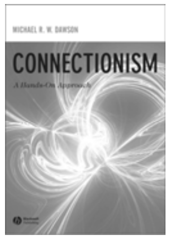
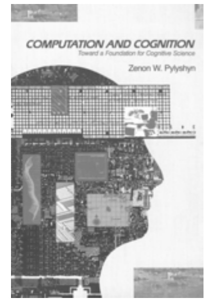
Computation And Cognition – And *Connectionism*



Michael R.W. Dawson,
Biological Computation Project,
University of Alberta

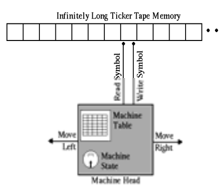
Computation And Cognition

- If cognition is information processing, then explanations of cognition must be framed in the tri-level hypothesis
  - Computational
  - Algorithmic
  - Implementational



Computational Level

- What classes of information processing problems is an architecture capable of solving?
- For classical cognitive science, can an architecture solve the same problems as a universal Turing machine?



Algorithmic Level

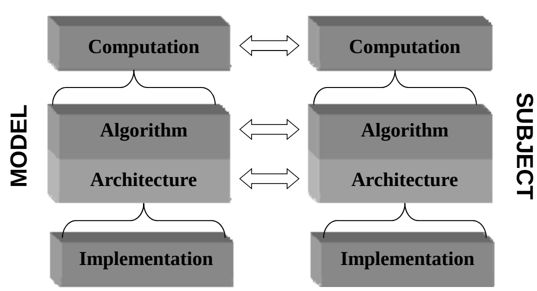
- What operations are executed to solve a particular information processing problem?
- For classical cognitive science, what sequence of symbol manipulations is executed?
  - The problem of "what to do next"

Implementational Level

- What physical properties are responsible for bringing information processes to life?
- For classical cognitive science, this often involves searching for functional primitives that are subsumed in neural circuitry
  - Functional architecture
  - Language of thought

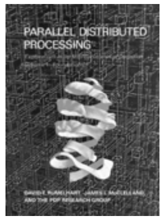
Comparative Cognitive Science

- Validation of theories requires establishing equivalences in the context of the three levels



Against Classical Cognitive Science

- Shortly after 1984, we were in the midst of the connectionist revolution
  - Classical approach is not biologically plausible
  - Brain is a brain, not a digital computer
- PDP as an information processing alternative
  - No rules
  - No symbols



Connectionism Critiqued

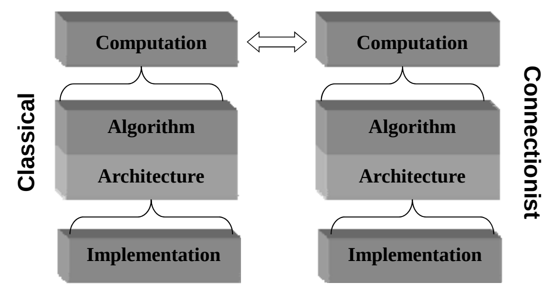
- Connectionism received a warm – and sometimes hot – reception from Pylyshyn
- “Connectionism appears to have fatal limitations. The problem with Connectionist models is that all the reasons for thinking that they might be true are reasons for thinking that they couldn’t be *psychology*” (Fodor & Pylyshyn, 1988, p. 48)
- “Voodoo. People are fascinated by the prospect of getting intelligence by mysterious Frankenstein-like means – by voodoo. And there have been few attempts to do this as successful as neural nets” (*Scientific American* quote, 1994).

Tri-Level Consideration

- Classical and connectionist cognitive science are frequently portrayed as being antagonistic opposites
- However, my own work is interested in exploring similarities between the two approaches
- This is done in the context of the tri-level hypothesis

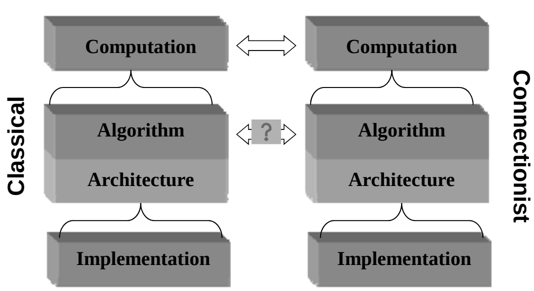
Computational Equivalence

- Many different kinds of proofs exist suggesting that PDP networks are equivalent to UTMs



Algorithmic Equivalence?

- What kinds of algorithms do networks execute?
- Can they be related to classical algorithms?



