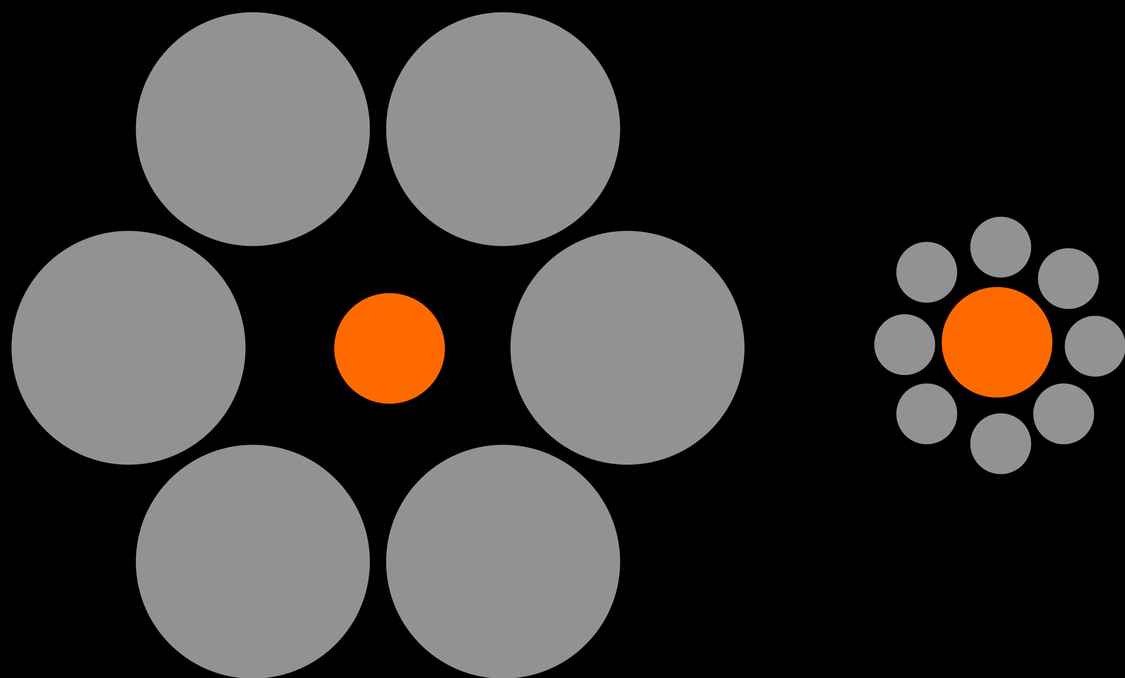
Constraint Satisfaction, Attractor Networks and Perception

Perceptual Illusions

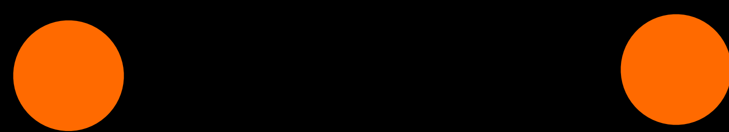
Context influences perception



Ebbinghaus-Tichener Illusion (1901)

Perceptual Illusions

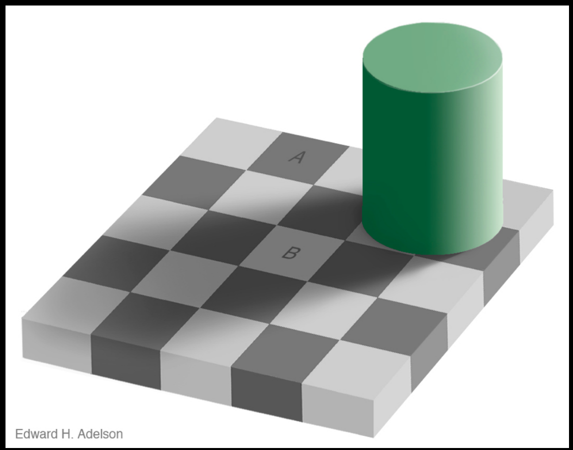
Context influences perception

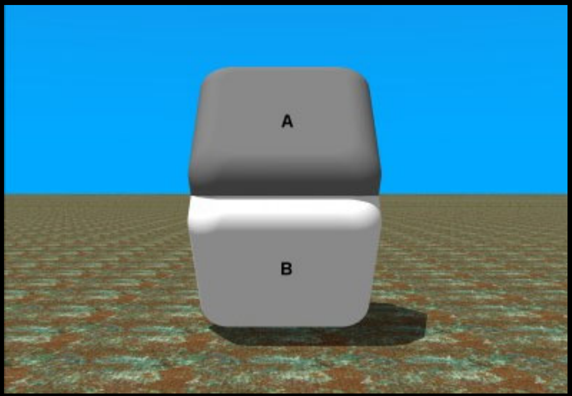


Ebbinghaus-Tichener Illusion (1901)

