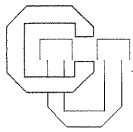
Harmonic Grammer – A formal multi-level connectionist theory of linguistic well-formedness: Theoretical foundations

Geraldine Legendre
Yoshiro Miyata
Paul Smolensky

CU-CS-465-90



University of Colorado at Boulder
DEPARTMENT OF COMPUTER SCIENCE

Harmonic Grammar - A formal multi-level connectionist theory of linguistic well-formedness: Theoretical foundations

Géraldine Legendre[1,2]
Yoshiro Miyata[1,3,4]
Paul Smolensky[1,3,4]

[1]*Institute of Cognitive Science*
[2]*Department of Linguistics*
[3]*Department of Computer Scence*
[4]*Optoelectronic Computing Systems Center*
University of Colorado at Boulder
Boulder, CO 80309-0430

ICS Technical Report #90–5 CU-CS-465-90

Acknowledgements

Warm thanks to Alan Prince, for very helpful discussions, and especially, a great pot of chicken soup and the term "isoharmonic." This work owes its existence to Mike Mozer, who failed to convince us not to do it. Thanks also to Jim Martin for his valuable comments on an earlier version. This work has been supported by NSF grants IRI-8609599 and ECE-8617947 to PS, by a grant to PS from the Sloan Foundation's computational neuroscience program. PS (in part) and YM have also been supported by the Optical Connectionist Machine Program of the Center for Optoelectronic Computing Systems, which is sponsored in part by NSF/ERC grant CDR-8622236 and by the Colorado Advanced Technology Institute, an agency of the State of Colorado. GL has been supported in part by a Junior Faculty Development Award from the Council on Research and Creative Work, University of Colorado, Boulder. The authors are listed in alphabetical order.

Abstract

In this paper, we derive the formalism of *harmonic grammar*, a connectionist-based theory of linguistic well-formedness. Harmonic grammar is a two-level theory, involving a low level connectionist network using a particular kind of distributed representation, and a second, higher level network that uses local representations and which approximately and incompletely describes the aggregate computational behavior of the lower level network. The central hypothesis is that the connectionist well-formedness measure *harmony*[1] can be used to model linguistic well-formedness; what is crucial about the relation between the lower and higher level networks is that there is a harmony-preserving mapping between them: they are *isoharmonic* (at least approximately). In a companion paper (Legendre, Miyata, & Smolensky, 1990; henceforth, "LMS_1"), we apply harmonic grammar to a syntactic problem, unaccusativity, and show that the resulting network is capable of a degree of coverage of difficult data that is unparallelled by symbolic approaches of which we are aware: of the 760 sentence types represented in our data, the network correctly predicts the acceptability in all but two cases. In the present paper, we describe the theoretical basis for the two level approach, illustrating the general theory through the derivation from first principles of the unaccusativity network of LMS_1.

Introduction

Our starting point is the approach to connectionist cognitive modeling called the *subsymbolic paradigm* (Smolensky, 1988):

(1) **Hypotheses of the subsymbolic approach to cognitive modeling**

a. There are two important levels for cognitive modeling.

b. At the lower level, the natural description of the cognitive architecture is as a massively interconnected network of simple parallel numerical processing units: call this LNet (Lower level Network).

c. In LNet, elements of the problem domain (e.g., in syntax, words and phrases) are represented not by individual units, but as distributed patterns of activity; a given unit in LNet has no semantic interpretation by itself: it plays a small part in the representation of many different elements.

d. When the representations and computational processing of LNet are described at the higher level of the semantically meaningful activity patterns, we get descriptions of the cognitive architecture at the higher level. Such descriptions will often be approximate, idealized, or incomplete.

e. Unlike the lower level, descriptions of the higher level are not computationally uniform. Some of these descriptions involve symbolic computation with hard rules. Others involve local connectionist networks, in which individual units have semantic interpretations corresponding to those of the patterns of LNet. Such networks will be called HNets (Higher level Networks).

f. The symbols and rules of symbolic accounts correspond in LNet to patterns of activity and to the aggregate effects of groups of connections on these patterns of activity.

g. An important goal of connectionist modeling is to develop LNets supporting higher level descriptions that are simultaneously (i) sufficiently close to symbolic cognitive theory to explain the successes of symbolic accounts, yet (ii) sufficiently different to improve upon these successes.

The central idea of harmonic grammar is to start by partially specifying a LNet for a domain of linguistic interest, and then, rather than fully specifying and simulating it, as is conventionally done in connectionist modeling, to embody the most important aspects of LNet in a higher-level net HNet. This model, or rather a notational variant of it, HNet′, is what gets simulated. HNet′ (or, equivalently, HNet) is interpreted as grammar fragment expressing linguistic regularities via *soft rules*. Whereas symbolic rules of well-formedness have the form (a), the soft rules of harmonic grammar have the form (b).

(2) a. Condition X must never be violated in well-formed structures.

b. If Condition X is violated, then the well-formedness (harmony) of the structure is diminished by C_X.

The status of the two networks LNet and HNet are rather different. The level of LNet is presumably closer to the neural level, and therefore provides a more appropriate model for questions related to neurolinguistics (although the problematic relationship between connectionist and neural models, emphasized in Smolensky, 1988, suggests caution here). For language acquisition and real-time language processing models, as well, LNet would presumably be the more appropriate network. But for grammar, it is HNet that is the focus of attention.

This paper procedes as follows. We begin with a number of technical preliminaries which, after brief introduction of the linguistic problem of unaccusativity, motivate LNet, a partially specified lower level model for the unaccusativity data. We then derive a corresponding higher level model, HNet, and then its notational variant, HNet′, which is the model discussed in LMS_1. HNet′ allows standard connectionist learning to automatically extract from the data the constants C_X in the soft rules (b) of harmonic grammar. We close by summarizing the methodology and identifying some of its novel features.