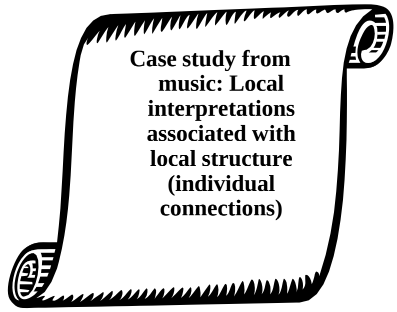
PDP Models Are Hard To Understand

- Problem: researchers rarely describe network algorithms, because network interpretation is not an easy task
- “If the purpose of simulation modeling is to clarify existing theoretical constructs, then connectionism looks like exactly the wrong way to go. Connectionist models do not clarify ideas, they obscure them” (Seidenberg, 1993)


Mark Seidenberg

Synthesis, Emergence, Analysis

- However, if you go to the trouble of peering into networks, you can be rewarded
- My students and I have spent a great deal of time interpreting PDP networks
- Synthesis
  - Build a network
- Analysis
  - Interpret its internal structure
- Emergence
  - Learn surprises about the phenomena by discovering network properties

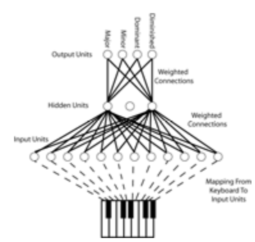


Chord Classification Problem

- One important task in music theory training and piano technical training is chord identification
- Example: listen to a chord
  - What general type of chord is it?
  - Independent of key
  - Independent of inversion

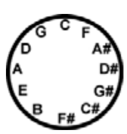
The Pitch Class Network

- 4 output value units
  - Major chord
  - Minor chord
  - Dominant chord
  - Diminished chord
- 3 hidden value units
- 12 input units
  - Piano keyboard
  - One octave
  - Starting note is A
- 48 training patterns
- Dawson/Schopflocher rule
  - Learning rate of 0.005
  - Weight start ±0.10
  - Biases start at 0.00
- Converged after 3964 epochs

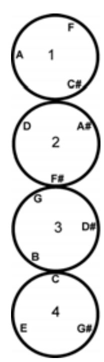


Network Analysis

- “Gee Whiz connectionism” is no more
- To find surprises, or emergent properties, you have to analyze internal properties first!
- We focused on the relation between connection weights and note names
- We found a set of equivalence classes similar to the ‘circle of fifths’, but based on other intervals between notes

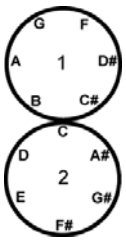


Circles Of Major 3rds



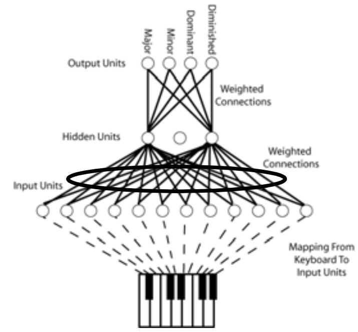
- One can create four different circles of major 3rds
- Each circle has three notes
- As you move from one note in the circle to the next, you cover an interval of a major 3rd (4 semitones)

Circles Of Major 2nds

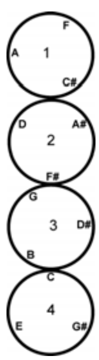


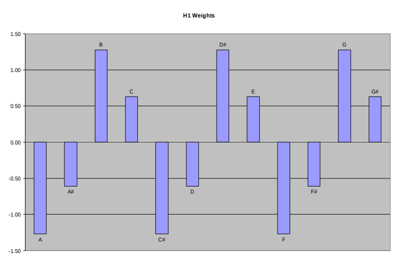
- One can create two different circles of major 2nds
- Each circle has six notes
- As you move from one note in the circle to the next, you cover an interval of a major 2nd (2 semitones)