Perceptual Illusions

Context influences perception

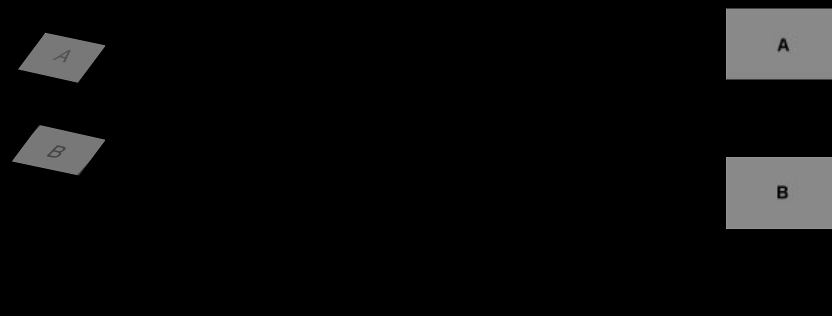




Edward Adelson (1995)

Perceptual Illusions

Context influences perception



Edward Adelson (1995)

Top-down Effects

Mask Illusion - Richard Gregory (1970)

Top-down Effects

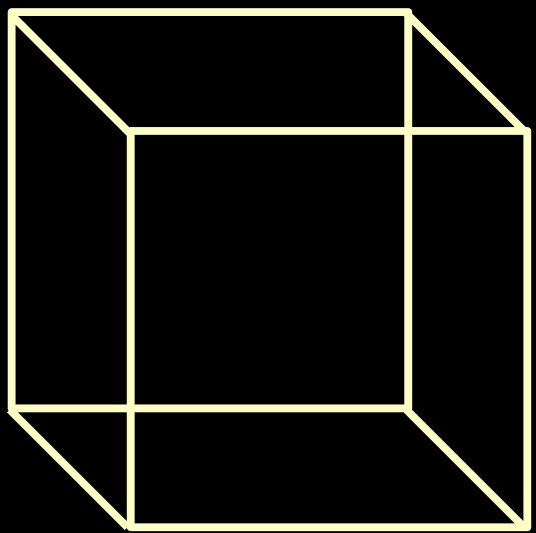
Mask Illusion - Richard Gregory (1970)

Top-down Effects

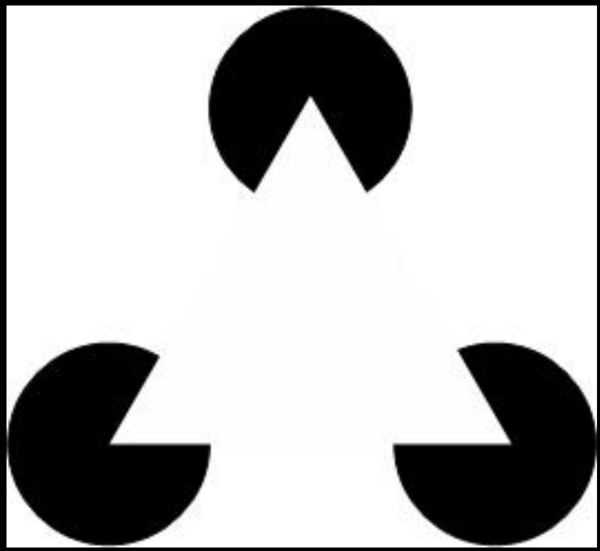
Instruction can also be a source of context