Technical preliminaries

The subsymbolic approach outlined in (1) and its application to the domain of language presents the following research challenges, among others:[2]

(3) A. Representation:
1. Develop a formalism for higher level description of distributed representations: a calculus of patterns of activity (it is these patterns that correspond to symbols; (1f))
2. Apply this calculus to the representation of constituent structure

B. Processing:
1. Develop a formalism for higher level description of connectionist processing: a calculus of the aggregate effects of groups of connections on patterns of activity (it is these effects that correspond to rules; (1f))
2. Apply this calculus to the processing of constituent structure

The next four subsections successively address these four problems: A.1, A.2, B.1, and B.2.

A calculus of patterns of activity

A natural "calculus of patterns of activity" is straightforward: vector calculus, where the vectors are the lists of activation values for the units. The central idea of distributed representation (1c) can be stated very simply: it is activity vectors (not, e.g., individual units) that have semantic interpretations, i.e., interpretations as elements of the problem domain (the kind of information that is represented by symbols in the symbolic paradigm).

If different symbols are represented by different patterns of activity over the same set of units, as hypothesized in (1c), how is it possible to represent several such symbols at once? Vector calculus suggests a simple answer: by *superimposing*, i.e, adding together, the vectors representing the individual symbols. In the symbolic paradigm, structures are formed by some kind of (e.g., string or tree) concatenation of their constituents; in the subsymbolic approach, patterns combine by superposition rather than concatenation. This principle is discussed at some length in Smolensky (1986b), where an important consequence is derived: to a given network using distributed superpositional representations, there corresponds another network using local representations which provides a higher level description of the distributed network. The formal relation between the lower and higher level models can be thought of as a "rotation of the coordinates" in the activation space, so that the new coordinate axes lie along the directions of the distributed patterns with semantic interpretation; alternatively, we can think of this relation as changing variables from the lower level variables — units' activation values — to higher level variables — the strength of semantically interpretable patterns.[3]

Vectorial representation of constituent structure

How can simple vector addition replace concatenation? One basic problem that immediately suggests itself is that the former is a commutative operation, while the latter is not; e.g., `concat(a,b) = ab ≠ ba = concat(b,a)`, while **sum(a,b) = a+b = b+a = sum(b,a)**. A solution to this and related problems, called tensor product representations, is formalized and analyzed in Smolensky (in press 1). The first step is to recognize that vector superposition really represents *conjunction* rather than *concatenation*, and that concatenation, and other structure-building operations, can be achieved through conjunction together with *filler/role decompositions*. In such decompositions, a structure, e.g. `abc`, is described as the conjunction of an unordered set of propositions of the form, *structural role r is filled by f*, which are denoted by the *filler/role bindings f/r*; thus, e.g., `abc` is identified with the conjunction of the filler-role bindings $\{a/r_1, c/r_3, b/r_2\}$. The vector representing `abc`, under this filler/role decomposition, is $\mathbf{abc} = \mathbf{a}/r_1+\mathbf{c}/r_3+\mathbf{b}/r_2$ The vectors representing filler/role bindings, e.g. b/r_2, are constructed from vectors representing the unbound fillers and vectors representing unbound roles, e.g., b and r_2, by an operation from vector calculus called *the tensor product*: $\mathbf{b}/r_2 = \mathbf{b}\otimes r_2$. The tensor product is similar to the outer product of matrix algebra, e.g. $(x, y, z)\otimes(\alpha, \beta) = (x\alpha, x\beta, y\alpha, y\beta, z\alpha, z\beta)$ except that the result is interpreted not as a matrix but as another vector; the resulting vector can in turn be used recursively in further products, allowing recursive representations employing higher-order tensors. Smolensky (in press 1) analyzes these ideas — decompositions of structures into conjunctions of filler/role bindings, the superpositional representation of

conjunction, and representation of filler/role variable bindings via the tensor product — and shows that together they formalize and generalize a number of previous connectionist approaches to representing structure, and that they represent structured data in a way that naturally permits the usual features of connectionist processing, e.g., massively parallel (and structure-sensitive) associative processing, graceful degradation, and statistical learning.[4]

Thus, for the purposes of this paper, we assume:

(4) **Tensor product representation of structure**
At the lower level, in LNet, a structure s is represented by the activation vector $\mathbf{s} = \sum_\alpha \mathbf{c}_\alpha$, where each $\mathbf{c}_\alpha$ represents a constituent of s, which is a filler/role binding f_α/r_α with respect to some filler/role decomposition of s; the constituent vectors are $\mathbf{c}_\alpha = \mathbf{f}_\alpha \otimes \mathbf{r}_\alpha$, where $\mathbf{f}_\alpha$ and $\mathbf{r}_\alpha$ are activity vectors respectively representing f_α and r_α (possibly recursively defined as tensor product representations themselves).

A calculus for connectionist processing

Viewed at the lowest level, connectionist processing is the spread of numerical activation by some set of numerical equations in which the connection strengths enter as parameters. A calculus of the aggregate effects of connection strengths on patterns of activity relies on the idea that these activation equations are trying to achieve some characterizable end product: a pattern of activity that encodes a set of inferences which is justifiable from some notion of statistical inference. Smolensky (1983, 1986a) developed such a higher level description, deriving from a well-defined statistical inference problem a measure called the *harmony function H* whose global maximum constitutes the solution to the inference problem. In a large variety of connectionist models, the activation functions turn out to be implementing local maximization of this function, which can be written very simply:

