Connectionism and Harmony Theory in Linguistics

Alan Prince, Paul Smolensky

CU-CS-533-91  July 1991
Connectionism and Harmony Theory in Linguistics

Alan Prince, Paul Smolensky

CU-CS-533-91 July 1991

Department of Computer Science
University of Colorado at Boulder
Campus Box 430
Boulder, Colorado 80309-0430 USA

(303) 492-7514
(303) 492-2844 Fax

Class notes from a 4-week course of the same name taught at the Linquistics Institute, University of California at Santa Cruz, sponsored by the Linquistics Society of America
ANY OPINIONS, FINDINGS, AND CONCLUSIONS OR RECOMMENDATIONS EXPRESSED IN THIS PUBLICATION ARE THOSE OF THE AUTHOR(S) AND DO NOT NECESSARILY REFLECT THE VIEWS OF THE AGENCIES NAMED IN THE ACKNOWLEDGEMENTS SECTION.
Connectionism and Harmony Theory in Linguistics

LSA Linguistic Institute
University of California, Santa Cruz
LINS 247
July 1991

Topics

Spreading Activation & Harmony Maximization

*Spreading activation*

PDP-I:1,2

*Harmony maximization*

PDP-III:3[49-53]; PDP-II:14[7-38]; PDP-I:6

Goldsmith-Larson Dynamic Linear Model

Goldsmith, John:

Local modeling in phonology.

A dynamical computational theory of accent systems.

Prince, Alan:


Closed-form solution of the Goldsmith-Larson Dynamic Linear Model of syllable and stress structure and some properties thereof

Inductive Learning & Error Minimization

*Hebb rule*

PDP-I:1[31-40]; PDP-III:4[83-86],5[90-93]

*Delta rule*

PDP-III:[86-89,93-96,121-130]; PDP-I:11

Past Tense Acquisition Model

PDP-II:18
Pinker, Steven & Prince, Alan.

Rules and connections in human language.

On language and connectionism: Analysis of a parallel distributed processing model of language acquisition. [Pinker & Mehler.]

**Back Propagation**

PDP-III:5[130-137]; PDP-I:8

**NetTalk**

Sejnowski, Terrence J. & Rosenberg, Charles R. Parallel networks that learn to pronounce English text.

**Recurrent back propagation**

PDP-I:8[354-361]

Elman, Jeffrey. Finding structure in time.

**Continuity & Similarity**

**Hungarian Vowel Harmony Model**

Hare, Mary. The role of similarity in Hungarian vowel harmony: A connectionist account.

**Harmonic Grammar**

Legendre, Geraldine, Miyata, Yoshiro, & Smolensky, Paul:

Can connectionism contribute to syntax? Harmonic Grammar, with an application.

Unifying syntactic and semantic approaches to unaccusativity: A connectionist approach.

Harmonic Grammar -- A formal mutli-level connectionist theory of linguistic well-formedness: An application.

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

**Harmonic Phonology**

Lakoff, George. A suggestion for a linguistics with connectionist foundations.

Goldsmith, John. Phonology as an intelligent system.

Prince, Alan & Smolensky, Paul. Class Notes

Sources

Unless otherwise indicated, all articles are in the Reader.

‘PDP-X:Y[a-b]’ denotes ‘PDP books, volume X, chapter Y, pages a-b’

On reserve in library & available through bookstore:


Available through Institute:

Reader: Published articles and reports

Class Notes:

To be cited as:

Linguistic Background (incomplete)
LINS 247
Connectionism and Harmony Theory in Linguistics
P. Smolensky/ A. Prince
July, 1991


Kiparsky, P. (ca. 1980) "Vowel Harmony," ms. MIT.


McCarthy, J. and A. Prince (1986) Prosodic Morphology, ms. UMass Amherst and Brandeis University, Waltham.


I. Spreading activation & Harmony maximization

   Phonological prominence:
   - stress, syllable structure

II. Inductive learning & Error minimization

   Past tense inflection (English)
   Pronunciation (English)
   Induction of grammatical categories through distributional analysis (Eg.

IV. Continuity & similarity

   Harmonic Grammar

   Vowel harmony (Hungarian)
   Unaccusativity (French)

V. Harmonic Phonology

   - Universal phonology
   - Typologies
     - syllable structure
     - stress systems
   - Phonologies of particular languages

Assignment:
- 5-10 p. paper due last class
- Outline due one week before last class

Materials:
- Reader (published papers/reports)
- Class Notes (unpubl. drafts by P&S; handouts; overheads
- Bookstore
- Library
Summary of 7/8

\[ W_{ij} \]

Nodes - Activation
Connections - Weights

\[ \text{Net}_i = \sum_j W_{ij} a_j \]

\[ a_i = f(\text{Net}_i) \]

Iteration 0 1 2 3

\[ \begin{array}{cccc}
0 & 0 & 0 & 0 \\
1 & 0 & 0 & b \\
2 & 0 & b & b \\
3 & b & b & b \\
\end{array} \]

Activation vector unit 2 3 4...

State space, dynamical system

Analogy: spreading activation \( \sim \) derivation (e.g. autosegmental spreading) \( \rightarrow \) structure building

Goldsmith - Larson D.L.M.

Prominence
- shes (over syllable)
- sonority (over segment)

Examples:
\[ \alpha \beta \quad \text{final} \quad \bar{a} = (0 0 0 0 b) \]

1 0 \( (b b b b b) \) leftward spreading

-1 0 \( (-b b -b b) \) leftward alternation

\( -\frac{1}{2} \quad 0 \quad (\frac{1}{8} \frac{1}{8} b \frac{1}{8} b b) \) damped ""...

\( (1 1 1 1) \)

\( -\frac{1}{2} \quad -\frac{1}{2} \quad \frac{1}{2} \quad 2 \quad 2 \quad \frac{1}{2} \) interrelated...
Connectionist computation is optimization.

I. The basic (low-level) computational mechanism of con. is spreading activation.

1. Interesting things happen in new & different ways - due to this novel mechanism.
   1. Goldsmith & Larson's Dynamic Linear Model (DLM)
   [2... to come]

2. High level understanding of spreading activation - global analysis - is possible through ...

1. ... Linear Analysis
   1. The activation pattern is a vector in a high-dimensional state space evolving through time - a dynamical syst.
   2. The connection strengths form a weight matrix which can be analyzed using linear algebra; i.e., the DLM can be solved in closed form.
   [3. Related methods allow us to use vectors to represent and matrices to process structured data...]

2. ... Harmonic Analysis
   1. In certain nets, activation spread steadily increases a numerical well-formedness measure on activation vectors -
      Harmony = \sum_j \sum_i a_i \cdot w_{ij} \cdot a_j

2. Activation spread in these nets is an algorithm for
3. H maximization can be interpreted as (soft) constraint satisfaction.

4. These constraints can be used to declaratively define grammar.
   Global H maxima define the grammatical structure (competency);
   local H maxima are computed by connectionist nets (performance).

3. Activation has several interpretations:
   1. (Graded) truth value of a hypothesis.
   2. Value of a (graded) linguistic quantity.
      1. DLM: (continuous) context-dependent prominence.

   [3. None – meaningless internal computational data.]
Harmonic nets:

1. Activation rule: 'If $Net > 0$, excite, else inhibit'

2. Connectivity:
   - Feedforward
   - Symmetric

3. Updating activations:
   - One-at-a-time ('asynchronous')
   - Smooth (unit activations change very little at each step)

Notes:
1. Ensures $\Delta q_i \Rightarrow \Delta H > 0$ 'each update separately locally bounded'
2. Ensures $\Delta q_i \Rightarrow \Delta H > 0$ 'each update separately globally bounded'
3. Ensures $\Delta q_i \Rightarrow \Delta H > 0$ 'all updates inhibitory'
GL DLM.

\[ b^t = \begin{pmatrix} 0 & 0 & 0 & b \end{pmatrix} \]

\[ H = \sum \sum a_{ij} w_{ij} \]

Time \[ \tilde{a} = \begin{pmatrix} 0 & 0 & 0 & 0 \end{pmatrix} \]

\[ \begin{array}{cccc}
0 & 0 & -b & b \\
0 & b & -b & b \\
-b & b & -b & b \\
\vdots & \vdots & \vdots & \vdots \\
\end{array} \]

\[ b^2 + (b)(-1)(b) = 2b^2 - b^2 = b^2 \]

\[ b^2 + (b)(-1)(-b) = 2b^2 - b^2 = b^2 \]

\[ H \text{ monotonically increases ... why?} \]

1. Net is feedforward 'all neurons go up'

2. Only one activation value changes at a time

Constraint satisfaction view:

If it is weak, i should be strong;
"weak" "weak" "strong" "strong" "weak" "weak"