Gestalt Figures

The parts interact to determine the whole



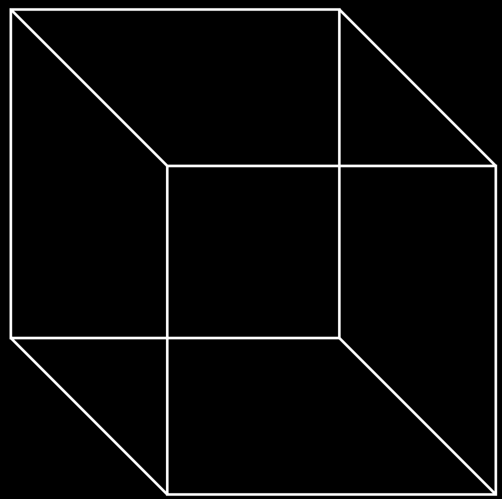
Necker Cube *(1832)*



Kanizsa Triangle *(1976)*

Perceptual Bistability

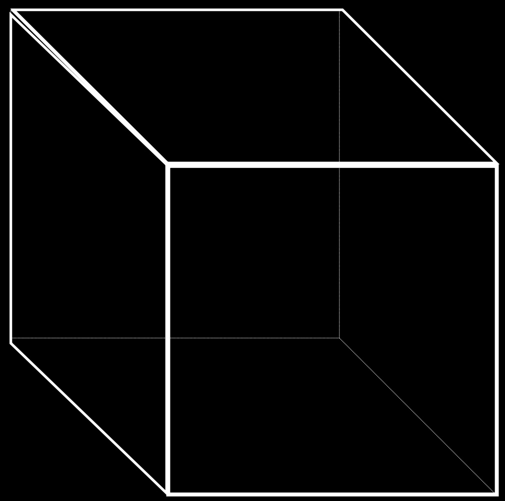
- **Instantly perceive a coherent figure** *(more or less)*
- **Two different interpretations**
- **Can't perceive both at once**
- **Alternate between perceptions:** ***bistablility…***



The Necker Cube

Perceptual Bistability

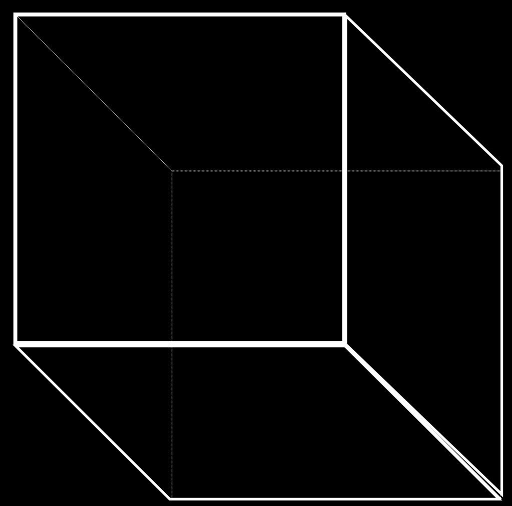
- Instantly perceive coherent figure (more or less)
- Two different interpretations
- Can't perceive both at once
- Alternate between perceptions: *from above*



The Necker Cube

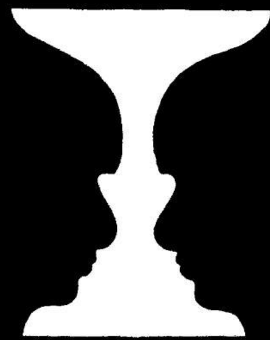
Perceptual Bistability

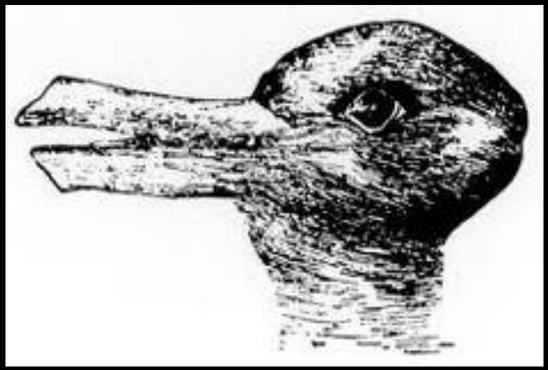
- Instantly perceive coherent figure (more or less)
- Two different interpretations
- Can’t perceive both at once
- Alternate between perceptions: *from below*



The Necker Cube

Other Examples





Gestalt Perception

Gestalt Perception

- **Principles that guide perception of the "whole"**
  - Similarity
  - Contiguity
  - Continuity
  - Closure
  - Symmetry

Gestalt Perception

- **Principles that guide perception of the "whole"**
  - –Similarity
  - –Contiguity
  - –Continuity
  - –Closure
  - –Symmetry
  - **–Emergence**

Gestalt Perception

- **Principles that guide perception of the "whole"**
  - Similarity
  - Contiguity
  - Continuity
  - Closure
  - Symmetry
  - **Emergence**

- **Nice heuristic description, but how do these work?**
  - How do we assess images along all of these dimensions at once, seemingly instantaneously?