(5) $$H(\mathbf{a}) = \sum_{i\,j} a_i w_{ij} a_j = \mathbf{a}^{\mathrm{T}}\mathbf{W}\mathbf{a}$$

where $\mathbf{a} = (a_1, a_2, \ldots)$ is the total activity vector of the network, and $\mathbf{W} = \{w_{ij}\}$ is the matrix of connection weights. (The negative of H is often referred to as "energy"; Hopfield, 1982; Hinton & Sejnowski, 1983, 1986.)[5] In a variety of networks, including both feed-back and feed-forward architectures, each update of the units' activations will increase H (Cohen & Grossberg, 1983, Golden, 1986, 1988, Hopfield 1982, 1984, 1987, Smolensky, 1983, 1986a, Hinton & Sejnowski, 1983, 1986.) Thus for a major subset of connectionist models, it is appropriate to regard the goal of the spread of processing to be the creation of an activation vector that maximizes H. For the purposes of formulating approximate higher level accounts of connectionist processing, we will thus assume that such harmony maximization is what processing in LNet achieves, even though a more detailed lower level account might well need to consider conditions under which the network fails to actually find the global maximum of H. Thus for our purposes we assume that under suitable approximation or idealization, the following holds:

(6) **Principle of harmony maximization**
Given an input vector **i** and connection weights **W**, processing in LNet establishes activity vectors **h** and **o** over the hidden and output units, respectively, that maximize $H(\mathbf{a})$, where $\mathbf{a} = (\mathbf{i}, \mathbf{h}, \mathbf{o})$ and the harmony function H is defined in (5).

Given the connection between H and statistical inference, this principle can be interpreted as follows: the network draws a set of inferences that provides a best fit to the input data and the statistical constraints embodied in **W** (e.g., the "Best Fit Principle" of Smolensky, 1988;Golden 1988). The value of H that is achieved in this maximization process is a quantitative measure of the degree to which it is possible to meet the statistical constraints in **W** by appropriately processing the given input data. This motivates the following central assumption of harmonic grammar:

(7) The H value achieved in processing an input is a quantitative measure of that input's well-formedness. An informant's judgements of acceptability of sentences can be modeled as a monotonic function f of the harmony values achieved in processing those sentences (the higher the harmony, the more acceptable).

In this paper, we take f to be the particular monotonically increasing function $f(x) = (1+e^{-x})^{-1}$: a logistic; the same methodology could be carried out for any other choice of (differentiable) f.

Connectionist processing of constituent structure

Now we bring together (4) and (5), assuming that the activity vector a in (5) is the structure representation s in (4). Then (6) states that processing in LNet maximizes:

(8) $$H(\mathbf{s}) = \mathbf{s}^T \mathbf{W}\, \mathbf{s} = \sum_\alpha \mathbf{c}_\alpha^T \mathbf{W} \sum_\beta \mathbf{c}_\beta = \sum_\alpha H_\alpha + \sum_{\alpha,\beta} H_{\alpha,\beta}$$

Here, the sum $\sum_{\alpha,\beta}$ is understood to include *one* term for each pair of distinct constituents $c_\alpha \neq c_\beta$ in s, and the first- and second-order harmony terms are defined by:

(9) a. $$H_\alpha = \mathbf{c}_\alpha^T \mathbf{W}\, \mathbf{c}_\alpha = \sum_{i,j} (\mathbf{c}_\alpha)_i\, w_{ij}\, (\mathbf{c}_\alpha)_j$$

b. $$H_{\alpha,\beta} = \mathbf{c}_\alpha^T \mathbf{W}\, \mathbf{c}_\beta + \mathbf{c}_\beta^T \mathbf{W}\, \mathbf{c}_\alpha = \mathbf{c}_\alpha^T [\mathbf{W}+\mathbf{W}^T]\, \mathbf{c}_\beta = \sum_{i,j} (\mathbf{c}_\alpha)_i [w_{ij}+w_{ji}] (\mathbf{c}_\alpha)_j$$

For example, if s = abc is decomposed as $\{a/r_1,\ c/r_3,\ b/r_2\}$, then $H(\mathrm{abc}) = H_{a/r_1} + H_{b/r_2} + H_{c/r_3} + H_{a/r_1,\ b/r_2} + H_{a/r_1,\ c/r_3} + H_{b/r_2,\ c/r_3}$. H_α is the internal harmony associated with constituent c_α, and $H_{\alpha,\beta}$ is the pairwise harmony arising from combining the constituents c_α and c_β in the same structure. Note that terms depending on more than two constituents cannot arise because H is quadratic.

Now if our specification of LNet were sufficiently complete that we knew the weight matrix W and the activity patterns representing the constituents $\mathbf{c}_\alpha$, we could compute each of the terms in (8) through the equations (9). This would amount to computing at the lower level. Alternatively, *we can operate with the harmony terms in (8) directly*, treating each H_α and $H_{\alpha,\beta}$ as an independent variable; this is computing at the higher level. Exploitation of this alternative is a central innovation of harmonic grammar.

The distinction between computing at the lower and higher levels can be viewed as follows. Equation (5) expresses the harmony as a function of the *lower level variables*: activities of individual units and strengths of individual weights. Equation (8), on the other hand, expresses the harmony as a function of the *higher level variables*: the constituents in the structure. The derivation of (8) from (5) [and (4)] amounts to a *change of variables* from the lower level variables associated with units and connections to the higher level variables associated with constituents. These higher level variables will shortly be used to define HNet.

We conclude these preliminaries with a few remarks.