Every rule application in this 'derivation' is harmonic.
Syllabification in Berber


\[ \sigma = 1 \text{ segment (except phrase-initially)} \]

\[ \text{coda} \rightarrow \leq 1 \text{ segment (except phrase finally)} \]

\[
\begin{array}{cccc}
\sigma & \sigma & \sigma & \sigma \\
tftkt & yarbbi & ratkhi
\end{array}
\]

What governs syllabification? 'Ideal' sonority profile:

Berber's 8-level sonority hierarchy:

\[
a > u, i > r, l > m, n > z > f, s, x > b, d, g > p, t, k
\]

\[
b > 6 > s > s > 2 > 1
\]

Dell-El-Medlaoui Algorithm:

For \( S = 8, 7, \ldots, 1 \)

SCAN LEFT-TO-RIGHT

IF FIND \( S_i S_{ih} \) THEN BUILD \( S_i = S_{ih} \)

ATTACH any remaining unsyllabified segments as CODAS

Ex.: tftkt \( \rightarrow \) tftkt \( \rightarrow \) tf tkt \( \rightarrow \) tf tkt
Harmonic BrbrNet

... –λ –λ –λ –λ –λ ... \( \vec{a} \)

... ... ... \( \vec{b} \)

Each \( a_i \in [0,1] \) during computation; finally,
each \( a_i = 0 \) ('non peak') or \( a_i = 1 \) ('peak')

\( a_i = \pm \{ \text{segment } i \text{ is a peak} \} \)

Each \( b_i \) is a measure of the sonority of segment \( i \);
(In G-L, \( b_i = \text{son}(s_i) \); here, \( b_i = 2^{-\text{son}(s_i)} \); \( \text{son}(s_i) \epsilon \{0,1,2,3\}; \)

'Exponential sonority scale in Brber'.)

Activation rule:

To \( a_i \): add \( c \cdot \text{Net}_i \); if this is > 1, set \( a_i = 1 \)

\( c \) small \( \Rightarrow \) \( a_i < 0 \), set \( a_i = 0 \)

Note:
1. 'If Net > 0, excite, else, inhibit'.
2. Connections are symmetric.
3. Activation updating is smooth.

: Harmonic

Theorem: 1. For sufficiently large \( \lambda \), \( \vec{d} \) is a local \( H \) maximum
if \( \forall a_i \epsilon \{0,1\} \) and the syllabification satisfies
the Brber syllabification constraints: *VV; *CCC

2. For sufficiently large \( \lambda \), \( \vec{d} \) is a global \( H \) maximum
if \( \forall i: a_i \epsilon \{0,1\} \) and corresponds to the correct Brber
Harmonic Grammar View of BrbrNet

Grammar of soft rules:

\[ a_i \xrightarrow{-\lambda} a_w \]  
* \((+\text{syl}_i)(+\text{syl}_{i+1})\)  
\begin{align*}
\text{strength: } & \lambda \\
\text{subtract 'strength' from } H & \text{ if violate} \\
\text{add 'strength' to } H & \text{ if satisfy}
\end{align*}

\[ a_i \xrightarrow{2^{\text{son}(s_i)}} a_j \]  
\((+s_i) \& (+\text{syl}_i)\)  
\text{strength: } 2^{\text{son}(s_i)} - 1

Principle of Harmonic Structural Assignment:

Assign the structure that maximizes \( H \)
Learning from Experience

The problem: suppose we have a collection of input patterns $P_1, \ldots, P_m$

which are matched with output target patterns $T_1, \ldots, T_n$.

Can we get a network to learn to produce $T_i$ given $P_i$?

Concretely, find $w_{ji}$ for $j$

Specifically, $j$

We have $\text{net}_j = a_1 w_{j1} + a_2 w_{j2} + \cdots + a_n w_{jn}

= \sum_i a_i w_{ji}$

And:
- excite out $j$ if $f(\text{net}_j) > 0$
- inhibit out $j$ if $f(\text{net}_j) < 0$

We therefore manipulate $\text{net}_j$ to achieve desired goal.

The inputs are fixed so we work on the weights.

First idea: Hebb Rule:

reward node-wise correlation.

- set $w_{ji} = a_j a_i$
- sum over all input patterns
Example of Hebbian Learning

![Diagram of Hebbian Learning](image)

<table>
<thead>
<tr>
<th>P</th>
<th>q</th>
<th>p∧q</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>-1</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
</tbody>
</table>

Technical Background

Vector \( \mathbf{x} = (x_1, x_2, \ldots, x_n) \) helps us to speak of chunks as significant:

"Pattern" = vector

\( \mathbf{p}_1 = (1, 1) \quad \mathbf{p}_2 = (1, -1) \)

Dot product \( \mathbf{a} \cdot \mathbf{x} \) as a number not another vector

\( (a_1, a_2, \ldots) \cdot (x_1, x_2, \ldots) \)

\( = a_1 x_1 + a_2 x_2 + \ldots + a_n x_n \)

\( = \sum_i a_i x_i \)
Matrix - rectangular array

\[
\begin{array}{cccc}
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9 \\
\end{array}
\]

Example

\[
\begin{align*}
\mathbf{w}_1 &= \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \\
\mathbf{w}_2 &= \begin{pmatrix} 4 \\ 5 \\ 6 \end{pmatrix}
\end{align*}
\]

Dot Product between Vectors of unit length.

\[
\mathbf{a} \cdot \mathbf{x} = 1 \cdot 1 \cdot \cos \theta
\]

Perpendicular (Orthogonal) Vectors

\[
\mathbf{a} \cdot \mathbf{x} = 0
\]

Dot Product gives correlation.

\[
\begin{align*}
\mathbf{w}_1 &= \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \\
\mathbf{w}_2 &= \begin{pmatrix} 4 \\ 5 \\ 6 \end{pmatrix}
\end{align*}
\]
Two Interesting Cases

\[(a \ b \ c) \begin{pmatrix} x \\ y \\ z \end{pmatrix} = (ax + by + cz)\]

\[\begin{pmatrix} a \\ b \\ c \end{pmatrix} \begin{pmatrix} x & y & z \end{pmatrix} = \begin{pmatrix} ax & ay & az \\ bx & by & bz \\ cx & cy & cz \end{pmatrix}\]

Hebb en bref:

\[\vec{w}_{ij} = \begin{pmatrix} \text{out}_i \\ \text{out}_j \end{pmatrix} \begin{pmatrix} \text{in}_1 \\ \text{in}_2 \end{pmatrix}\]

\[W = \begin{pmatrix} \text{out}_1 \cdot \text{in}_1 & \text{out}_2 \cdot \text{in}_2 \\ \text{out}_1 \cdot \text{in}_2 & \text{out}_2 \cdot \text{in}_1 \end{pmatrix}\]

\[= \begin{pmatrix} \text{out}_1 \\ \text{out}_2 \end{pmatrix} \begin{pmatrix} \text{in}_1 & \text{in}_2 \end{pmatrix}\]
**Multiplying Matrices**

If $A \cdot B = C$, then:

$$c_{ij} = \text{row}_i(A) \cdot \text{col}_j(B)$$

For example:

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} ax + by \\ cx + dy \end{pmatrix}$$

**Adding Matrices**

To add two matrices, add corresponding elements.

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} + \begin{pmatrix} e & f \\ g & h \end{pmatrix} = \begin{pmatrix} a+e & b+f \\ c+g & d+h \end{pmatrix}$$

**Adding Vectors**

Add vectors by adding corresponding elements.

$$\begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} z \\ w \end{pmatrix} = \begin{pmatrix} x+z \\ y+w \end{pmatrix}$$
Total Hebb Experience

- Start off $w_{ji} = 0$
- $W = \sum_{p} (s_{p}^{i}) (\text{input}_p - p)$

Q: What if a trained net faces a new input?

A: Compute

$$\overrightarrow{\text{out}_{\text{new}}} = W \cdot \overrightarrow{\text{input}_{\text{new}}}$$

$$= \left[ \sum_{s} (s_{i}^{o}) (\text{input}_{s}) \right] \cdot (\text{input}_{\text{new}})$$

DOT PRODUCT!

