

# REPRESENTATION OF FEATURE SYSTEMS IN A NON-CONNECTIONIST MOLECULAR MACHINE

László Kálmán  
Institute of Linguistics  
Hungarian Academy of Sciences  
Budapest I., P.O.Box 19.  
H-1250 HUNGARY

## ABSTRACT

This paper is part of an enterprise whose aim is to represent linguistic knowledge in the form of a molecular machine (a dynamic network). That is, the molecules of the network not only store, but also send, receive, and process information. It is claimed that such a network can be conceived of as a model of the coalition structure of a connectionist network. The paper describes how the class of feature systems called unary feature hierarchies (whose importance is supported by phonological theory but will not be argued for in the paper) can be represented in the molecular machine.

## INTRODUCTION

Of the various branches of modern linguistic theory, phonology, as usual, was the first one to come up with really restrictive theories of features and feature systems, largely supported by empirical evidence coming from the study of a legion of linguistic phenomena. Phonology will most probably remain main source of evidence as far as the functioning of feature-based cognitive processes is concerned.

The present paper sets out to outline how a particular kind of feature system, close to several recent theories of phonological features, can be represented in connectionist networks. However, since some of the mechanisms involved, seriality and synchronization in particular, fall outside the scope of the existing connectionist networks, a non-connectionist model simulating would-be serial connectionist networks will be used instead. The automaton described in the paper is able to unify feature structures as programs to be run on the machine itself.

## UNARY FEATURE HIERARCHIES

The feature systems under scrutiny can be termed unary feature hierarchies. The underlying concept is close to the 'feature geometry' approach to autosegmental phonology (Clements 1985) in that a feature specification consists of features appearing on tiers, and features on adjacent tiers can be linked by association lines. Tiers are ordered in a multi-dimensional space, and the set of features that may appear on a particular tier is predefined. Unarity, on the other hand, means that a feature either appears within a given 'span' (of time) or it does not (instead of having different values); the absence of a feature simply means lack of information or underspecification. Features can only be linked to specified slots; on the other hand, tiers are adjacent to features rather than to tiers (hence the use of hierarchy instead of geometry); that is, the following configuration may be ruled out if the H tier is adjacent to F but not to G:

```
F/G tier  ----F----G----
           |      |
H tier     ----H----H----
```

Figure 1. This configuration may be ruled out.

Unarity is opposed to traditional binary feature systems (with a marked '+' and an unmarked '-' value for each feature) and to ternary systems (with a marked '+', an unmarked '-', and an unspecified '0' value), while feature geometries (and hierarchies) replace the old-fashioned 'feature bundle' conception, in which each segment consisted of an unordered set of feature-value pairs. Compare the following (sketchy) representations of lax vowels in English:

Features	Segments
	e æ i ʌ o ɔ u
Low	- + - - - + -
High	- - + - - - +
Back	- - - + + + +
Round	- - - - + + +

Figure 2. English lax vowels as feature bundles.

Tiers	Segments
	e æ i ʌ o ɔ u
Root tier	x x x x x x x
H/L/B tier	
	L H B B B B
R tier	
	R R R
H/L tier	
	L H

Figure 3. Unary feature hierarchy for English lax vowels.

That is, in this approach, feature hierarchies not only express universal restrictions on feature structures (as feature geometries do, cf. Clements 1985), but language specific redundancy rules and feature co-occurrence restrictions as well. In addition to the above, we shall assume the possibility of forbidding that a feature (or, rather, any feature of a given tier) be linked to another. Link prohibitions are not intended to be a feature value, hence the below representations

	a.	b.
Tier 1	----F----	----F----
Tier 2	-----	----- X

Figure 4. Configurations with and without link prohibition.

specify identical elements, in spite of the fact that no feature can be linked to F in Figure 4a. (X stands for link prohibition.) For example, consider the following feature hierarchy, characterizing a language where plural and dual nouns usually behave in the same way (e.g. Hebrew):

	1	2	3	4	5
Category tier	---N---	---N---	---N---	---N---	---N---
Number tier			X		
	-----	NSg	-----	NSg	NSg
Dual tier					X
	-----			D	-----

- 1: singular noun
- 2: plural noun
- 3: singulare tantum
- 4: dual noun and duale tantum
- 5: plurale tantum

Figure 5. Sample feature system.

In terms of the above, the features of plural nouns and pluralia tantum are identical, in spite of the link prohibition concerning the 'Non-Singular' feature of the latter.