(10) a. In an given problem, some of the constituents c_α will be "inputs," others will be "outputs," and still others will be "hidden" constituents. E.g., a sentence interpretation model might take as input constituents a string of words, might produce as output constituents the elements of some meaning representation, and might fill in as hidden constitents, say, non-terminal nodes in a parse tree. The "inferred" constituents — the hidden and output constituents — are determined through the principle of harmony maximization (6): they are the choice of c_αs that maximize H in (8).

b. The higher level harmony values H_α and $H_{\alpha,\beta}$ will later be interpreted as the constants in the soft rules of (b). They will be computed by a numerical fit to data. When is this parameter fitting exercise appropriately constrained? Note that, since H in (8) is quadratic in the constituents, the number of different terms that may appear on the right-hand-side of (8) scales as $(\#fillers)^2$, while the number of possible structures on the left-hand-side of (8) scales roughly as $(\#fillers)^{\#roles}$. Thus, as the number of roles in the structure increases, the formalism rapidly becomes more and more constrained, and the significance of good parameter fit becomes increasingly more meaningful.

c. Is it correct to treat the higher level harmony variables H_α and $H_{\alpha,\beta}$ as independent? This clearly depends on the structure assumed in LNet. Crudely speaking, if there are many more lower level variables than higher level variables, it is likely that any set of values for the higher level variables H_α and $H_{\alpha,\beta}$ can be achieved by some choice of the lower level variables, the representations $\mathbf{c}_\alpha$ and weights W. On the other hand, if it is possible to identify some strong constraints on the lower

level variables — e.g., if a large number of different constituents were constrained to be represented as different activity patterns over a much smaller number of units — then, effectively, there might be fewer lower level variables than higher level ones, so that the space of possible values for the higher level variables might be genuinely constrained by the fact that they are derived from the lower level.

A sample application: Unaccusativity in French

Unaccusativity

Since the problem of unaccusativity is discussed in some detail in LMS_1, we are extremely brief here. In many languages, in particular French, intransitive verbs divide into two classes: *unergatives* and *unaccusatives*, which yield different acceptability judgements in certain syntactic environments called *diagnostic contexts* (here, simply "contexts"). Consider, for example, the sentence *La glace est facile à faire fondre* "Ice is easy to make melt." Here, the diagnostic context is Object Raising or "OR," which is a sentence frame _____ *est facile à faire* _____ having two slots; the first, "argument," slot is filled by the NP *La glace*, and the second, "predicate," slot is filled by the intransitive verb *fondre*. The data of LMS_1 are 760 such French sentences, generated from four different diagnostic contexts, 143 different intransitive verbs, and arguments with varying semantic features. The pattern of acceptability judgements for these 760 sentences is quite complex. The acceptability patterns across different contexts of roughly half the verbs can be explained by a standard symbolic syntactic account which postulates that all the diagnostic contexts have well-formedness conditions requiring the argument to be a deep Direct Object of the predicate, and that each intransitive verb is marked in the lexicon as requiring its argument to be either a deep Subject (unergative verbs) or a deep Direct Object (unaccusative verbs). The other half of the data can only be explained by assuming that the acceptability reflects not only these "structural features" (deep Subject, Direct Object), but also semantic features of the argument and predicate. At the same time, these semantic features alone do not seem to be sufficient either; structural and semantic features are both required to explain these data.

LNet

Applying (1g) and (4) to the case at hand, we assume:

(11) a. Symbolic structural descriptions of sentences are approximate higher level descriptions of the patterns of activity in a lower level model LNet representing those sentences. In particular, the argument of an intransitive verb fills, among other roles, either the structural role of deep Subject or that of deep Direct Object.

b. In particular, these patterns of activity can be approximated as tensor product representations based on a role filler/role decomposition exemplified as follows for the vector representing the Object Raising (OR) sentence *La glace est facile à faire fondre*:

La_glace_est_facile_à_faire_fondre =
La_glace/ARG + OR/CONTEXT + fondre/PRED + DIRECT_OBJECT/STRUCTURE =
A + C + P + S

That is, the vector representing a sentence can be approximated as the sum of four vectors, each of which represents a kind of constituent that is specially designed for the particular data under study: an argument A, a context C, a predicate P, and a "structure" (deep grammatical function) S (either Subject or Direct Object).

We will not further specify this partial description of LNet; in particular, we will not specify vectors representing the individual fillers and roles. In the most general case, these vectors may be presumed to be fully distributed, giving rise to a representation of sentences in which every unit is part of the representation of each constituent; it will not matter to our analysis whether this fully distributed case or a more localized special case obtains, e.g., one in which the four roles of (11b) are localized to disjoint regions of the network.

There is no particular point in drawing a picture of LNet; we need only imagine a large network holding a representation of the sentence as a pattern of activity which is the sum of four constituent patterns (each of which may well involve activation over the entire network), according to (11b). Three constituents of this vector — the argument, context, and predicate — are specified in the input: the surface word string of a given sentence. The fourth constituent — the structure feature — is not given in the input; it is "hidden" (10a), and must be inferred by the network through activation spread to maximize harmony. Following (7), the degree of acceptability of the sentence to the network is taken to be $f(H)$, where H is the harmony of this activation pattern, and f is the logistic function. In LNet, acceptability is a distributed property; there is no "output unit" giving the network's acceptability judgement.

Why do we assume the filler/role decomposition of (11b)? Because it is the simplest imaginable one with which to start. The remarkable success of the consequent model provides some evidence in favor of this very simple assumption. It should be made clear, however, that the methodology permits assuming a different filler/role decomposition of the sentences, and following it through to a corresponding higher level model HNet operating in terms of the different constituent filler/role pairs, just as we now do for the filler/role decomposition assumed in (11).