How the trained net behaves

1. If $\text{input}_{\text{new}} \perp$ all items in training set, output is $\emptyset$.

2. If $\text{input}_{\text{new}} \perp$ all but one, get a scaled version of the output for that one.

3. If $\text{input}_{\text{new}}$ is non-orthogonal (similar to several), output is a weighted sum of outputs proper to them, weighted by degree of similarity. (Dot product)

$$\overrightarrow{\text{out}_{\text{new}}} = \sum_{k} (w_{k}^{i}) \overrightarrow{\text{input}_{k}} \cdot \overrightarrow{\text{input}_{\text{new}}}$$
### Non Hebb-Learnable

<table>
<thead>
<tr>
<th>NODE1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>OUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_1$</td>
<td>+</td>
<td>-</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>$P_2$</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>$P_3$</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>$P_4$</td>
<td>+</td>
<td>-</td>
<td>-</td>
<td>+</td>
</tr>
</tbody>
</table>

### Hebb

<table>
<thead>
<tr>
<th>$w_{51}$</th>
<th>$w_{52}$</th>
<th>$w_{53}$</th>
<th>$w_{54}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_1$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P_2$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P_3$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P_4$</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Diagram

[Diagram of a network with nodes and connections labeled.]
**Delta Rule**

\[ \Delta w_{ji} = \varepsilon (t_j - a_j) a_i \]

\[ \Delta W = \varepsilon \sum_k \left( \frac{e_k}{\text{error}_k} \right) \text{ (input}_k \text{)} \]

**Using the Delta Rule**

**Inputs:** \[ \mathbf{E} P_1, \ldots, P_n \]

**Output Targets:** \[ \mathbf{E} T_1, \ldots, T_n \]

**Procedure:**
- Present Input Set \((1 \text{ by } 1)\)
- Compare Output Produced by net to desired target
- Make step proportional to error \((\varepsilon)\) in right direction

**Do it again**

And again

\[ W \leftarrow W + \Delta W \]
If such a set exists, a set of weight functions will find this procedure converges to

\[ E = \frac{1}{2} \sum (e - g)^2 \]

\[ w_i = -\frac{e_i}{\sum e_i} \]

\[ \text{sum squared error} = \]
Historical remarks

* Learning 'internal' representations - 'hand' vs 'easy' learning
  (Bottom-up learning; Bottom-up 'hidden units')

* Technical breakthrough

Define recursively: \( \delta_j = (\sum_{k} \delta_k W_{kj}) f'(net_j) \) interior

\[ \delta_j = 'error \ at \ j' \]

\[ \delta_j = ??? \ for \ 'hidden' \ unit? \]

Gradient descent: \( \Delta w = \text{direction of steepest descent in Error} \)

\[ \Delta w_{ij} = -\varepsilon \frac{\partial E}{\partial w_{ij}} \]

Optimization technique
\[ E = (t - o)^2 \]
\( \Delta x \Rightarrow \Delta y := \text{a change in } x \text{ of amount } \Delta x \text{ leads to a change in } y \text{ of amount } \Delta y \)

\[ \frac{\Delta y}{\Delta x} := \text{rate of change of } y \text{ w.r.t. } x \]

\( \Rightarrow \frac{dy}{dx} \) \hspace{1cm} \text{for very small } \Delta x, \Delta y

**Chain rule:**

\[ \frac{\partial z}{\partial x} = \frac{\partial z}{\partial y} \cdot \frac{\partial y}{\partial x} \]

\[ \frac{\partial z}{\partial x} = \sum_k \frac{\partial z}{\partial y_k} \cdot \frac{\partial y_k}{\partial x} \]

**Additive property:**

\[ \frac{\partial E}{\partial w_{ji}} = -\frac{\partial E}{\partial \eta_i} \cdot \frac{\partial \eta_i}{\partial w_{ji}} = \delta_j \cdot a_i \]

**Definitions:**

\[ a_j = f(n_j) \]

\[ \delta_j = -\frac{\partial E}{\partial n_j} \]

\[ n_j = \sum_i w_{ji} a_i \]
\[ a_i = \text{net}_i \ (\text{linear net}) \]

\[ t_3 = a_1, \quad \text{input:} \{0, 1\} \]

\[ a_3 = w_{32} a_2 = w_{32} w_{21} a_1 \]

\[ E = \frac{1}{2} (1 - w_{32} w_{21})^2 + \frac{1}{2} (0 - w_{32} w_{21})^2 = \frac{1}{2} (1 - w_{32} w_{21})^2 \]

Start off at \( \tilde{w} = (0, 0) \) \( \Rightarrow \) ????
$z = \frac{1}{2} \left[ 1 - xy \right]^2$
\[ z = \frac{1}{2} [ 1 - xy ]^2 \]
$z = -1/2 \left( 1 - xy \right)^2$
\[ z = \frac{1}{2} \left[ 1 - f(y^f(x))^2 + \frac{1}{2} [f(y/2)]^2 \right] \]
\[ z = \frac{1}{2} \left[ 1 - f(y+f(x)) \right]^2 + \frac{1}{2} \left[ f(y/2) \right]^2 \]
Figure 2: Hierarchical clustering results for 48 phonemes.

Figure 4: Fits based on linear regression of the three factor solution on the attribute dimensions "place of articulation" and "vowel height", plotted in Fig. 3. Data based on hidden unit activations. The groupings are based on a centroid hierarchical cluster analysis of the original correlation matrix.

Rosenberg
Cog.Sci. '87
Connectionist Learning

processing architecture
  +
E function
  +
minimization technique

\{ \text{connectionist learning algorithm} \}

Extensions of back prop:

- More general processing architectures
- Distal targets
- Jordan: intention \( \rightarrow \) sound \( \rightarrow \) motor \( \rightarrow \) sound

\[ \text{error} \rightarrow \text{production} \rightarrow \text{results} \rightarrow \text{mental model} \]

* Constrained weights
  - Eg: copies for processing strings
  - Output:
    \[ \begin{array}{cccc}
    & h_1 & h_2 & h_3 \\
    w_1 & & & \\
    w_2 & w_3 & w_4 & \cdots \\
    w_7 & w_8 & w_9 & w_{10} \\
    \end{array} \]

* Recurrent networks

* Different E
  - Occam's razor - simplicity vs accuracy
  - Particular kinds of target patterns
  - Eg: \((0001000)\) for classifier

Alternatives to supervised learning: no 'targets', new E

- Reinforcement learning
- Unsupervised learning
  - 'Completion task': observe & predict, extracting structure
  - No negative evidence

\(* \rightarrow \text{Elman} *)
Strategies & Principles of Connectionist Applications in Linguistics: A summary

Low-level strategy

1. Connectionist principle:
   Learning is some kind of statistical analysis of co-occurrence.

   Linguistic interpretation:
   Temporal sequence of connectionist patterns → temporal sequence of linguistic units

   Linguistic principle:
   Linguistic categories can be induced from positive examples thru the prediction of the next item.

   Example:
   Inducing word boundaries, lexical categories: Elman

2. Connectionist principle:
   Processing is activation-driven: The [continuous] activation of a unit is increased/decreased over time in proportion to the activation of its neighbors in the network.

   Linguistic interpretation:
   activation → feature value (e.g. accentual prominence, sonority, tone)
   network neighbors → tier neighbors

   Linguistic principle:
   The [continuous] derived value of a feature is increased/decreased over time in proportion to the derived value of that feature for its left- and right-neighbors in the tier.

   Example:
   Goldsmith & Larson theory of stress & sonority

3. Connectionist principle:
   Processing is similarity-driven.

   Linguistic interpretation:
   temporal sequences of activity patterns → segmental strings
   connectionist unit → distinctive feature
   similarity → sharing of distinctive features

   Linguistic principle:
   The tendency for a missing feature in segment $s$ to be filled with a value consistent with that of a preceding segment $s'$ is increased by
   similarity between $s$ and $s'$
   similarity between $s'$ and other segments near it
   proximity in string/time of $s'$ and $s$
Example:
Transparency in Hungarian vowel harmony: Hare.

Methodological aside:

Low-level strategy for application of connectionism to linguistics (radical):

Low-level connectionist networks [principles] → specific linguistic account/‘model’ [→ higher-level analysis (possibly)]

High-level strategy (conservative):

Low-level connectionist networks → high-level connectionist computational principles → linguistic principles defining a grammatical formalism → specific linguistic accounts

Note: This strategy is made possible only through technical innovations in the high-level analysis of connectionist computation.

High-level strategy

4. Connectionist principle:
Connectionist representations can be analyzed on a higher level as symbolic structures.

Linguistic interpretation:
Henceforth, all grammatical representations will be symbolic: traditional, unrevised. The rules too will be traditional in form, but their interpretation and interaction will be new. (Conservatism.)

5. Connectionist principle:
Processing (at both lower- and higher level) is harmony-driven.

Linguistic interpretation:
connectionist input → linguistic ‘input’ to grammar (string of words, phonemes)
connectionist well-formedness → linguistic well-formedness
pattern of activity arising from an input → structural description of the input
pattern of connections → grammar

Linguistic principle A:

Grammatical well-formedness is quantitatively measured by Harmony. The grammar assigns to any input the structural description that maximizes Harmony. This maximal Harmony value is the well-formedness of the structure: high for well-formed and low for ill-formed.