### CONNECTIONIST MODEL OF FEATURE SYSTEMS

Obviously, the presence of a feature in a segment corresponds to a relatively high activation level of a node or a coalition of nodes in a connectionist network. The vertical geometry of tiers determines the possibilities of linking; the essential function of links is to synchronize the activation of features, but they also express their dependency relations. Thus association lines correspond to the fact that the activation of a node or coalition controls the activation of another node or coalition. Finally, link prohibitions can be represented as the inhibition of particular tiers. The adjacency of tiers will correspond to super-coalitions, i.e. features on adjacent tiers are features whose activation can be synchronized at all.

In our present knowledge, none of the existing connectionist networks can learn and encode the sequential activation of nodes or coalitions of nodes, albeit the organization of human memory is most probably highly serialized. To overcome this deficiency (and for other reasons as well) we designed a molecular machine (cf. Kálmán and Kornai 1985) to model the functioning of a sequential connectionist network.

In this machine, each molecule contains a more or less stable piece of knowledge encoding the coalition structure it participates in, in the form of a layout of the surrounding network.

Molecules are able to send messages to each other; a message has the same form as the pieces of information stored in the molecules, i.e. it is a directed graph representing a substructure of molecules with their dependencies. When a molecule receives a message, it unifies the message with its own map, and it forwards the resulting information. The ultimate addressees are the 'output molecules', which can perform extra actions when receiving a properly addressed message. That is, the message acts as a program that will, after some modifications it undergoes, activate a set of output molecules in a well-defined order. Modifications correspond to the effects of the rules stored in the molecules on the path of the message.

One of the most important characteristics of the messages sent (and other maps) is that they may contain missing steps, i.e. distinguished nodes encoding lack of information concerning the way between two nodes. Missing steps can be eliminated by unification, provided that the required path is specified in the map of a processing molecule. This mechanism corresponds to the function of 'super-coalitions', i.e. the synchronization and sequencing of the activation of otherwise distinct coalitions.

## INFORMATION PROCESSING WITH THE MOLECULAR MACHINE

The basic operation performed by the molecular machine is a kind of unification, differing from the corresponding operation used in unification-based grammars by virtue of the differences in the feature structures and their representations. Since the map graphs of the molecular machine contain no node variables, unification essentially means the elimination of missing steps. Missing steps encode both alternative and conjunctive relations:



Figure 6. Sample directed graph with missing step.

The missing step (the asterisk) in Figure 6 can only be eliminated by supplying a path from A and B to either C or D, e.g. by unifying the graph in Figure 6 with the following:



Figure 7. Sample directed graph without missing step.

Figure 7 is also the resulting graph. Any other unification will provide a partial solution at best.

The control function of the directed graphs in question is due to the fact that the addressed molecules are able to reduce the messages they receive. For example, in the case of the graph on Figure 7, both molecules A and B will remove their addresses from the top level before forwarding the graph. The graph on Figure 6, on the other hand, will be unable to activate any of the addressees because of the missing step it contains.

The unification and control functions of the molecular machine together define a particular conception of phonological rule systems, which can be argued for on independent grounds as well. Under this approach, the function of a phonological rule system is essentially to resolve problems arising from morphological processes, i.e. to fix ill-formed representations resulting from affixation by supplying missing features (e.g. in vowel harmony), linking or delinking features according to the derived context (e.g. in voice assimilation). Note that delinking in the present form of the machine consists of adding an inhibitory link leading to the feature to be delinked.

## FURTHER PLANS, IMPLEMENTATION

It is far from evident that syntactic and semantic rule systems should carry out operations similar to those in phonological rules. On the other hand, the operation of the molecular machine are general enough to eventually encompass syntactic and semantic processes such as recognition and completion of syntactic patterns, inference making through unification, etc. Some of these operations are outlined in Kálmán and Kornai (1985) and Kálmán (1986).

A small molecular machine is implemented in Zoltán Schreter's (FPSE, Geneva University) CNS connectionist simulation system running on Olivetti M24/M28 PCs. However, owing to the capacity of the machines (and of TLC Lisp, in which the system has been written) the number of molecules is extremely limited, and the performance obtained is rather poor.

## REFERENCES

Clements, George N. 1985. The geometry of phonological features. Phonology Yearbook 2, 225-252.

Kálmán, László and András Kornai. 1985. A finite-state approach to generation and parsing. Paper presented at the Generative Grammar Fiesta, Salzburg.

Kálmán, László. 1986. Semantic interpretation in a dynamic knowledge representation. Műhelymunkák (Working Papers of the Institute of Linguistics) 1, No. 2, pp. 31-51.