On the other hand, Archangeli (1984) holds as most highly valued those systems which "include the minimal number of features" — that is, systems maximizing the number of features having no specifications at all. In this sample, we find that the degree of total underspecification of features is as follows:

(6)

| features eliminated | number of underspecification systems |
|---|---|
| 5 | 6 |
| 4 | 457 |
| 3 | 3235 |
| 2 | 12816 |
| 1 | 25335 |
| 0 | 18109 |

By the criterion that total elimination of specifications for a feature is highly desirable, there are 6 "most highly valued" underlying systems.[8] Again, a child attempting to learn English would have to sift through a huge number of hypotheses to arrive at one of the few systems which totally underspecifies the maximum number of features.

The two notions of simplicity do not lead to selecting the same underlying systems. The two systems which employ the fewest specifications eliminate in one case four features and in the other case two. The six systems maximally eliminating features require between 192 and 217 specifications — the mean for these systems is 201. In short, what is an optimal system by one criterion is merely an average system by the other criterion.

To diminish the possibility that these results are the consequence of a too-small sample, a second sample of underspecification systems, based on 100,000 random feature orders, was taken. In this sample, there were 99,917 distinct underlying representations. The properties of this sample are in essence the same as those of the sample of 60,000 orders. The range of feature specifications remains between 169 and 281 features: the peak of this function lies at 187, or 35%. Again, the highly efficient systems specifying fewer than 34% of features accounts for only 1.8% of the sample. In this sample, there are 4 systems employing the minimum number of specifications which remains at 169 features.

In terms of the number of features totally eliminated, we find the following.

---

[8]A feature may not be eliminated if there exists any segment pairs which contrast only in specification for that feature. For English, six features can be eliminated, namely [round], [consonantal], [lateral], [nasal], [sonorant] and [delayed release]. The features which are eliminated in the 6 systems using the fewest features are these features except for [sonorant].

(7)

| features eliminated | number of underspecification systems |
|---|---|
| 5 | 919 |
| 4 | 6975 |
| 3 | 22080 |
| 2 | 36626 |
| 1 | 29493 |
| 0 | 3824 |

Again, in each of the systems allowing maximal feature elimination, the features eliminated are [round], [consonantal], [lateral], [nasal], and [delayed release]. In that set of 919 systems the range of feature specifications is from 170 to 233, with the mean at 188. Note that this coincides with the peak of the feature elimination function for the entire sample, indicating that the systems maximally eliminating features are neither significantly less efficient nor more efficient at reducing the total number of feature values.

An additional sample of 300,000 orders resulted in 299,264 distinct underlying systems: the range remains from 169 to 281 features specified, with the peak still at 187 features. 1.9% of this sample use fewer than 34% of the features. In this sample, there are 11 systems specifying the minimum number of specifications — again, there is a .003% chance of encountering one of these maximally efficient systems. There are 2,557 systems in this sample which use the minimum number of features. The average number of specifications within this subset of the sample is 188, which is the average for the entire sample. Two systems in this sample employ the minimum number of features and 169 feature specifications, which is the minimum in the whole sample. These two systems maximally satisfy both simplicity criteria, so if formal simplicity is to be taken seriously, one of these two systems would be the underlying representation selected by a child. However, these systems can be identified only after a very extensive search.

One final search of possible underspecification systems was undertaken, this time using 2,000,000 orders.[9] This resulted in 1,973,479 distinct ways to underspecify the English phonemic inventory, and again the vast majority of these underlying systems use 188 feature specifications. There were 66 systems using the minimum number of feature specifications, which remains at 169. There are also 14,799 systems which require the minimum number of features, which is 11, and in that group, there are 5 systems which also require the minimum number of feature specifications. Presumably criteria other than simplicity would be required to select the best underlying system out of this group of 5.

The conclusion to be drawn from these four exercises in underspecification are the following. First, there are vast numbers of underspecification systems possible for English — if the trend continues, it is not unreasonable to expect that there are nearly 16! distinct ways of underspecifying the phonemic inventory of English. Second, the overwhelming majority of underspecification systems for English require about 188 features, give or take about a dozen features. The probability of encountering a highly efficient system using the fewest features and specifications is very small — one chance in 150,000. A vast number of underspecification systems must be searched in order to arrive at one which is minimalist. Finally, the number of feature values saved by seeking a highly parsimonious underlying systems as opposed to taking one of the less parsimonious but more frequent systems in the middle, is around 17 feature values. Very little is saved in reward for seeking a simpler system.