The terms in the Harmony function define a Harmonic Grammar, and may be interpreted as soft rules of the form:
If structural configuration X occurs in the structure, add $H_X$ to the Harmony of the
structure.
The numbers \( H_x \) can be automatically determined from the data via back propagation.

Note: Harmonic Grammar permits (but does not require) the study of **very complex interactions of multiple factors**.

Example: Unaccusativity phenomena in French: Legendre, Miyata, Smolensky

**Problem:** Understanding the grammar is almost as difficult as understanding the original data...

**Linguistic principle B:**

Optimality. A grammar is a *preference relation* \( \succ \) among structural descriptions; it assigns to any input that structure which is *optimal* (preferred by \( \succ \) to all other structures).

The preference relation \( \succ \) among structures is built up compositionally from preference relations among the substructures or subdimensions from which structures are built.

The basic preference relations and their means of combination are drawn from a universal repertoire.

6. **Connectionist principle:**
   Learning is error minimization.

Linguistic interpretation: ???
Example applications: ???
We chose four diagnostic contexts:

<table>
<thead>
<tr>
<th>Context</th>
<th>Example 1</th>
<th>Example 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object Raising (OR)</td>
<td>La glace est facile à faire fondre. (Ice is easy to make melt.)</td>
<td>Je croyais Marie déjà sortie. (I believed Marie already gone out.)</td>
</tr>
<tr>
<td>Croire “believe” (CR)</td>
<td>Parti avant l’aube, Pierre est arrivé à destination le jour même. (Gone before dawn, Pierre arrived at his destination the same day.)</td>
<td>Gone before dawn, Pierre arrived at his destination on the same day.</td>
</tr>
<tr>
<td>Participial Absolute (PA)</td>
<td>La neige fonduée a formé de la boue. (The melted snow formed mud.)</td>
<td></td>
</tr>
<tr>
<td>Reduced Relatives (RR)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Specifying HNet' based on linguistic analysis

Data: 760 French sentences generated from the four diagnostic contexts, 143 intransitive verbs, and 3 types of arguments with their acceptabilities judged by human informants in 5 graded levels.

Choice of constituents:
Inputs: We chose the following to code each of the 760 French sentences:
(a) Argument: two semantic features - AN (animate) and VO (volitional).
(b) Verb: two aspectual features - TE (telic) and PR (progressive); 143 individuated verbs as lexicon.
(c) Context: four diagnostic contexts (OR, CR, PA, and RR)
Hidden: two hidden constituents, motivated linguistically by the unergative/unaccusative distinction.

Choice of constituent interactions:
175 independent quadratic terms representing pairwise constituent interactions. (143 lexical, 32 non-lexical)
E.g., we assume each diagnostic context can interact with (have preferences for) semantic features of the argument and aspectual features of the verb (include the terms Hn,AN, Hn,TE, etc.).
E.g., we assume each diagnostic context cannot interact with (have preferences for) individual verbs (eliminate terms Hn,AN, Hn,TE, etc.)

Treatment of hidden constituents:
We assumed that the two hidden constituents are mutually exclusive: turn on one of the hidden units that maximizes H and turn off the other.

Results

- Examples of soft rules:
  After training the HNet' on these data, we interpret the weights in HNet' as a set of soft rules:
  - verb *s'asseoir* “walk” prefers hidden constituent 2 - if hidden constituent 1 is chosen, decrease H (well-formedness) by 4.5.
  - verb *aller* “go” prefers hidden constituent 1 - if hidden constituent 1 is chosen, increase H by 3.4.

- The performance:
  As a collection of these soft rules, the grammar embodied in HNet' could account for the signs [+ or -] of acceptabilities for all of the 760 sentences, and for the graded acceptabilities [5 levels] for all but 30 of them.

- Strong interactions among syntactic and semantic components:
  The network achieved this performance with strong interactions among various syntactic and semantic components: almost every rule is strong enough to flip a decision between acceptable & unacceptable.

- Extendability of the grammar:
  We tested how well the grammar generated can extend to new verbs. After training on 536 sentences with 87 verbs, we gave the network 80 sentences with 20 new verbs. The network could account for the signs of acceptabilities of all but 2 of the 80 sentences by adding an appropriate new lexical entry for each verb.

- Predictions:
  The network was able to make certain predictions about acceptability patterns allowed for verbs and arguments with certain feature combinations. We are in the process of evaluating these predictions.
Goal: Account for the acceptability judgements of 8393 Ss
(3608 intransitive; 4785 transitive)
[an exploration in D.I.C.F.I.; for an exploration in T.I.C.F.I.,
see Prince & Smolensky, 1991]

Data:

- Constructions (11): OR, CR, PA, PE, RR; ON, PCL;
  PRQ, FOR, PR, IMP

- Targets: Subj, D.O.

- Embedded Predicate
  - Identity: 408 = 183 transitive + 225 intransitive
  - Features: Given aspectual features - TE, PR
    Learned (arbitrary) features - n = 6

- Embedded Argument Structure
  - Transitives: Subj, D.O., semantic
  - Intransitives: Arg functions: VO, AN, DEF
    deep G.R., Subj, D.O.

- Acceptability judgements: +, +?, ?, -, ?-

Account:

- Implemented as a local connectionist network
  - Trained using backpropagation on judgements
  - Current best account: if 8393 Ss:
    104 errors (1.2%) = 90 transitive (0.5%) = 14 intransitive (4%)

- As a set of Harmonic Grammar rules:
  - If construction = CR & target = D.O. & deep G.R. = D.O.
    then add .55 to Harmony = well-formedness
  - If predicate = morvrir 'die' & deep G.R. = Subj
    then subtract 2.6 from Harmony
  - If construction = CR & target = D.O. & Subj = +VO
    then subtract 1.5 from Harmony

- Harmonic Structural Assignment Principle:

  assign the structure that maximizes Harmony

- As the higher-level description of a lower-level
  connectionist network obeying certain general
  principles of connectionist representation and processing
  (in particular, pattern completion via Harmony maximization)

Basic linking assumption:

linguistic well-formedness = connectionist well-formedness

= Harmony
OPTIMALITY

Alan Prince
Center for Complex Systems
University of Arizona

Paul Smolensky
Center for Computer Science
University of Colorado

(1) Optimization in Grammar Choice (Chomsky, 1951, 1965)

UG(D) £ G, G2. . . . . All G's agree on D.
Eval (G') £ G. G* contains the real rules of L.

Chomsky's idea was to find a coding for the operations and structures of grammar such that the length (i.e., shortness) of the encoding of G was the measure of G's value.

- For phonology, we must determine Eval(LEXICON) £ Eval (Rules).
Otherwise, we can drive Eval(Rules) to optimum by no Rules.

(2) Optimization as a grammatical mechanism.
Simple Case:
Dooa (IN) £ OUT1, OUT2.
WP {OUT1, } £ OUT*.
The real output.

More generally,
B(IN, Dooa, OUT*) > B(IN, Dooa, OUT1).
"Greatest Benefit."

(3) The ray against optimization.

a. Have to use numbers -- parameter city, and use counting.
b. Many messy tradeoffs: if (k > k', k > k''), what then of d vs. k?
c. Optimization is computationally intractable, in general.

(4) Counter-argument:

c. It is not incumbent upon grammar to compute. Well-definition is all; not efficiency or even algorithmicity (Chomsky).

k'. Actually, ORDER, not counting, is the key.
b'. We will argue that many cases of apparent tradeoff can be handled without tradeoff calculation, by a principle of combining orders.

(5) Optimization as a grammatical mechanism: the case of Berber Syllabification (Ddl & El-Meddawi 1987, 1988)

(6) ANY segment may be syllabic:

(7) ONSETS: syllables must have onsets, except phrase-initially.

(8) SONORITY: syllabification is sensitive to relative sonority.
a. iMz * iZmi vs. tMz * tMzh
b. raLsh * raLshw1 i.e., mAl * MAl.

c. Ddl and E.L. establish the following 8-point hierarchy:

a > u > i > d > n > z > f > b > k

low V > high V > liq > nasal > voiced fric > vless fric > voiced stop > vless stop

(9) Ddl's Algorithm (DEA)

Form core syllables (X)Y ("CV"), where X is any segment and Y is successively replaced by the features describing the steps of the sonority scale; in descending order, iterate from Left to Right for each fixing of the nuclear variable Y.

(10) Secondary phenomena (to be ignored):
a. obstruents de-syllabified *# and *#
b. sonorants G's optionally desyllabify *#
c. tonotomorphemic geminates never break an onset.