HNet

Combining (6), (7), (8), and (11b), we have:

(12) The acceptability of a sentence s consisting of the argument A, the context C, and the predicate P is:

$$\text{acceptability}(s) = f\,[\max_S H(\text{A+C+P+S})]$$

where S ranges over the two possible structures, Subject and Direct Object, and

$$H(\text{A+C+P+S}) = H_A + H_C + H_P + H_S + H_{AC} + H_{AP} + H_{AS} + H_{CP} + H_{CS} + H_{PS}$$

This equation for H involves a prohibitively large number of higher-level parameters H_α and $H_{\alpha\beta}$. We eliminate a great many of these parameters by appealing to a number of linguistically motivated constraints:

(13) a. H_A: assume all arguments in the sentences used are equally well-formed internally
b. H_P: assume all predicates in the sentences used are equally well-formed internally
c. H_S: assume the grammar has no intrinsic preference between deep Subjects and deep Direct Objects
d. H_{AC}: assume the grammatical restrictions on the diagnostic context can refer only to general semantic features of the argument (not, e.g., to specific NPs); we take these features to be VO (volitionality) and AN (animacy)
e. H_{AP}: assume the lexical entry for the predicate can only express preference for general semantic features of the argument (again, VO and AN)
f. H_{AS}: assume the grammatical preferences for semantic/structural correspondences of the argument can only depend on its general semantic features (VO, AN)
g. H_{CP}: assume the grammatical restrictions on the diagnostic context can refer only to general semantic features of the predicate (not, e.g., to specific verbs); we take these features to be TE (telicity) and PR (progressivity)[6]
h. H_{CS}: assume the grammatical restrictions on the diagnostic context can refer to the structure (Subject vs. Direct Object) of the argument
i. H_{PS}: assume the lexical entry for a predicate can include a structural preference, but not an absolute bias on grammaticality

Assuming these constraints to hold, and dropping H_A, H_P and H_S because they do not vary across sentences (13a-c), we can rewrite the harmony function:

$$(14) \qquad H(A+C+P+S) = H_C+H_{VO,C}+H_{AN,C}+H_{VO,P}+H_{AN,P}+H_{VO,S}+H_{AN,S}+H_{C,TE}+H_{C,PR}+H_{C,S}+H_{P,S}$$

Now we recognize this as the harmony function of another network, HNet, illustrated in Figure 1. HNet uses a local representation, with a single unit for each each context, argument feature, predicate feature, structural feature, and individual predicate. The units in HNet correspond to patterns in LNet; HNet is a higher level network that is *isoharmonic* to LNet: the harmony of corresponding states in the two models are the same.

This network can be used to compute acceptability as follows. A given sentence is represented over the input units. We activate which ever of the hidden units gives the greatest harmony; this can be achieved by having the two units compete so that the unit with the greater net input wins: the net input to each hidden unit is precisely the contribution to the total harmony that that hidden unit would make if it were to have activity value 1. The hidden units thus are a little "winner-take-all" group in which the winning unit gets activity value 1, and the other, 0. Now we compute the harmony H of the network as a whole using (5) (with the variables H_α and $H_{\alpha\beta}$ now playing the roles in (5) of the weights W). Putting this value H into f, we get the acceptability $f(H)$, following (7).

The weights in this network are the harmony values H_α and $H_{\alpha\beta}$ of (10); from the point of view of the original lower level model, each of these weights represents the aggregate harmony of a set of weights and activity vectors, as indicated in (9). We'd like to work backwards from the data to infer what these aggregate values must be in order to produce the observed well-formedness pattern, but training the harmony values that are distributed throughout this network is not straightforward. A few simple modifications in the network, though, will fix this.

HNet′

In order to perform standard supervised learning from the data, we now create a network HNet′ that is precisely equivalent to HNet, but which possesses a single output O unit which explicitly computes acceptability. The main trick, illustrated in Figure 2, is simple: replace the connection between input units α and β of HNet, carrying weight $H_{\alpha\beta}$, by a conjunction unit $c_{\alpha\beta}$ whose activity is the product $a_\alpha a_\beta$, and connect $c_{\alpha\beta}$ to O with a connection of strength $H_{\alpha\beta}$. Then the contribution to O's net input coming from $c_{\alpha\beta}$ is $a_\alpha a_\beta H_{\alpha\beta}$, which is just the amount of harmony in HNet contributed by the original connection between α and β. Thus the total net input to O is the total harmony H. If O uses f to transform its net input to its activation value, then its value is exactly the acceptability judgement of Figure 1.

The only remaining step concerns the hidden units. In defining HNet′, the trick of replacing HNet connections with conjunction units should not be applied to the connections to the hidden units; these connections just stay as in HNet. As noted above, the net input to each hidden unit is precisely the contribution to the total harmony that that hidden unit would make if it were to win the competition. Suppose we define the activation functions of HNet′'s hidden units so that their activity values prior to competition are just equal to their net inputs; this is also their contribution to H. To maximize H, we let these two units compete so that the one with the higher activity value retains its value, while the other has its value set to zero. Now, the hidden units send their activation values to O (along connections with strength 1); the hidden unit capable of contributing the greatest harmony sends that harmony value up O, whose net input now includes the correct contribution from the hidden units, namely, the harmony arising from picking the structural feature that maximizes harmony.

The network HNet′ we have just defined, shown in Figure 3, is precisely the network used in LMS_1; and we have completed its derivation from basic principles. This network can now be trained using the appropriate form of standard back propagation (Rumelhart, Hinton, & Williams, 1986), as described in LMS_1. Note that the assumption of independence of the higher level variables, discussed in (10c), is relevant here, if the training procedure incorporates no constraints between weights. Note also that the "learning" in HNet′ is not a plausible model of language acquisition (for one thing, positive and negative data are crucial); "learning" in HNet′ is purely a computational procedure for parameter fitting, a formal trick for automating (a particularly nasty) part of the job of the harmonic grammarian: determining the numerical constants C_X in the soft rules (b). Presumably, a plausible model of real language acquisition would operate at the lower level, in LNet, rather than in HNet′.

Summary of the methodology

The methodoloy of harmonic grammar exemplified above can be summarized as the following series of steps.