---

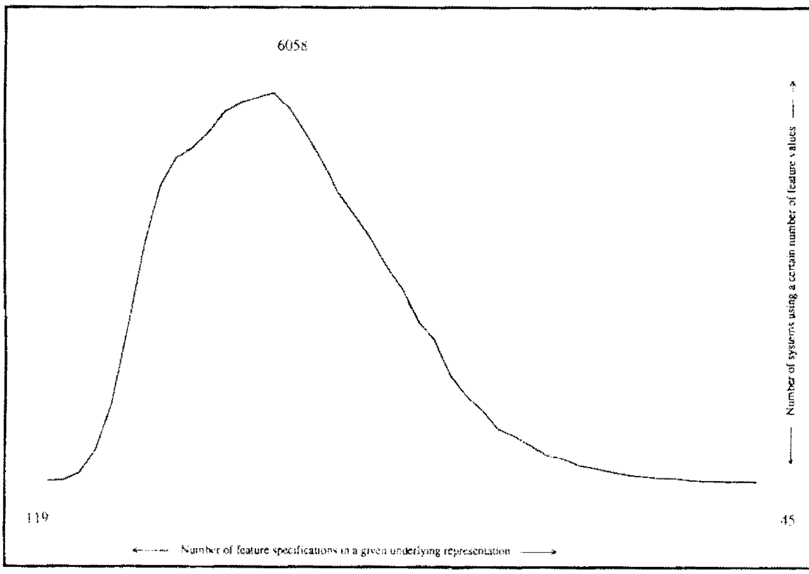
[9]This search was conducted on a Cray YMP-64: I would like to thank the Ohio Supercomputer Center for a grant of computer time which enabled this portion of the study to be conducted. Over 65 hours of CPU time were required to complete this search: it would require approximately 11,000 years to complete an exhaustive search.

### 3.3 Underspecification and Feature Geometry

It might reasonably — but incorrectly — be suspected that the proliferation in underspecification systems is an artifact of the SPE model of features for place of articulation, where vowels are redundantly specified [–coronal] and consonants are redundantly [–back]. Therefore, a sample of English segmental underspecification systems was taken, based on a feature geometric approach where vowels (but not consonants) are specified with the features [hi], [low], [back], [round] and [ATR]; and consonants (but not vowels) are specified with the privative features [coronal], [labial] and [dorsal]. In such a system, there is no underspecification of coronality for vowels — rather, vowels are non-specified for [coronal]. This will eliminate artificially inflated degrees of underspecification arising from the SPE model of place features.

However, it turns out that the same problem arises even adopting a geometric view of features. A sample of 99,989 distinct orderings of features was taken, and from this sample, 99,450 resulting systems proved to be unique, again showing that there are vast numbers of competing hypotheses to consider in arriving at an optimal underlying representation, and the feature-elimination function is graphed in (8).

(8) ELIMINATION OF FEATURES IN ENGLISH — GEOMETRIC REPRESENTATIONS




In this sample, the number of features remaining specified ranged from a low of 119 to a high of 162, out of 374 features specified on the surface. While this is lower than what was encountered in underspecifying English based on SPE-style feature analysis, this is entirely due to the fact that surface representations have fewer features specified, in comparison to the SPE approach. Expressed as a percentage of the total number of specifications, between 32% and 43% of surface features remain specified underlyingly (compare this to the range 32% to 53% for the SPE-based analysis). The mean count of specifications for this sample was 133, or 35% (compare this to 38% with SPE features): the peak of the graph and the mean coincide in this case. In the 34%–36% specification range we find 50% of cases; in the 33%–37% range find 79% of the systems. Again, 1.8% of the sample lies at the most efficient value of 32% specification. In comparison to a SPE-style feature analysis, there is a slight improvement in the mean efficiency of underspecification with a geometric analysis of features. For the most part, there is little change from the earlier finding, that there are many schemes which result in similar degrees of underspecification, and only a few highly efficient systems.

The degree of total underspecification of features is as follows:

(9)

| features eliminated | number of underspecification systems |
|---|---|
| 3 | 2768 |
| 2 | 16485 |
| 1 | 43968 |
| 0 | 36229 |

In other words, 2.7% of the sample eliminates the maximum number of features. One out of every 36 systems sampled eliminates the greatest number of features, in contrast to one out of every 9993 systems using an SPE-style specification of features.