(11) EXAMPLES:

f/ *saluh / £ > (A)aluh £ > (A)alu[h] £ > (A)alu[h] (IT)
f/ *xn- / £ > (A)xn £ > (I)xn[ ] £ > (I)xn[ ] (IT)
f/ *xn-n- / £ > (A)xn £ > (I)xn[ ] £ > (I)xn[ ] (IT)

(12) DEA serves to optimize the quality of syllables: high sonority items like to be nuclei.

(13) But DEA per se makes no contact with this. E.g. Why descend the scale?

(14) Harmonic Syllabification (HS).

Form the best core syllable from free material in the domain, good (r) = good (nuc). Iterate until you can do more, H[2] (G) = L.

(15) This is a fully 'Harmonic' process: 'Harmony' from Smolensky, 1986.

Go to the most well-formed available state.

(16) Note that HS is different from DEA. HS has no directional bias, does not sensibly run in a direction (LR) to deal with sonority plateaus.

(17) Evidence for directionality weak anyway.
a. /kwi/ £ kw, *kwr but better onset (DE note)
b. /kku/ £ kku, *kku but no final syllabic obstruent.
c. /byan/n/ £ byan, *yavan but no geminate qua CV.
d. /ugmn/ £ ugm, ugn but both ok. DE note only m, L.
e. /bbr/ £ bbr, only it bBr, DEA fails. Nb. closed o bad.

(18) Conclusion: optimization, with grades of well-formedness (representational Harmony), is a mechanism of grammar.

(19) To make this approach fly, we need two things:
a. Ways of evaluating combinations of phonological dimensions where the dimensions themselves needn't have an internal w.m. structure, but may be scalar. Harmony scale on dim. comb.

(presupposition: context markedness: SPE, Kean, Kon, Archangeli & Pulleyblank)
b. Ways of combining different relevant dimensional scales to any Harmony scale.

(20) Dimensional Combination. Example: weight and stress. (Priss, 1990)
a. Separate prominence (not markedness) scales: H>l, stress > unstress.
b. Combine implausibly: if H, then stressed. (ISP = Hay's 'quantity sensitivity').
c. Good are: H', L'. Bad is H''.
d. Unstressed light.

e. Weaker implication: if stressed then Heavy.

Good are: H', L'. Bad is: L'.

(21) Combination respecting strength of implications:

H', L' > L > H''. This is a Harmonic ordering, expressing degree of caused of the two dimensions.
(21) Combination of Harmony Scales. Example: 'unbounded' stress systems like Stress the rightmost heavy syllable, if no heavy then stress the rightmost syllable (R/R). Prince (1983) and Hayes (1989) see these as enhancements of prominence rather than 'unbounded foot' phenomena. We extend this result, thereby limiting feet to the authentic binary rhythmic units.

(22) The pattern R/R. Two factors:
a. WEIGHT. Heavy syllables should be stressed.
b. POSITION. Edgelessness: Stress should be near the end.

(22) WEIGHT dominates POSITION. WEIGHT >> POSITION. (Nl. One stress.)

(24) WEIGHT must be satisfied, if possible. But there can be several choices:
   a. cv.CVv, cc.CV, cv.CVv, ccv.CVv.
   b. cv, ccv, cv.CVv, ccv, cvv.

(25) Indeterminacies are resolved then by POSITION. (24b) is chosen. \( \Rightarrow \text{bat} \).

(26) No indeterminacy in (cv) \( \Rightarrow \text{cv,cv,cv} \).

(27) This is LEXICOGRAPHIC ORDER - a generalization of alphabetic order. Alphabetically, strings are ordered first by initial character (\( \Rightarrow \) WEIGHT), then ties are broken by the ordering on the second character (\( \Rightarrow \) POSN), etc. (cf. \( \Rightarrow \) Sibert, etc. NE: NO TRADEOFF.

(28) LEXICOGRAPHIC PRODUCT of two orders, \( P >> O \).

(29) Example: alphabetic ordering itself: POSN \( \Rightarrow \) ALPHA.

(30) Claim: Lexicographic Order is the essential mode of combining Harmony scales.

(31) Berber Syllabification is also a kind of Lexicographic Order: analyses in which /m/s are nuclei are absolutely preferred to competitors in which /m/s are not but /z/s are.

We can say:

\( x \)'s should be peaks \( >> y \)'s should be peaks, for \( x > y \) in sonority.

With suitable technical development, this might replace the serial (iterative) HS.

(32) Pronominal-driven Stress. Claim: works by Lexicographic ordering

PROMINENCE >> POSN, with development of notions of prominence and position.

(33) Hindi (modern) (Kelkar, 1968; Hayes, 1991)

'Stress falls on the heaviest available syllable, and in the event of a tie, the rightmost nonfinal candidate wins.' (Hayes)

(34) Heaviness: CvCe, CvCC (\( \mu \mu \mu > Cv \)), CvC (\( \mu \mu > Cv \))

(35) Heaviness is best:

\( \text{ki\text{DI\text{AR}}} \) ja\text{AA\text{AB}} as\text{BA\text{A}} ru\text{.pi),(\text{REEZ\text{gaa,rii)}}

(36) Among equals, last nonfinal is best:

\( \{\text{mu}\} \) ru\text{.KAA\text{ya}} FUS\text{.ia,kee} roo\text{.ZAA\text{a}}

\( \{\mu \mu \} \) A\text{A}\text{S\text{.maa,jaah}} a\text{a\text{a\text{a,MAA\text{N,jaah)}}}

(37) Analytic: \( \text{WEIGHT} \Rightarrow \text{POSN} \)

But POSN is constructed from NONFINALITY, and RTMOSTNESS.

(38) NONFINALITY >> RTMOSTNESS

(39) WEIGHT >> NONFINALITY >> RTMOSTNESS.

(40) Piha (Everett & Everett, 84; Everett, 88; Levin, 85; Davis, 88; Hammond, 90; Hayes, 91).

"Stress the rightmost token of the heaviest syllable type in the last 3 syllables of the word."

(41) Parallel: 'rightmost heavy or rightmost syllable'.

HEAVINESS >> EDGEMOSTNESS

(42) What is 'Heaviest'? C = voiceless, G = voiced const.

a. ?A.bai 'toucan'
   \( \Rightarrow \text{CV}\text{Gv} \)

b. ?a.bai PA 'Amapal'
   \( \Rightarrow \text{rimout, Cv wins} \)

c. so.Al.pi 'uqes'
   \( \Rightarrow \text{w > Cv} \)

d. KAA.gai 'word'
   \( \Rightarrow \text{CvGv} \)

e. ppal.GAI.bi.\text{ai} 'banana'
   \( \Rightarrow \text{Gv\text{vv} \in 3r window, nb, pou.} \)

f. ka.pi,ga,ka.gai,ka.bai BAi
   \( \Rightarrow \text{rimout: in window: nb, kai} \)

(43) Full 'Heaviness' Hierarchy:

\( \text{CV} \text{v} \Rightarrow \text{GV} \Rightarrow \text{CVG} \Rightarrow \text{CVGV} \Rightarrow \text{CVGV} \), where C=voiceless, G=voiced.

(44) Evidence that the stress is these:

a. Some speakers treat it as a beat with hand gestures.

b. Some speakers devise everything after stressed syllable.

c. Some speakers delete something after stressed syllable.

d. May be some tropism of tone to stress.

(45) ISSUE:

A) How is 3 syllable limitation established?

B) How is 'heaviness' hierarchy established?

(46) Ans. to (A). Domain of Stress is Minimal Prosodic Word, minWd.

(Cf. Prosodic Circumscript ions (McCarthy & Prince, 1990), whereby a prosodically delimited subdomain serves, in lieu of the expected morphological category, as the domain of a process.)

(47) Piha stress must lie in maximal minWd, for \( P = \sigma \).

(48) Evidence for involvement of Wd category. Closely-knit phrases show ONLY ONE STRESS when last word is 2 \( \sigma \).

ba\text{.sai} # bi.SAI 'cloth # red'

?a.ba,pa # go.CI 'city same # where-at'

7i.sii.HQA. # si.BAI.7i 'liquid fuel # much'

?a,pa.PAI # Hi,ta,ba 'head # hurts'

\( \text{ka\text{H\text{AJ}}} \) # ?o.ga,ga.GAI 'arrow # want!'

Analysis: \( \text{FWd} = \text{max minWd} \) is established at -edge of morph-word.

(50) ANS. to (48). 'Heaviness' Hierarchy
CVV > CVV > CVV > CVV > CVV
a. xVV > yVV in every case.
b. C > G > mil
This is just onset goodness (R. Shaefer).

Prominence Order = Weight >> Onset

(51) Fall Story.
IN-PWD >> WEIGHT >> ONSET >> RIGHTMOSTNESS.