    (remember the 100 step rule)

Constraint Satisfaction

Constraint Satisfaction

- These problems can be recast more generally as constraint satisfaction problems:

Constraint Satisfaction

– Simultaneously satisfy many interdependent relationships, or "constraints"
(e.g., matches between sensory cues, or sensory input and memory representations)

Constraint Satisfaction

– **Simultaneously satisfy many interdependent relationships, or “constraints”**
(e.g., matches between sensory cues, or sensory input and memory representations)

Constraint Satisfaction

– There may be no perfect solution, so…

Constraint Satisfaction

– Look for the best "fit" — one that satisfies as many of the constraints as possible

Constraint Satisfaction

– Some constraints may be more common or important than others

Constraint Satisfaction

- Connectionist models lend themselves naturally to the solution of such problems...

Constraint Satisfaction

Constraint Satisfaction

- Hypotheses = unit activity values

Constraint Satisfaction

- Hypotheses = unit activity values
- Constraints = connections between units

Constraint Satisfaction

- Hypotheses = unit activity values
- Constraints = connections between units
- Importance of constraint = weight of connection

Constraint Satisfaction

- Hypotheses = unit activity values
- Constraints = connections between units
- Importance of constraint = weight of connection
- *A priori* probability of truth of a hypothesis = biases

Constraint Satisfaction

- Hypotheses = unit activity values
- Constraints = connections between units
- Importance of constraint = weight of connection
- *A priori* probability of truth of a hypothesis = biases
- Evidence for a given hypothesis = external input

Constraint Satisfaction

- Hypotheses = unit activity values
- Constraints = connections between units
- Importance of constraint = weight of connection
- *A priori* probability of truth of a hypothesis = biases
- Evidence for a given hypothesis = external input
- Satisficing = settling process

Constraint Satisfaction

- Hypotheses = unit activity values
- Constraints = connections between units
- Importance of constraint = weight of connection
- *A priori* probability of truth of a hypothesis = biases
- Evidence for a given hypothesis = external input
- Satisficing = settling process
- Success = goodness of fit

Constraint Satisfaction

- Hypotheses = unit activity values
- Constraints = connections between units
- Importance of constraint = weight of connection
- *A priori* probability of truth of a hypothesis = biases
- Evidence for a given hypothesis = external input
- Satisficing = settling process
- Success = goodness of fit

$$P(Hypothesis|Data) = \frac{P(Data|Hypothesis) \cdot P(Hypothesis)}{P(Hypothesis,Data)}$$

Constraint Satisfaction

- Hypotheses = unit activity values
- Constraints = connections between units
- Importance of constraint = weight of connection
- *A priori* probability of truth of a hypothesis = biases
- Evidence for a given hypothesis = external input
- Satisficing = settling process
- Success = goodness of fit

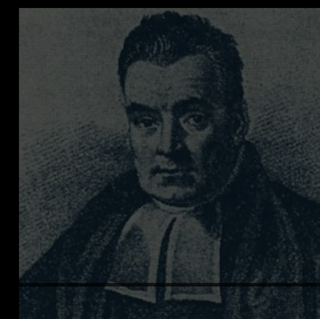
prior

likelihood

posterior

$$P(Hypothesis|Data) = \frac{P(Data|Hypothesis) \cdot P(Hypothesis)}{P(Hypothesis, Data)}$$