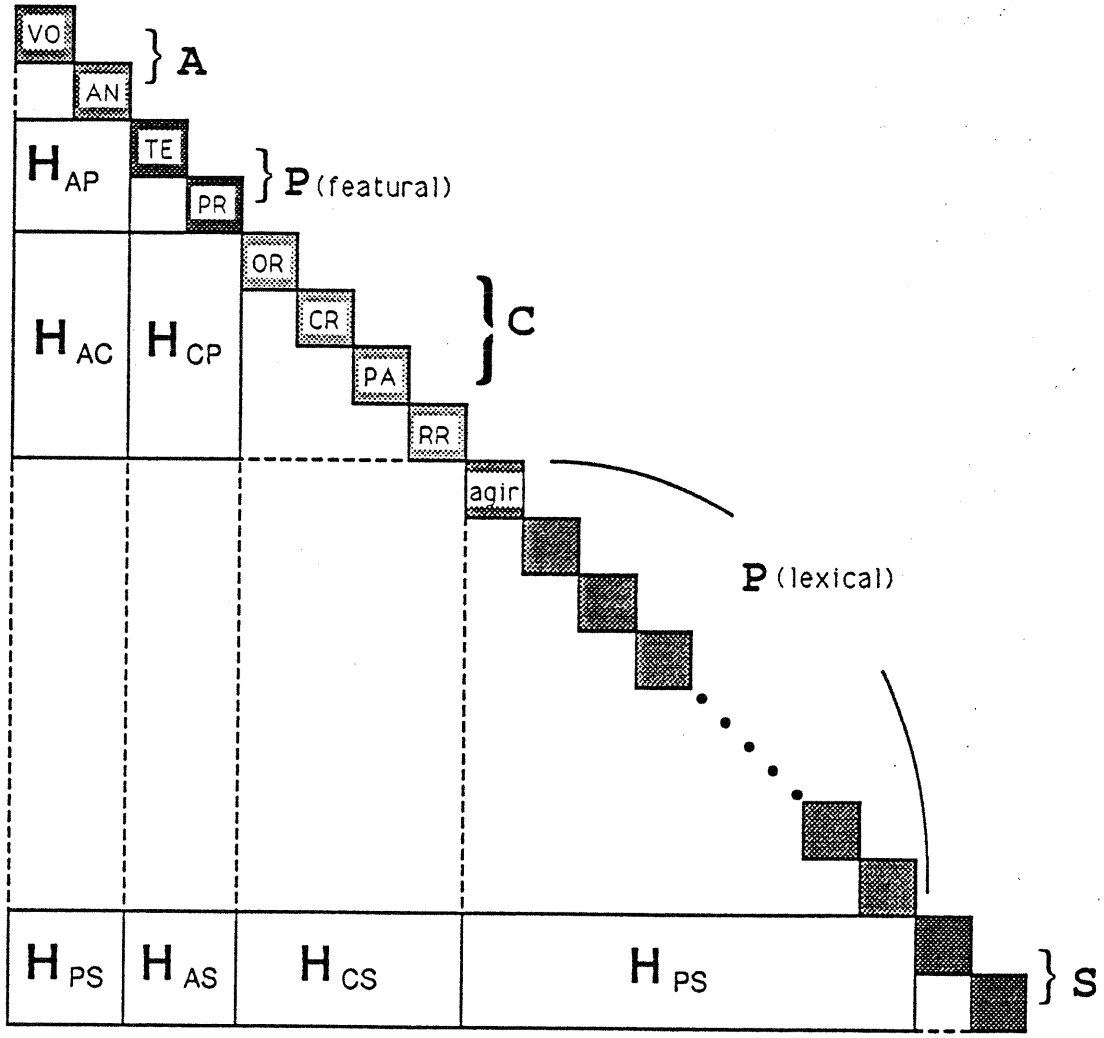
(15) a. Choose a filler/role decomposition for the structures whose well-formedness is to be accounted for.
b. Postulate a lower level model LNet using a tensor product representation with this filler/role decomposition.
c. Take the formula (5) for the harmony of LNet in terms of its weights and activities, and change variables ...
d. ... to get a formula (8) for the harmony as function of the constituents of the structure being represented; this function involves aggregated harmony values indexed by pairs of constituents; treat these values as independent high level variables.
e. Prune the number of these variables by appealing to linguistic constraints (at least) until the number of variables is considerably fewer than the number of data points to be accounted for.
f. Embody the resulting harmony function as a local connectionist network HNet whose connection strengths are the high level harmony variables.
g. Create HNet′ by adding to HNet an output unit that explicitly computes the harmony and corresponding acceptability value by means of additional conjunctive units and winner-take-all linear hidden units.
h. Train HNet′ using more-or-less standard connectionist supervised learning.
i. Interpret HNet′ as embodying soft grammatical and lexical rules.
j. Analyze these rules for new linguistic insight into the original linguistic problem.

Step (15j) is the subject of current research.

The method exhibits the following novel features:

(16) a. It is founded on a distributed lower level connectionist model that is only partially specified.
b. It operates primarily through a higher level formalism that approximately describes certain aspects of the aggregate behavior of the lower level network in terms of another, local, connectionist network.
c. The grammatical and lexical rules of the formalism are soft, and represent a set of quantified tendencies; but the model is fully formal, in that it makes precise predictions (even graded ones) of acceptability or well-formedness.
d. The strength of the soft rules is determined automatically from the data.
e. Existing linguistic knowledge plays the important role of constraining the form of the grammar.

Figure 1.