Stress penultimate, final, antepenultimate, dependent on foot structure AND VV location.

(53) In words with only SHORT VOWELS, stress is penultimate:

Sata  pa.Thla dhana.Sera  bhina.Sera
ja.sin.Nema  sun.NARA

This is actually a Moraic-Trochaic (RL) pattern (Hayes) = weak vowels in open syllables reduce; those in closed do not; nb. no closed final syllables.

(54) LONG VOWEL FINAL: Stress is Final. For SHORT penult V-

adina.TAA  gora.min.TIL  eka.bat.TAA

LONG V in antepenultimate stressed if VV followed by two LIGHT:

ji.muta.BAA.bana  kaka.ROO.hari  BAA.sana

PENULT LONG V is stressed:

pu.RUU.kha  SAA.bhaha  ba.raba.MAA.saa

(55) ANALYSIS: Main Stress falls on MOST PROMINENT in final circumscribed window; when there is equality, penult is favored (Non-finality).

(56) Window: 2 units = FF, Fr, sF. = 2F, if headless feet exist (H66)

(57) Elementary Prominence Orders:
FOOT: Foot-heads >> Non-heads;
VV: VV >> V

(58) Relation: VV >> FOOT

(59) As usual, PROMINENCE >> POSN.
for POSN = NONFINALITY >> RIGHTMOSTNESS

(60) Cases:

... [CvC]CvCv... = prom/rightmost Fhead. dhana.SERA
dito  ja.sin.Nema
cw cv.CvCv  prom wins. ku(u).ra.NAA
[CV.Cv.Cv]CvCv  prom wins. BAA.sana
... [Cv.Cv]cv.  = prom/nonfinality wins. SAA.bhaha

(61) Fact Note: Following Hayes, we assume ...cv.CvCv... galau + beena = galauENea but CAA.rilo 'fourth'
VV shortens in nonfinal, non-mainstressed syllables. If wrong, then FROM = heavy.

that in light-heavy#, heavy is stressed.

(63) Broad Vista. To run grammar on optimization, we need to be able to say WHEN to perform
an operation. I.e. when it is better to act than to do nothing:
B (X -> Y) > B (X -> X )

(64) Basic Harmonic Condition: f(Y) > h(X). Well-formedness increasing: up Harmony Hill.
But not sufficient: else Wd = Taa.

(65) From universal perspective, need to order Transitions and combinations of Transitions and contexts to obtain markedness implications about processes.

(66) Example. LANGO Vowel Harmony = one property thereof. (Pulleyblank, Yes.)
a. ATR Harmony spreads forward from all vowels over a SINGLE CONSONANT:
in V1-CvCv, V2 harmonizes with 1st vowel.
b. But over TWO intervening Cs, he trigger V must be high:
only in hi-CvCv do we get spread of ATR.

(67) Appears to involve complex TRADE-OFF in virtues of spreading from various sources, and
difficulty of doing same in various environments. But there is NO TRADE-OFF.

(68) Approach. Suppose we know the following:
a. Spread ATR from HI > Spread ATR from Non-HI. "s INSTANT"
b. Spread over single C > Spread over CC. "DISTANCE"

(69) Form the Lexicographic Product

HEAVY >> DISTANCE: (a → v) + (b → v)

Spread from HI over C > Spread from H over CC
> Spread from Non-HI over C > Spread from Non-HI over CC.

(70) Now, an individual language may insert DO-THING anywhere in the order.
All things preferable will be done, then.

(71) Insert DO-Nothing at bottom: any action is better than inaction: no constraint, full ATR harmony.

(72) Lango. Insert Do-Nothing above last element in order.
Spreading from Non-HI-over-CC is worse than Nothing: But everything else is better. So as accordingly.

###
Constraint Interaction and Harmony Maximization in Grammar

Alaa Prince
Brandeis University

Paul Smolensky
University of Colorado

I. Basic Idea -- General
A. Picks up two themes in modern linguistic thought (see attached bibliography; most of the entries are concerned with these):

1. Constraints or processes that give way or are conditional.
   EG.
   a. Do such-and-such EXCEPT when the result conflicts with some other constraint. EG. Make final \( o \) extrametrical EXCEPT in monosyllables. EG. Foot form.
   b. Do such-and-such ONLY when necessary to achieve well-formedness. EG. Epenthesis a glottal stop ONLY to provide syllable with onset. Epenthesis a Vowel ONLY to make word big enough to hold foot.

2. Direct application of universal principles of markedness in the operation of individual grammars.
   EG. Prefer syllables to have onsets. Prefer the collocation of \( *Hl \) and \( *Atr. \) In the standard theory (eg. SPE, ch. 1-8, et seq.), there are no more than observations made about what grammatical processes happen to end up doing. But they can form the foundation for principle theory of grammar itself; can be directly appealed to in specific grammars.

B. Rapport with mode of functioning of central species of connectionist networks: harmony increase to maximum.

C. Fundamental harmonic Imperative: among the possibilities, choose the one that maximizes harmony (relative wf'edness).

II. What we need to carry out this project.
-Notion of preferences (approx = markedness)
-Ways of combining orders. (Lexico. Direct.)
-Way of assessing harmony of structures, as determined by constraint order. (LOCUS. Tableaux)

III. Constraint Domination. Some familiar basic types of constraint interactions.
A. Do something EXCEPT. A positively-regarded option is embedded lowdown.

1. Extrametricality: \( m\alpha n\alpha \), \( *\langle m\alpha n\alpha \rangle \) LexWD=PrWD >> EXM.

LexWD=PrWD. Lexical Words should be Prosodic Words, therefore contain feet.

EXM. Final syllables should be foot-loose.

2. Imbrication in Cibumba (Hyman, 1991):
   Short Base   Long Base
   \( f\alpha l \text {-i-} \ e \) \( *s\text{matik}-i \text {-i-} \ e \)
   \( \text{f}\alpha \text{l}-\text{t-} \ e \) \( *s\text{mati}-i \text {-k-} \ e \)

   Generalization. Stems CVC suffix \( \text{l} \); Longer bases insert \( \text{l} \) after last vowel in bases.
   Analysis. Take \( \text{l} \) to be default. Then suffix \( /l/ \) to PrWD.

FB >> PrWDW

FB. Feet should be binary at some level of analysis (\( \alpha \alpha \), \( \beta \beta \)).
PrWDW. PrWD words should be consist exhaustively of well-formed units.

3. Foot WFedness. (\( \#\alpha \alpha \# \), \( \#\beta \beta \), \( \#\beta \beta \))

FB >> GomWD=PrWD >> EXM >> PARSE >> RH

RH. Rhythmic Harmony. Trochaic feet \( \alpha \beta \), \( \beta \alpha \) are better than \( \gamma \) \( \beta \alpha \) PARSE. All material should be parsed into prosodic structure. (= Itô's Prosodic Licensing).

Say Stray Erasure en route to the Phonetics.

4. Inflection 1: gr-\( \text{-em-adwet.} \) *umgradwet. *gumradwet.

   Goodfirstsyll >> Leftmostness of prefix

   Goodfirstsyll. No Coca.

   Leftmostness of Prefix. 'Prefix' means affixal content is Leftmost in Affix + Stem collocation.


   Edge-preserve >> Rightmostness of suffix (light syll).

   Edge-preserve. Segment at edge of base should appear at edge of reduplicated base (pa-parth)

   Rightmostness of suffix. 'Suffix' means affixal content is Rightmost in Affix + Stem collocation.

   B. Do something ONLY. A rejectionist option is lowdown & overruled by a higher-order constraint that is, the high-order constraint can force the disfavored option.

   HAVE-ONSET >> FILL-NODES

   HAVE-ONSET = \( \sigma \) should dominate ON constituent.

   'Epenthesis' = Insert empty syllable structure (Selkirk, Itô).

   FILL-NODES = nodes should dominate segments - AVOID epenthesis.
Universal Typology in Harmonic Phonology:  
0th-order Syllable Structure

Paul Smolensky & Alan Prince  
July 25, 1991

Fundamental Definitions

**LOCS:**
Suppose given:

- a role-filler decomposition of \( S \) specifying for each \( s \in S \) a description of \( s \) as a multi-set \( M \) of role/filler pairs
- an ordering \( \Rightarrow \) of fillers
- a function \( \text{predom} \) specifying a predominant element of any multi-set \( M \) of role/filler pairs; define:
  \[ \text{residue}(M) = M \setminus \text{predom}(M) \]

Then the corresponding **Lexical Order on Composite Structures** \( S \) is the ordering on \( S \) defined recursively by:

\[ M_r \succ M_f \text{ iff} \]

\[ \text{filler}(\text{predom}(M_r)) \Rightarrow \text{filler}(\text{predom}(M_f)) \]

or

\[ \text{filler}(\text{predom}(M_r)) = \text{filler}(\text{predom}(M_f)) \text{ and } \text{residue}(M_r) \succ \text{residue}(M_f) \]

Two cases of interest:

**Filler-driven LOCS:** \( \text{predom}(M) = \) any role/filler pair in \( M \), containing a maximal filler (w.r.t. \( \Rightarrow \)).