3.4 Separating vowels and consonants

The size of the search can be further reduced by restricting possible redundancy relations which could result in deletion of features. This can be done by splitting the task of underspecifying the inventory of a language into two independent subtasks, namely underspecifying consonants and underspecifying vowels. As a further simplifying assumption, the glides [w] and [y] are not included in either subsystem: this has the benefit of not introducing otherwise unnecessary vocalic features into the consonantal system, and also eliminates the need to consider the property of syllabicity in the search of the vocalic subsystem. Two additional searches of underspecification systems were undertaken. The first was an exhaustive search of the 120 possible systems[10] for representing vowels, and the second was a search of 300,000 possible underspecification systems for consonants (where consonants are given a articulator-based analysis).

This search of vowel systems resulted in 54 distinct ways to specify the vowel system. The amount of underspecification is very restricted: underlying systems for vowels require between 34 and 38 values to represent English vowels.[11]

(10)

| N values | number of systems with N features |
|---|---|
| 34 | 17 |
| 35 | 8 |
| 36 | 15 |
| 37 | 12 |
| 38 | 2 |

[10]This represents 5!, which is the number of feature orders possible with the five features required to specify English vowels, namely [high], [low], [back], [round],and [tense].

[11]Again it must be remembered that this algorithm only identifies a subset of the underspecification systems which are found by the "any feature of any segment" algorithm. A sample of 90,000 feature orders using that method of searching resulted in 4,291 distinct ways to underspecify the English vowel system. Expanding the search to 900,000 orders uncovered 4,723 systems.

The search of consonants resulted in 81,639 unique underlying systems out of 300,000 searched. The range of feature specifications lies between 70 to 102. Given 193 surface feature specifications, this gives a range of 37% to 53%, with the mean at 42%. There were 28 highly efficient systems uncovered which employ only 70 features. Increasing the size of the search to 600,000 cases resulted in 105,361 distinct ways of underspecifying English consonants. These systems require between 70 and 103 feature specifications; in this set there are 34 systems employing the minimum number of feature values. This means that there are 578 distinct underlying systems of representation which have the same degree of minimal specification of features — any one of the 34 most efficient consonantal systems combined with any of the 17 most efficient vocalic systems.

3.5 Simplicity in Radical Underspecification

Underspecifying an inventory within the tenets of RU requires an additional layer of rules, namely context free rules supplying contrastive feature specifications. A system lacking only contrastive feature values can be mapped onto a set of RU-style representations by deleting the plus or the minus value for any feature. If there are *k* features containing both plus and minus values under context sensitive underspecification, there are $2^k$ RU-style representations. Thus, the 12 context sensitive ways of underspecifying the vowel system [ieaou] can be transformed into 96 RU representations. Those systems would not all necessarily be distinct: in fact, 76 of those representations turn out to be distinct.[12]

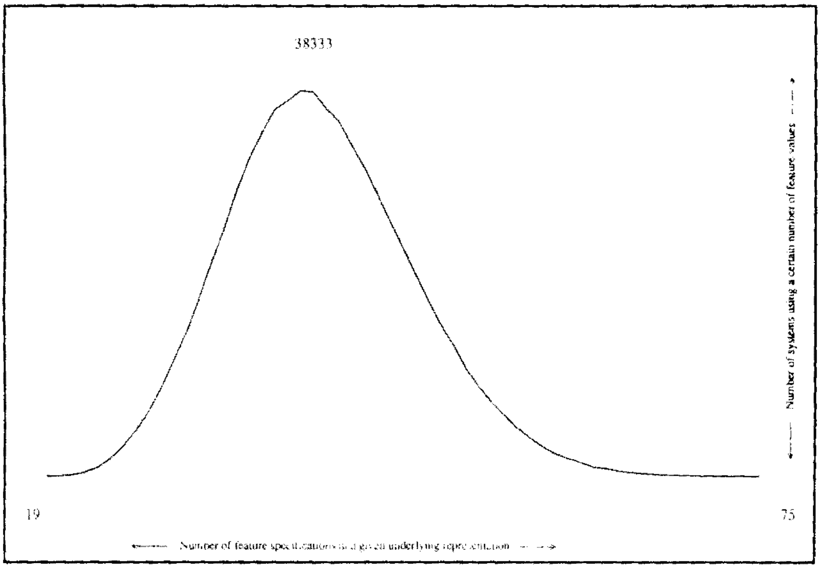
The first substantial problem considered here is what the simplest RU specification is for the English phonemic inventory (adopting a geometric representation of consonant and vowel place features). Previous sections have noted that an exhaustive search of context sensitive underspecification systems for English is impractical since there are huge numbers of possibilities to consider. It is even less practical to perform an exhaustive search of possible RU systems, since there are approximately 32,000 times more RU representations.[13] So while there are around $1.2\times 10^{16}$ possibilities to consider using only context sensitive rules, there are about $3.8\times10^{20}$ possibilities with context free rules added.