VO
AN
A
H AP
TE
PR
P (featural)
H AC
H CP
OR
CR
PA
RR
C
agir
P (lexical)
H PS
H AS
H CS
H PS
S

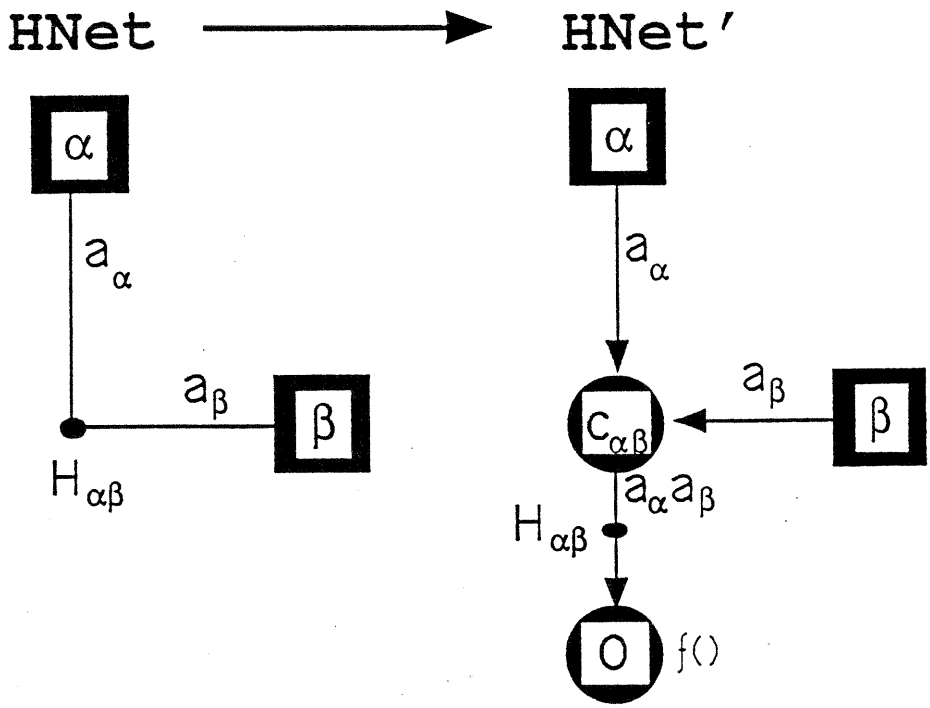

HNet
HNet'
α
a_α
a_β
β
$H_{\alpha\beta}$
α
a_α
a_β
β
$C_{\alpha\beta}$
$a_\alpha a_\beta$
$H_{\alpha\beta}$
O
f()

Figure 2.

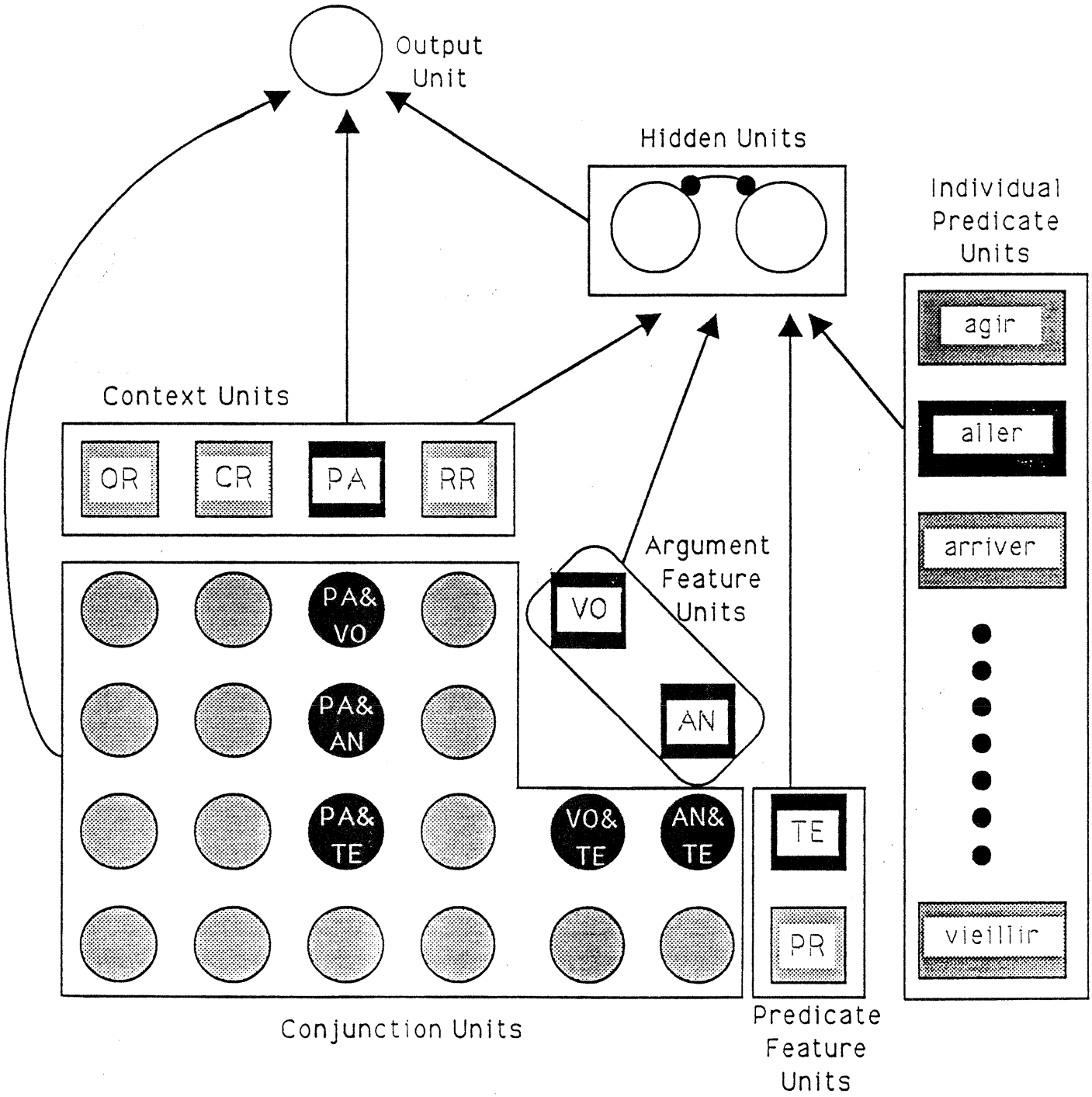

Output Unit
Hidden Units
Individual Predicate Units
agir
aller
arriver
vieillir
Context Units
OR
CR
PA
RR
Argument Feature Units
VO
AN
PA& VO
PA& AN
PA& TE
VO& TE
AN& TE
TE
PR
Conjunction Units
Predicate Feature Units

Figure 3.