Constraint Satisfaction



- Hypotheses = unit activity values
- Constraints = connections between units
- Importance of constraint = weight of connection
- *A priori* probability of truth of a hypothesis = biases
- Evidence for a given hypothesis = external input
- Satisficing = settling process
- Success = goodness of fit

prior

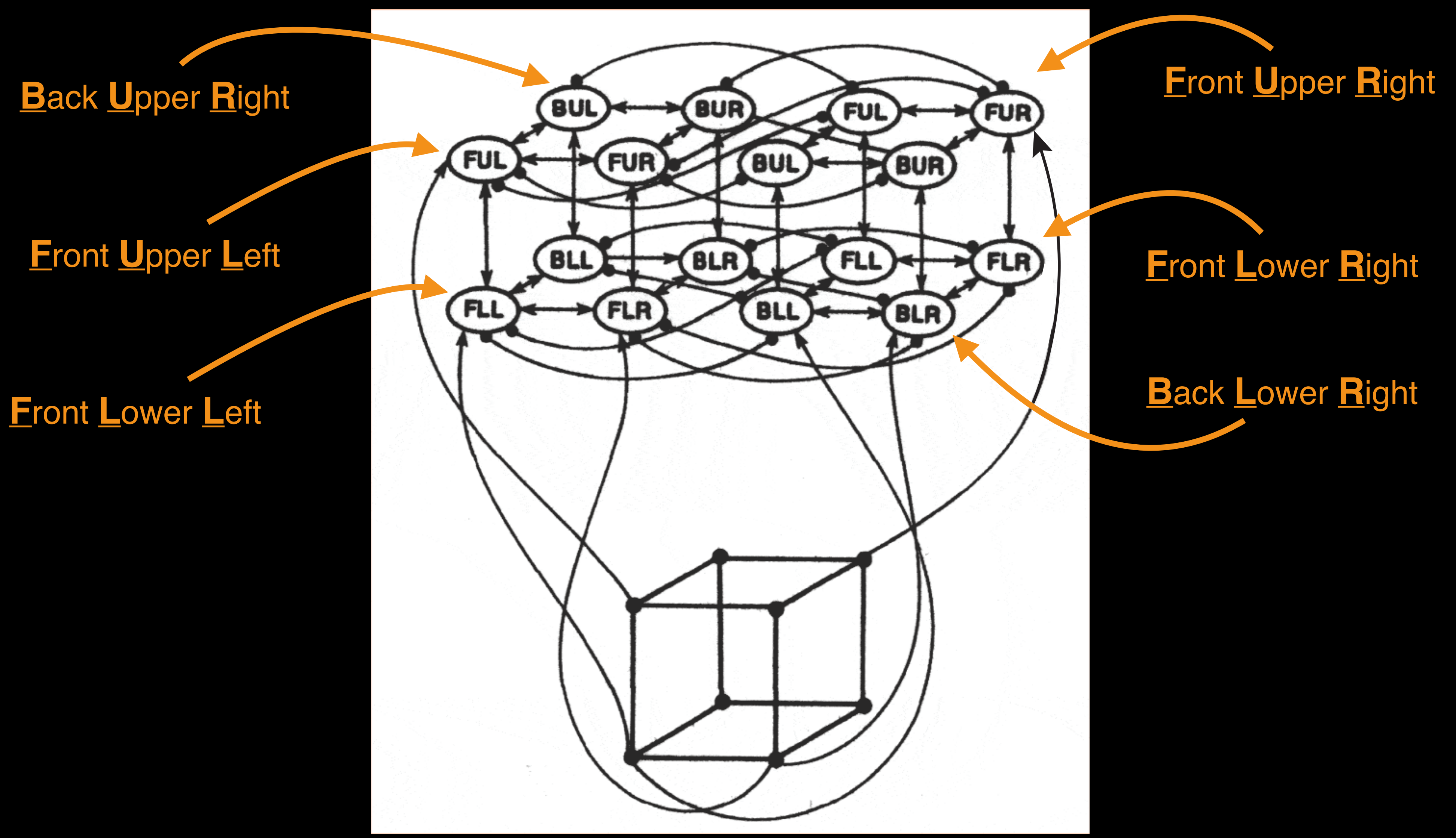
likelihood

posterior

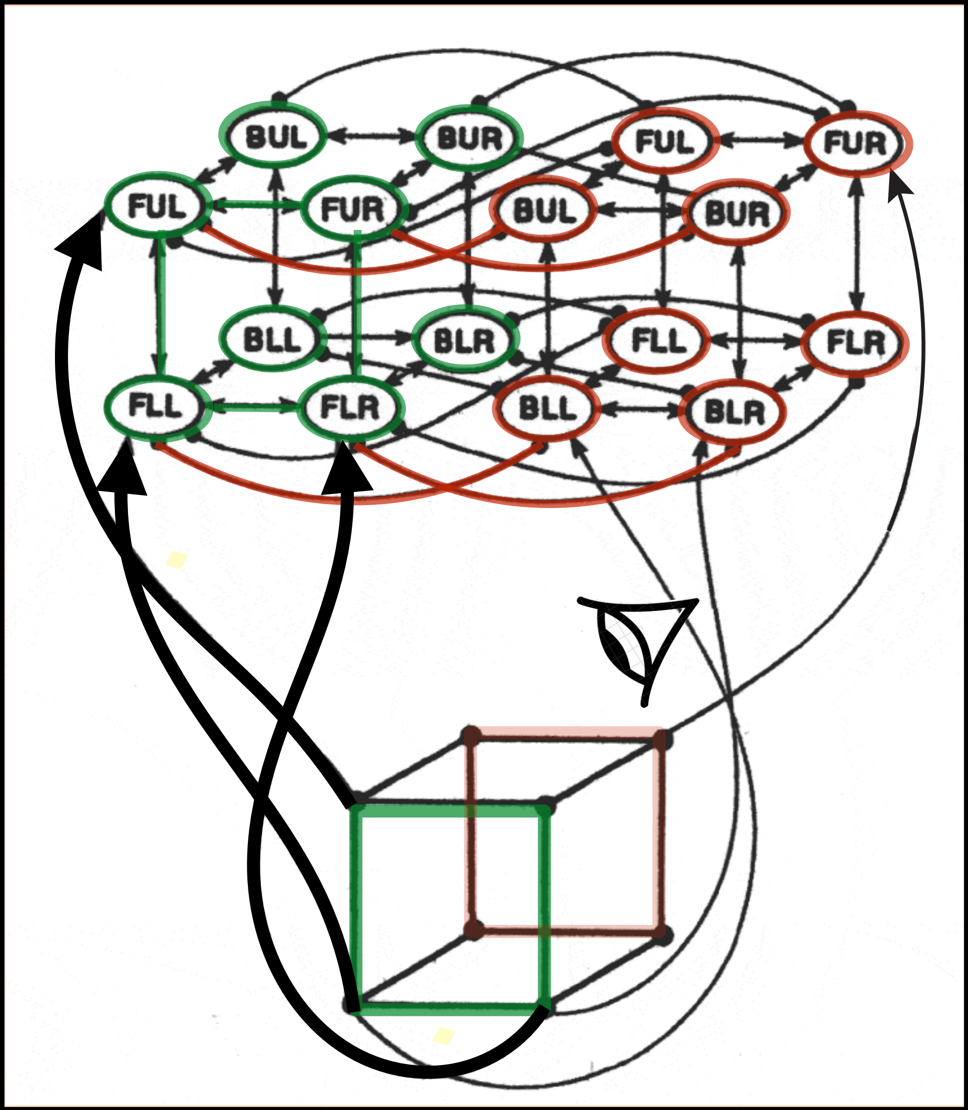
$$P(Hypothesis|Data) = \frac{P(Data|Hypothesis) \cdot P(Hypothesis)}{P(Hypothesis,Data)}$$

- How can this be applied to psychological phenomena?