**Role-driven LOCS:** \( \text{predom}(M) = \) a role/filler pair in \( M \), containing a maximal role, w.r.t. an ordering on roles \( \supset \) which must be further supplied

**SDG:** The **Strict Domination Grammar**

\( p_1 \Rightarrow p_2 \Rightarrow \ldots \Rightarrow p_n \)

is the preference relation \( \Rightarrow \) defined from a sequence \( p_1, \ldots, p_n \) of Boolean predicates over possible structural descriptions, defined as the following role-driven LOCS:

- the 'roles' are the predicates \( p_1, \ldots, p_n \) and the 'filler' of each role is the truth value of each predicate: \( M_r = \{ \pm p_1, \pm p_2, \ldots \} \) where the sign of \( p_k \) indicates whether \( s \) satisfies predicate \( p_k \) (+) or not (-). [We sometimes write \( M_r \) more mnemonic by suppressing the \( +p_k \) and writing the \(-p_k \) as \( \ast p_k \)]

- the ordering on fillers is \( + \supset \ast \Rightarrow \) (T \( \Rightarrow \) F; 'preferences')

- the ordering on roles is \( p_1 \supset \ldots \supset p_n \)

[Aside: If in the definition of SDG we change the LOCS from role- to filler-driven, then the order we get is 'Prefer fewer marks', where a 'mark' is a violated preference.]

**FUT(\( P \)):** The **Factorial Universal Typology** determined by a set \( P \) of universal preferences is the typology arising from the strict domination grammars determined all possible permutations of the preferences in \( P \).

[That is, as far as the grammatical module governed by the set \( P \) is concerned, FUT(\( P \)) characterizes the set of possible languages as exactly those arising from all possible orderings of the preferences in \( P \).]
0th-order (C/V) Syllable Structure

Goal:
To explain the typology:
onsets: required/optional/ * forbidden;
codas: * required/optional/forbidden

[nuclei: assume required by definition of syllable]

<table>
<thead>
<tr>
<th>onsets</th>
<th>required</th>
<th>optional</th>
</tr>
</thead>
<tbody>
<tr>
<td>codas</td>
<td>Σ_cv</td>
<td>Σ_cv_v_c</td>
</tr>
<tr>
<td>forbidden</td>
<td>Σ_cv</td>
<td>Σ_cv_v_c</td>
</tr>
<tr>
<td>optional</td>
<td>Σ_cv_v_c</td>
<td>Σ_cv_v_c</td>
</tr>
</tbody>
</table>

Main Claim:
The universal typology of C/V syllable structures is FUT(P), where the set of preferences P is:
ONS: Prefer syllables with onsets.
-COD: Prefer syllables with no coda.
PARSE: Prefer parsed segments.
FILL: Prefer filled syllable positions.

Note: Best syllable is CV, which satisfies all preferences. Thus /CV/ is assigned CV. So it is not possible to prohibit onsets or to require codas.

Onsets (ignore -COD for now):

Look at underlying form with preferred coda (i.e., none) and an onset wanting repair (i.e., none): /V/;
Possible structures:

\[ \sigma \]
\[ NUC \]
\[ V. \]
\{ * ONS \}

\[ \sigma \]
\[ ONS \]
\[ NUC \]
\[ V. \]
\{ * PARSE \}

\[ ^V. \]
\{ * FILL \}

Phonetic assumptions: unparsed segments are not realized (deletion); unfilled syllable positions are somehow realized (epenthesis).

The preferred structure is the one with the lowest mark; given an order
\[ p_1 \gg p_2 \gg p_3 \]
the preferred structure is the one having \{ * p_3 \}; the order of \( p_1 \) & \( p_2 \) don’t matter. So rather than 3! = 6 cases there are 3 different ones (ignore {} for now):

1a. ONS \( \gg \) \{PARSE \( \gg \) FILL\} \( \gg \) PARSE \( \gg \) ONS \( \gg \) FILL Preferred: ^V.
1b. ONS \( \gg \) \{FILL \( \gg \) PARSE\} \( \gg \) FILL \( \gg \) ONS \( \gg \) PARSE Preferred: [V]
2a. \{PARSE \( \gg \) FILL\} \( \gg \) ONS \( \gg \) \{FILL \( \gg \) PARSE\} \( \gg \) ONS Preferred: V.

(Here, \( \sim \) denotes the equivalence of two orderings, that is, that they determine the same maximally harmonious forms.)
Using the first instance of each case, we can view the three cases hierarchically by letting
1-1: Prefer one-to-one pairing of segments and syllable positions
     = either \{PARSE \gg FILL\} or \{FILL \gg PARSE\}
we get:

1. ONS \gg 1-1
   a. 1-1 = \{PARSE \gg FILL\} Preferred: ^V.
   b. 1-1 = \{FILL \gg PARSE\} Preferred: [V]
2. 1-1 \gg ONS
   1-1 = \{PARSE \gg FILL\} \sim 1-1 = \{FILL \gg PARSE\} Preferred: V.

That is:
1. Onsets are mandatory, since ONS is the predominant preference
   a. an onset is epenthized;
   b. an onsetless nucleus is deleted.
2. Onsets are optional, since it is more important not to epenthese or delete (1-1) than it is to
   have an onset.

Codas:

Same story, mutatis mutandis. Look at /CVC/ (preferred onset, coda needing repair).
Possible structures:

\[
\begin{align*}
\text{ONS} & \quad \text{NUC} & \quad \text{COD} \\
\text{C} & \quad \text{V} & \quad \text{C}
\end{align*}
\]

\[
\begin{align*}
\text{ONS} & \quad \text{NUC} \\
\text{C} & \quad \text{V} & \quad \text{C}
\end{align*}
\]

{ * -COD} \hspace{2cm} \text{CV.}[C] \hspace{2cm} \text{CV.C^*}

\{ * PARSE\} \hspace{2cm} \{ * FILL\}

Cases:

1. -COD \gg 1-1
   a. 1-1 = \{PARSE \gg FILL\} Preferred: CV.C^*.
   b. 1-1 = \{FILL \gg PARSE\} Preferred: CV.[C]
2. 1-1 \gg -COD
   1-1 = \{PARSE \gg FILL\} \sim 1-1 = \{FILL \gg PARSE\} Preferred: CVC.

That is:
1. Codas are prohibited, since -COD is the predominant preference
   a. an extra nucleus is epenthized;
   b. a coda is deleted.
2. Codas are optional, since it is more important not to epenthese or delete (1-1) than it is to
   lack a coda.

Note: For the onset discussion, the relevant case of FILL is 'Prefer filled onsets' (FILL_{ONS}), while for
the coda discussion, it is 'Prefer filled nuclei' (FILL_{NUC}). These need not be assumed to be subsumed
under a single general preference, FILL. Similarly, the relevant cases of PARSE are PARSE_v and
PARSE_c for the onset and coda discussions, respectively. If there is but one general preference FILL
and another such PARSE, then these general preferences must be relatively ranked once and for all in
a given grammar, predicting that, PARSE_v \gg FILL_{ONS} iff PARSE_c \gg FILL_{NUC}, which in turn implies
that in any $\Sigma_{CV}$ language, either onsets and nuclei are both epenthesized ($/V/ \rightarrow ^\ast V$, and $/CVC/ \rightarrow CV.C^\ast$.) or nuclei and codas are both deleted ($/V/ \rightarrow [V]$ and $/CVC/ \rightarrow CV.[C]$). Presumably this is empirically false, so we must allow the two kinds of PARSE and two kinds of FILL preferences to each have their own place in the dominance hierarchy.
uninflected | nonfuture | future | gloss
--- | --- | --- | ---
(A) kentapai | ke$tapai-in | kentapai-ur | dugong
kejar | ke$jar-in | ke$jar-ur | river
miyar | miy$m-in | miyar-ur | spear
yarput | yarput$m-in | yarput-ur | snake, bird
yaraman | yar$m-an-in | yaramanur | horse
pirjen | pirjen-in | pirjen-ur | woman

(B) mela | mel$m-a | mel-ur | sea
wanka | wank$a-n | wanka-ur | arm
kunka | kun$k-a | kunk-ur | groin
guuka | guu$k-a | guu$k-ur | water
ka$a | ka$t-a | ka$t-ur | child
gawa | gaw$a-n | gawu-ur | wife
kenji | ken$-ji | kenji-ur | wife
pape | pape$n | pape-ur | father's mother
wite | wite$n | wite-ur | mother's father

(C) yalulun | yalu$m-n | yalulur | flame
mayaran | mayar$an | mayara$r | rainbow
wiwalan | wiwala$r | bush mango
karikarin | karikari$m-n | karikari$ur | butter-dish
yilyilin | yily$m-lin | yily$lur | oyster (sp.)