Footnotes

1. Other connectionist approaches to linguistics appealing to the notion of harmony include John Goldsmith's "harmonic phonology" (Goldsmith, to appear) and George Lakoff's "cognitive phonology" (Lakoff, 1988).

2. In addition, there are corresponding problems related to learning, but these are not yet relevant to this research.

3. In Smolensky (1986b), the dynamical question is also considered: do the lower and higher level models evolve isomorphically in time? In this paper, we do an end run around the dynamics, working directly with the optimal equilibrium states the (incompletely specified) dynamics is trying to find.

4. Smolensky (in press 2) uses this technique as the technical basis for a reply to the putative dilemma of Fodor & Pylyshyn (1988): connectionism must choose between associationist and structure-sensitive processing.

5. This simple form for H arises from treating biases as weights to an extra unit with constant value 1, and treating input lines as though they originated in units interior to the network. In this form, H in the text is maximized when each unit achieves its maximum or minimum activation values. Networks whose units are not driven to their limits — e.g., quasi-linear units with sigmoid non-linearities, discussed in Smolensky (1986b) and very popular since Rumelhart, Hinton, & Williams (1986) in "back-propagation networks" — can be analyzed by adding to H a term $-\sum_i h(a_i)$ which does not introduce further interactions among the units, but is designed to penalize units with extreme values. E.g., for the popular logistic non-linearity $a_i = (1+e^{-input_i})^{-1}$, we set $h(a) = a\ln a+(1-a)\ln(1-a)$. This function h, like the other terms in H, has an interpretation in terms of statistical inference and information theory.

6. This constraint is particularly important since without it, every pair of context and individual predicate would have its own free parameter, giving rise to 572 parameters of this type alone — with only 760 data points to fix the parameters.

References

Cohen, M. A. & Grossberg, S. (1983). *Absolute stability of global pattern formation and parallel memory storage by competitive neural networks.* IEEE Trans SMC-13, 815–825.

Fodor, J. A. & Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition,* **28**, 3–71.

Golden, R. M. (1986) The "Brain-State-in-a-Box" Neural Model Is a Gradient Descent Algorithm. *Mathematical Psychology,* **30-1**, 73–80.

Golden, R. M. (1988) A Unified Framework for Connectionist Systems. *Biological Cybernetics,* **59**, 109–120.

Goldsmith, J. (to appear). Harmonic phonology. To appear in J. Goldsmith (ed.) *Proceedings of the Berkeley Workshop on Nonderivational Phonology.* Univ. of Chicago Press.

Hinton, G.E. & Sejnowski, T.J. (1983). Analyzing cooperative computation. *Proceedings of the Fifth Annual Conference of the Cognitive Science Society.* Rochester, NY.

Hinton, G.E. & Sejnowski, T.J. (1986). Learning and Relearning in Boltzmann Machines. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group, *Parallel distributed processing: Explorations in the microstructure of*

cognition. Volume 1: Foundations. Cambridge, MA: MIT Press/Bradford Books.

Hopfield, J.J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences, USA* 79, 2554–2558.

Hopfield, J.J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the National Academy of Sciences, USA* 81, 3088–3092.

Hopfield, J.J. (1987). Learning algorithms and probability distributions in feed-forward and feed-back networks. *Proceedings of the National Academy of Sciences, USA* 84, 8429–8433.

Lakoff, G. (1988). A suggestion for a linguistics with connectionist foundations. In D. Touretzky, G. Hinton, & T. Sejnowski (eds.), *Proceedings of the 1988 Connectionist Models Summer School.* Morgan Kaufmann Publishers.

Legendre, G., Miyata, Y. & Smolensky, P. (1990) *Harmonic Grammar - A formal multi-level connectionist theory of linguistic well-formedness: An application.* Technical report #90-4, Institute of Cognitive Science, University of Colorado at Boulder.

Rumelhart, D.E., Hinton, G.E., & Williams, R.J. (1986). Learning internal representations by error propagation. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group, *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 1: Foundations.* Cambridge, MA: MIT Press/Bradford Books.

Smolensky, P. (1983). Schema selection and stochastic inference in modular environments. *Proceedings of the National Conference on Artificial Intelligence.* Washington, DC.

Smolensky, P. (1986a). Information processing in dynamical systems: Foundations of harmony theory. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group, *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 1: Foundations.* Cambridge, MA: MIT Press/Bradford Books.

Smolensky, P. (1986b). Neural and conceptual interpretations of parallel distributed processing models. In J. L. McClelland, D. E. Rumelhart, & the PDP Research Group, *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 2: Psychological and biological models.* Cambridge, MA: MIT Press/Bradford Books.

Smolensky, P. (1988). On the proper treatment of connectionism. *The Behavioral and Brain Sciences.* 11, 1–23.

Smolensky, P. (in press 1). Tensor product variable binding and the representation of symbolic structures in connectionist networks. *Artificial Intelligence.*

Smolensky, P. (in press 2). Connectionism, constituency, and the language of thought. In B. Loewer & G. Rey (Eds.), *Fodor and his critics.* Blackwell's.