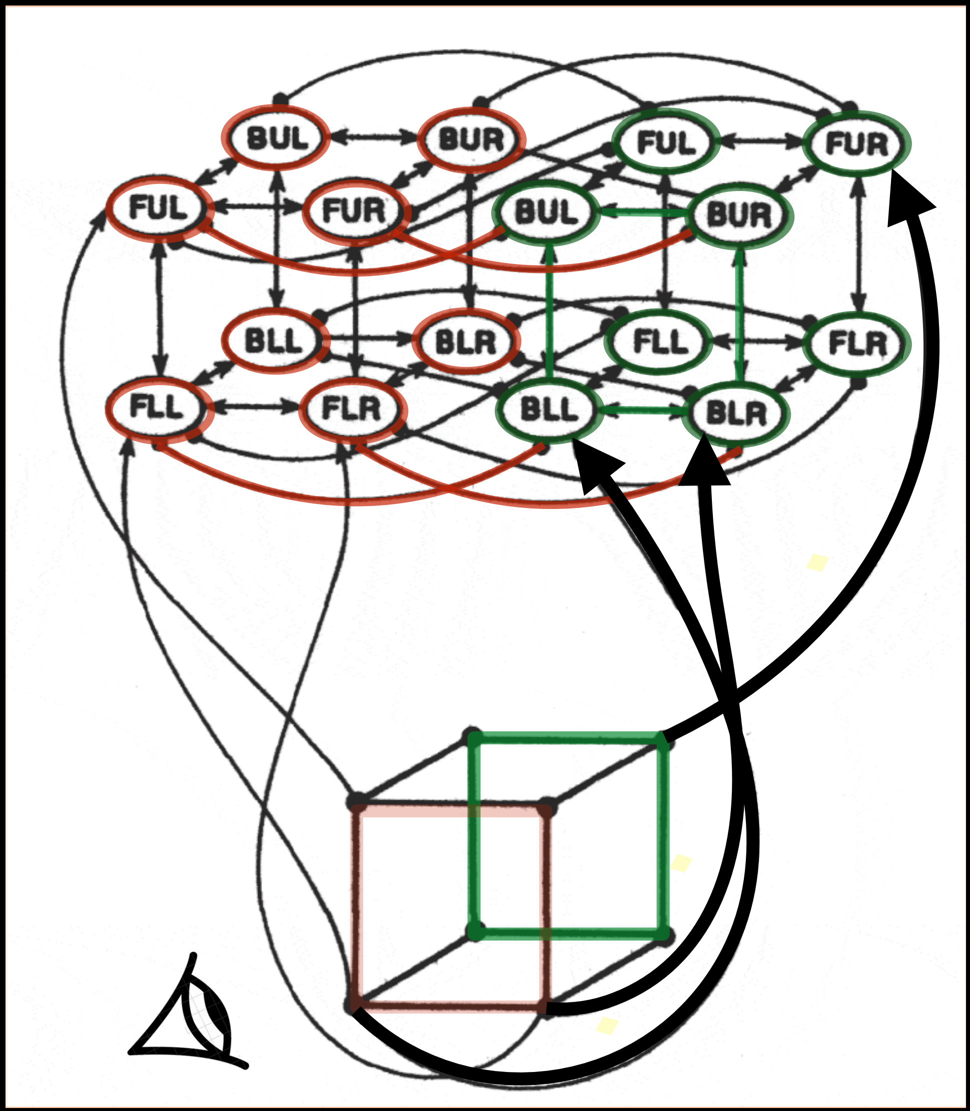
Constraint Satisfaction Network



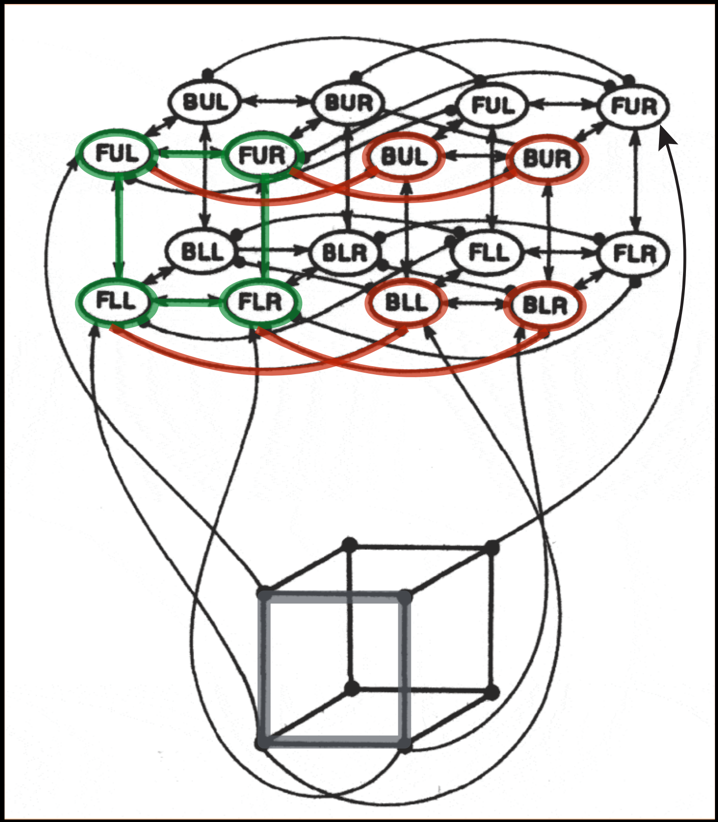
Constraint Satisfaction Network



Constraint Satisfaction Network