(D) jurarapin | jurarap$m-in | jurarap$ur | shark
galu$ | galu$m-in | galu$ur | story
putukan | putuk$a-n | putuk-ur | short
murkuniman | murkun$m-an | murkunima$r | cullah
gawu$mawun | gawu$mawun | gawu$mawur | termite
tipiti$p | tipiti$p-in | tipiti$ur | rock-cod (sp.)
japutimun | japutimun | japutimun$ur | older brother
munjunmunmun | munjunmunmun | mun^munmun$ur | wooden axe
tupmutumpur | tupmutump$mur | tupmutump$ur | dragon fly

(E) yukarp$a | yukarp$m-an | yukarp$ur | husband
wulunkar | wulun$m-an | wulunkar$ur | fruit (sp.)
wusatil | wu$atil-in | wusatil$ur | meat
kantukantun | kantuk$a$ntun | kantuk$a$ntur | red
karwakarwana | karwakarw$m-an | karwakarw$mur | wattle (sp.)

CVC-

uninflected | nonfuture | future | gloss
--- | --- | --- | ---
tera | ter-in | ter-ur | 'thigh'
yura | yur-in | yur-ur | 'body'
ruta | rut-in | rut-ur | 'neck'
tjute | tjil-in | tjil-ur | 'hair'
marpa | mar-in | mar-ur | 'hand'
punta | pur-in | pur-ur | 'excrement'
wunta | wun-in | wun-ur | 'rain'
kanta | kanm-in | kan-ur | 'grass'
njuta | njit-in | njit-ur | 'fire'
Connectionism for Linguistics

Paul Smolensky

*LSA Linguistic Institute
July 17, 1991*

Collaborators:
- Géraldine Legendre (Linguistics, Colorado)
- Yoshiro Miyata (Cognitive Science, Chukyo)
- Alan Prince (Linguistics, Brandeis)

Connectionist Computation
- What is it? Low level characterization:
  - Data: activation values
  - Processing: spreading activation
  - Learning: associationist strength modification with experience

Why is it? Cognitive motivations:
- Biological neural networks: abstract out a *computational* description
- Psychological models of:
  - Perception
  - Memory
  - Associationist processing
  - Empiricist learning

Computation & linguistics
- Psycholinguistics
  - Neurolinguistics
  - On-line processing
  - Acquisition

The form of grammar

If cognition is *connectionist* computation, what are the implications for grammar?

Relation between connectionist and symbolic computation
- Eliminativism: Connectionist *replaces* symbolic
- Implementationalism: Connectionist *implements* symbolic
- Revisionism: Connectionist *revises/enriches* symbolic

[A major research program is required to pull this off; multiple levels.]

Grammar
- Eliminativism: Grammar is a fiction.
- Implementationalism: Grammar is unchanged.
- Revisionism: Grammar inherits new *kinds* of rules & representations from connectionist computation

Requires *high-level* characterization of connectionist computation [which in turn requires a major research program to develop]
Computing done by linguist vs. computing done by speaker/hearer.

Examples of new ways of computing/thinking:
1. Continuous representations
2. Similarity-based processes
3. Parallel soft constraint satisfaction
4. Quantitative measures of well-formedness
5. Optimality as main organizing principle

Are these good ways of computing/thinking? You tell me.

Plan for lectures:
For \( p \in \{1,2,3,4,5\} \)
   A. Develop high-level principle \( p \) of connectionist computation.
   B. Propose a way of importing it into linguistics.
   C. Develop an example application.

Logical progression of principles: 1 \( \rightarrow \) 5 (lowest, ‘most connectionist’ \( \rightarrow \) highest, ‘most symbolic’)

Conservative presentation of linguistic applications: 5 \( \rightarrow \) 1 — wins.

Today: 5; Next week: 4 - 1

5. Optimality as an organizing principle of grammar

A. Connectionist principle:

**Harmony maximization.** In a central and interesting class of connectionist networks, activation flow creates from an input a pattern of activity that [locally] maximizes a numerical well-formedness measure, the *Harmony*:

\[
\Sigma_i \Sigma_j a_i w_{ij} a_j
\]

Linguistic interpretation:
connectionist input \( \rightarrow \) linguistic ‘input’ to grammar (string of words, phonemes)
connectionist well-formedness \( \rightarrow \) linguistic well-formedness
pattern of activity arising from an input \( \rightarrow \) structural description of the input
pattern of connections \( \rightarrow \) grammar

Here, all grammatical representations will be symbolic: traditional, unrevised. The rules too will be traditional in form, but their interpretation and interaction will be new. (Conservatism.)
B. Proposed linguistic principle:

**Optimality.** A grammar is a preference relation > among structural descriptions; it assigns to any input that structure which is optimal (preferred by > to all other structures).

The preference relation > among structures is built up compositionally from preference relations among the substructures or subdimensions from which structures are built.

The basic preference relations and their means of combination are drawn from a universal repertoire.

Linguistic-internal motivation:
Grammars are more explanatory when they explicitly state well-formedness constraints rather than describing procedures to satisfy those constraints.
(Prefer principles to rewrite rules/transformations.)

Phonology has not made the progress towards this goal that syntax has — because the constraints are so often of the form, 'prefer ... if possible' rather than 'obey ... or be ill-formed'.
(But re:syntax, cf. Sadock's constraints on inter-component discrepancies.)

C. Applications in phonology [for many more, cf. Prince & Smolensky, LINS 247]

a. Berber syllabification

[insert brief statement of problem]

**Idea:** Dell—El-Medlaoui algorithm is trying to build the best syllables possible, as defined primarily by the goodness of their peaks. Replace the algorithm with a formal statement of this well-formedness metric.

More sonorous segments make better syllable peaks:
align sonority prominence hierarchy 8=7=...=1 with syllable structure prominence hierarchy: peak=non-peak
(8,peak) > (7,peak) > ... > (1,peak).

Given a syllabification $s$, let $M_s$ = the (multi-)set of peaks in $s$.
Let $\text{first}(M_s) = \text{best peak of } M_s$, and let $\text{rest}(M_s) = M_s \setminus \text{first}(M_s)$.
Then recursively define:
$M_s > M_t$ iff
$\text{first}(M_s) > \text{first}(M_t)$
or
$\text{first}(M_s) = \text{first}(M_t)$ and $\text{rest}(M_s) > \text{rest}(M_t)$
Note: This means of composing the ordering $>$ on peaks to an ordering $>$ on syllabifications is a generalization of alphabetical ordering of strings: reinterpret \textit{first} as 'left-most element'. It turns out to be useful quite generally in the theory, so we dub it Generalized Lexicographic Ordering, \textbf{GLO}. To define a GLO $>$, we must specify the sets $M_s$, the function \textit{first}, and an ordering $>$ of individual elements of $M_s$.

\textbf{b. Lardil}

A \textbf{Strict Domination Grammar} is a preference relation $>$ defined from a sequence of Boolean predicates over possible structural descriptions

$$p_1 \gg p_2 \gg \ldots \gg p_n$$

by the following case of GLO:

For a given structure $s$, let $M_s = \{ \pm p_1, \pm p_2, \ldots \}$ where the sign of $p_k$ indicates whether $s$ satisfies predicate $p_k$ (+) or not (−). Let $\text{first}(M_s) =$ first element of $M_s$. On each predicate separately, $+p_k$ (preference satisfaction) $>$ $-p_k$ (preference violation).

Note: This is a form of \textit{markedness theory}. Each $-p_k$ is a kind of 'mark' on the structure, and the ordering $>$ gives a formal means of deciding which structure as a whole is least marked, by examining individual marks.

For clarity, we write 'Prefer $p_k$' for $p_k$ and 'Avoid $q_k$' when $p_k = \sim q_k$.

The Lardil preferences:

- $p_1$: Avoid subminimal words [less than one foot].
- $p_2$: Avoid non-final defective syllables.
- $p_3$: Avoid complex consonants in coda position [allowed: one non-back coronal].
- $p_4$: Prefer filled onsets.
- $p_5$: Prefer free final V.
- $p_6$: Avoid defective syllables.
- $p_7$: Avoid free material.
- $p_8$: Prefer coincidence of stem & syllable boundaries.