Examining First Layer Connections

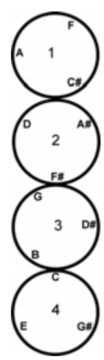


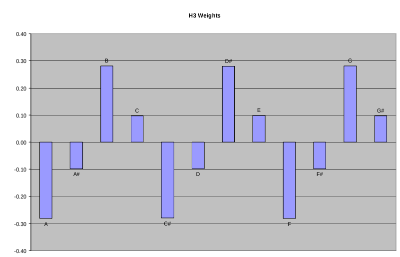
H1 Weights And Circles Of Major 3rds



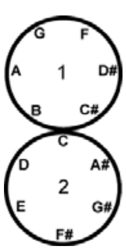


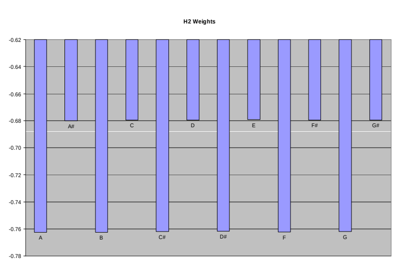
H3 Weights And Circles Of Major 3rds



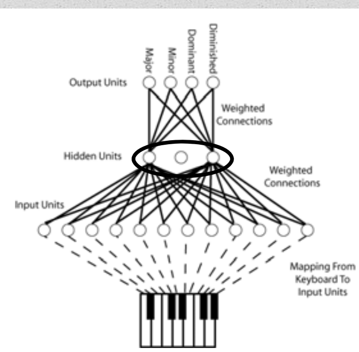


H1 Weights And Circles Of Major 2nds

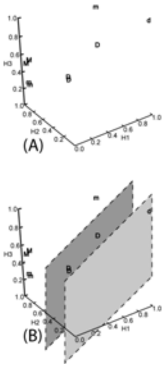




Examining Second Layer Processing



Carving Hidden Unit Space



- The chords are arranged in a 3D hidden unit space
- Output value units “carve” two parallel planes through this space
- Each unit can carve the space to separate one chord type from all others

Implications

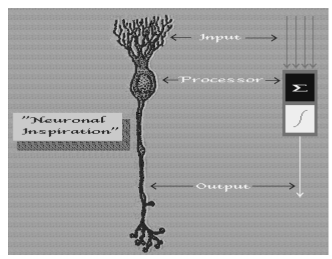
- **Our network outperformed earlier networks of Laden and Keefe**
- **Interpretation of the network revealed an unusual set of equivalence classes of notes**
- **Results in a new understanding of musical regularities, and makes some predictions that can be explored by studying human listeners**

Bunny Laden

Douglas Keefe

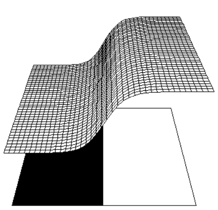
Value Unit Architecture: Local interpretations associated with local activity in hidden units

Three Processing Steps



1. Compute incoming signal using the net input function
2. Compute internal activity using activation function
3. Send an output signal using the output function

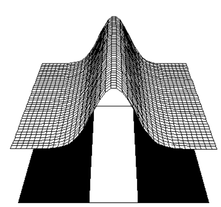
Integration Device



David Rumelhart

Geoff Hinton

Value Unit

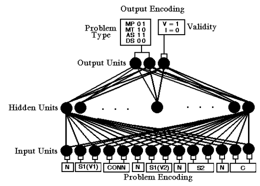


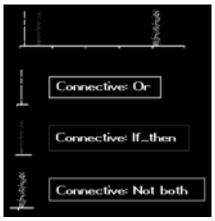
•We have been exploring an alternative activation function

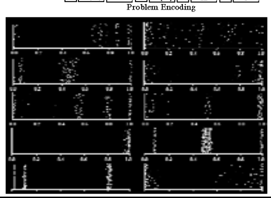
•It has many properties that assist in discovering network algorithms

Dawson, Medler & Berkeley (1997)

- Network trained to classify logic problems
- Bands associated with highly local and interpretable features
- Band features used to identify 7 "rules" in network, including 5 of classical logic



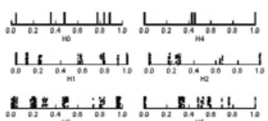


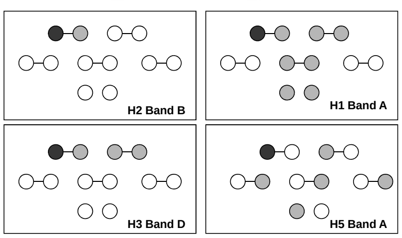


Value Unit Architecture: Interpretations distributed over ensembles of hidden unit activities

Dawson & Piercey (2001)

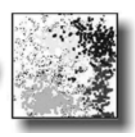
- Hinton's kinship problem
- "Who is James' father?" "Andrew"
- 6 families, 52 queries per family, 312 patterns
- 21 inputs, 6 hidden, 9 output
- Local bands uninterpretable
- Intersection of bands results in clean coarse coding interpretation