One can approximate the properties of the entire set of RU systems by taking a large randomly selected set of systems which are subjected to context sensitive rules of underspecification, and subject them to a large random selection of patterns of context free underspecification. A computer program therefore randomly selected 1,000 context sensitive underspecified systems and for each of them searched 300 of the roughly 32,000 possible RU transformations. This resulted in 293,533 unique RU systems. The feature-elimination function for this sample is shown below.

[12]Building on the less restrictive "any feature of any segment" algorithm which yields 50 underspecified systems, there is an upper limit of 672 RU-style representations for [ieaou], and of these 478 are distinct.

[13]On the average, 18 features are required to specify the segments of English. The three privative features [coronal], [labial] and [dorsal] are intrinsically "radically" underspecified. In effect, then, there are about 15 features subject to context free underspecification, hence the factor of 32,000.

(11) ELIMINATION OF FEATURES IN ENGLISH — RADICAL UNDERSPECIFICATION




It is obvious that this graph is essentially identical to previous graphs. The average number of features specified ranges from 51 to 120, with the mean at 74; 4 systems employ the minimum number of feature specifications. In other words, there is less than 1 chance in 70,000 of discovering one of these highly efficient systems.

This task was further broken down into separate searches of consonants and vowels. An exhaustive search of the 1,728 ways to specify the English vowel inventory was conducted,[14] and resulted in 1,060 distinct underlying systems of respresentation. These systems required between 13 and 25 feature specifications, with the average being 18. There were 4 systems which employed the minimum number of feature specifications.

An exhaustive search of RU-style representations of English consonants is impractical, so an approximation of this set was made by randomly selecting 1,000 CU systems and subjecting each to 900 randomly selected patterns of context free underspecification. This sample yielded 689,275 distinct systems using between 19 and 75 feature specifications, with an average of 41. There was one system employing 19 features: the probability of finding that system is obviously quite small.

## 4. Conclusions

A metaphor often applied to the task of learning a language is that the child constructs competing hypotheses and applies the simplicity metric to each, keeping only the most parsimonious grammar. The results of this study provide a reason to reject such a view of language acquisition, insofar as it applies to underspecification of a phonemic inventory. In order for the child scientist to be reasonably convinced of having discovered

[14]This is based on 32 RU transformations of the 54 CU-style representations for English vowels.

the most parsimonious system, the child would need to spend decades simply enumerating the competing hypotheses, to say nothing of comparing the formal simplicity of the resulting systems. As an alternative view of underspecification, it is claimed here that the null hypothesis is that features specified on the surface are also specified underlyingly. In consequence, only direct evidence for underspecification — for example the transparency of certain vowels for harmony processes — can motivate entertaining the hypothesis that a given feature is underspecified.

## References

Archangeli, D. 1984. *Underspecification in Yawelmani phonology and morphology.* Doctoral dissertation, MIT: Cambridge. Published 1988 by Garland, New York.

Archangeli, D. 1988. Aspects of underspecification theory. *Phonology* 5:183–207.

Archangeli, D. and D. Pulleyblank. 1992. *Grounded Phonology.* MS, University of Arizona and University of British Columbia.

Clements, G.N. and R. Sonaiye. 1989. Underlying feature specification in Yoruba. MS, Cornell University and Obafemi Awolowo University.

de Haas, W. 1988. Phonological implications of skeleton and feature underspecificaiton in Kasem. *Phonology* 5: 237-254.

Halle, M. 1959. *The sound pattern of Russian.* Mouton: The Hague.

Halle, M. 1962. Phonology in generative grammar. *Word* 18:54–72.

Hyman, L. 1988. Underspecification and vowel height transfer in Esimbi. *Phonology* 5: 255–273

Mohanan, K.P. 1991. On the bases of underspecification. *Natural Language and Linguistic Theory* 9: 285-325.

Pulleyblank, D. 1986. *Tone in lexical phonology.* Dordrecht: Reidel.

Pulleyblank, D. 1988. Vocalic underspecification in Yoruba. *Linguistic Inquiry* 17: 233–270.

Ringen, C. 1988. Transparency in Hungarian vowel harmony. *Phonology* 5: 327–342.

Sagey, E. 1986. *The representation of features and relations in nonlinear phonology.* Doctoral dissertation, MIT: Cambridge.

Stanley, R. 1967. Redundancy rules in phonology. *Language* 43:393–436.

Steriade, D. 1987. Locality conditions and feature geometry. *NELS* 17: 595-617.

Vago, R. 1988. Underspecification in the height harmony system of Pasiego. *Phonology* 5, 343–362.