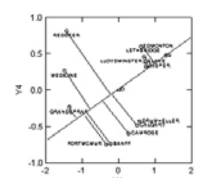




Dawson, Boechler & Valsangkar-Smyth (2000)

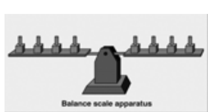
- Network trained to rate distances between Alberta cities
- Is there an internal spatial representation?
- Hidden units analogous to place cells in rat hippocampus
- Connection weights metric – encode projected distances

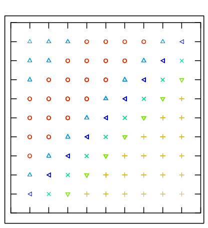




Dawson & Zimmerman (2003)

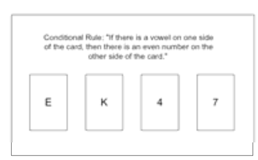
- Network trained to make ideal responses to Piaget's balance scale task
- Four hidden units use course coding to determine whether balance scale will tip left, tip right, or stay balanced
- Interpretation of network led to new additive rule for defining behaviour of balance scale
- Interpretation of network led to a new classification of problems based on a novel 2D pattern space

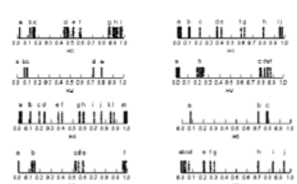


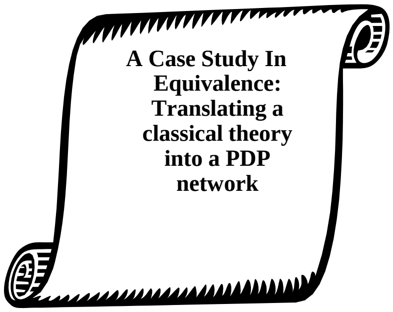


Leighton & Dawson (2001)

- Series of networks trained to give different kinds of responses to Wason Card Selection Task
- All hidden units produce bands
- Bands support an inductive set of rules for solving this task, instead of a more traditional deductive theory
- Interpretations also were used to assess difficulty of different kinds of responses







Theory Translation

- If two theories are really qualitatively different, then you can't translate one into the other
- Is this true for symbolic and connectionist theories?

Thomas S. Kuhn

The Mushroom Problem

- Problem: determine whether a mushroom is poisonous or not
- Consider 8124 different mushrooms
- Each mushroom is described using values on 21 different features

Theory 1 (Classical)

- What is the mushroom's odor?
  - If almond or anise then edible
  - If another definite odor then poisonous
  - If no odor then go to next step
- What is the spore print color?
  - If white then go to next step
  - If green or purple then poisonous
  - If some other color then edible
- What is the gill size of the mushroom?
  - If broad then edible
  - If narrow then go to next step
- Examine the stalk surface above the mushroom's ring
  - If fibrous then edible
  - If silky or scaly then poisonous
  - If smooth then go to next step
- Does the mushroom have bruises?
  - If not, then edible
  - If it does, then poisonous

Deadly

Tasty!

Decision Tree To Production System

<u>Odor</u>: Almond or Anise

<u>Odor</u>: Creosote or Fishy or Foul or Musty or Pungent or Spicy

<u>Odor:</u> None
<u>Spore Print:</u> Black or Brown or Buff or Chocolate or Orange or Yellow

<u>Odor:</u> None
<u>Spore Print:</u> Green or Purple

<u>Odor:</u> None
<u>Spore Print:</u> White
<u>Gill Size</u>: Broad

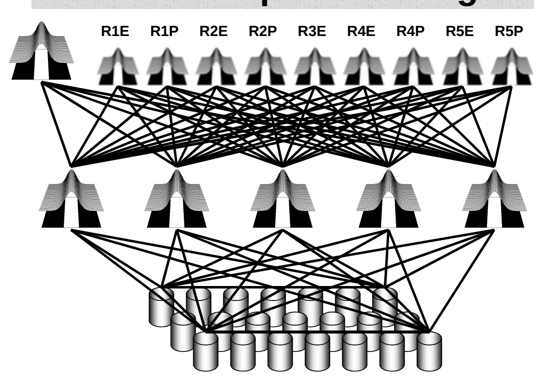
<u>Odor:</u> None
<u>Spore Print:</u> White
<u>Gill Size</u>: Narrow
<u>Stalk Surface Above Ring</u>: Fibrous

<u>Odor:</u> None
<u>Spore Print:</u> White
<u>Gill Size</u>: Narrow
<u>Stalk Surface Above Ring</u>: Silky or Scaly

<u>Odor:</u> None
<u>Spore Print:</u> White
<u>Gill Size</u>: Narrow
<u>Stalk Surface Above Ring</u>: Smooth
<u>Bruises</u>: No

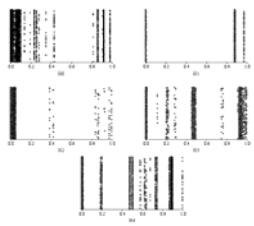
<u>Odor:</u> None
<u>Spore Print:</u> White
<u>Gill Size</u>: Narrow
<u>Stalk Surface Above Ring</u>: Smooth
<u>Bruises</u>: Yes

Extra Output Learning



Hidden Unit Banding

- The hidden units of this network demonstrate a high degree of banding
- Can be locally interpreted
- Distributions over hidden units can also be interpreted

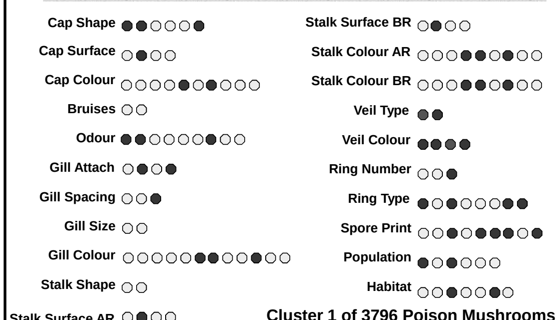


Theory 2 (Connectionist)

CLUSTER	POISONOUS	EDIBLE
1	3796	0
2	0	704
3	0	96
4	0	528
5	40	0
6	72	0
7	0	12
8	0	12
9	0	2832
10	8	0
11	0	12
12	0	12

Each cluster is "pure" in terms of network's main response

Definite Features



Cluster 1 of 3796 Poison Mushrooms

Each cluster is laden with definite features

Clusters Map Onto Productions!

<u>Odor</u>: Almond or Anise — C1

<u>Odor</u>: Creosote or Fishy or Foul or Musty or Pungent or Spicy — C2, C3

<u>Odor:</u> None — C9
<u>Spore Print:</u> Black or Brown or Buff or Chocolate or Orange or Yellow

<u>Odor:</u> None — C6
<u>Spore Print:</u> Green or Purple

<u>Odor:</u> None — C4
<u>Spore Print:</u> White
<u>Gill Size</u>: Broad

<u>Odor:</u> None — C8, C12
<u>Spore Print:</u> White
<u>Gill Size</u>: Narrow
<u>Stalk Surface Above Ring</u>: Fibrous

<u>Odor:</u> None — C5
<u>Spore Print:</u> White
<u>Gill Size</u>: Narrow
<u>Stalk Surface Above Ring</u>: Silky or Scaly

<u>Odor:</u> None — C10
<u>Spore Print:</u> White
<u>Gill Size</u>: Narrow
<u>Stalk Surface Above Ring</u>: Smooth
<u>Bruises</u>: No

<u>Odor:</u> None — C7, C11
<u>Spore Print:</u> White
<u>Gill Size</u>: Narrow
<u>Stalk Surface Above Ring</u>: Smooth
<u>Bruises</u>: Yes

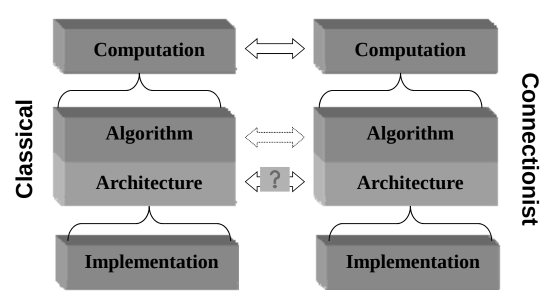
Implication

- We can translate a symbolic theory into a PDP network
- Perhaps PDP is <u>not</u> a "paradigm shift"
- Classical versus PDP debate requires more sophistication

Walter Schneider

What Kind Of Sophistication?

- Do other algorithmic equivalences exist?
- Do they map onto the same architecture?



Synthetic Psychology

- “The study of Connectionist machines had led to a number of striking and unanticipated findings; it’s surprising how much computing can be done with a uniform network of simple interconnected elements” (Fodor & Pylyshyn, 1988, p. 3)

- **What kind of computing do they do?**
- **Build networks, then interpret them and their algorithms**
- **It might be more related to classical computing than is often claimed**