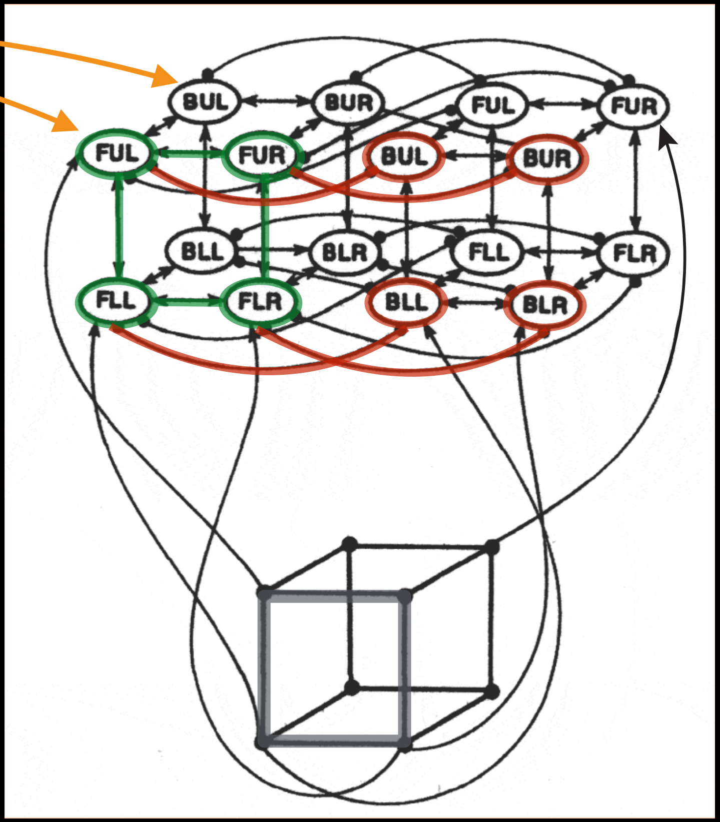


Constraint Satisfaction Network



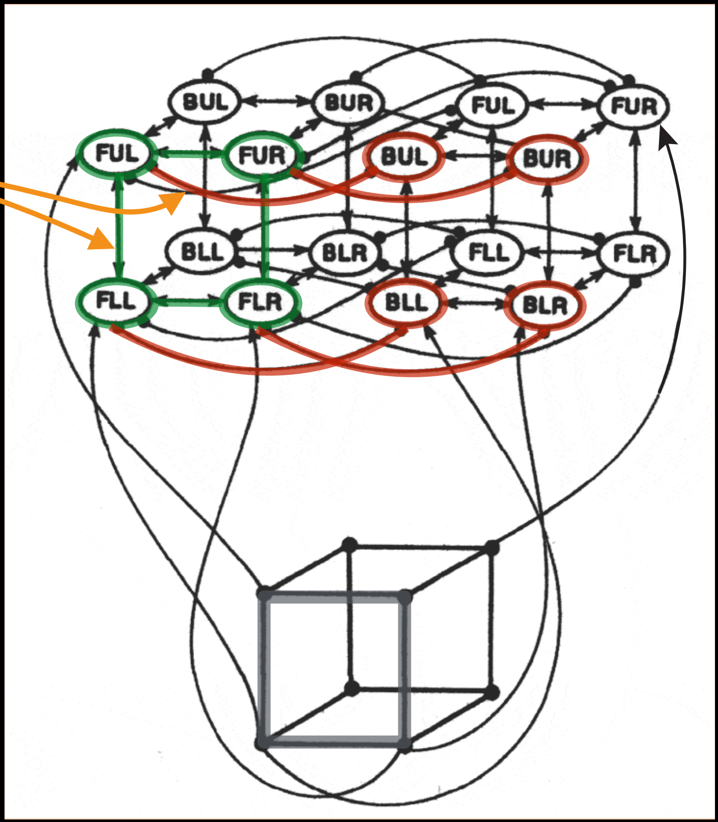
Constraint Satisfaction Network

- ***Hypotheses*** = unit activity values



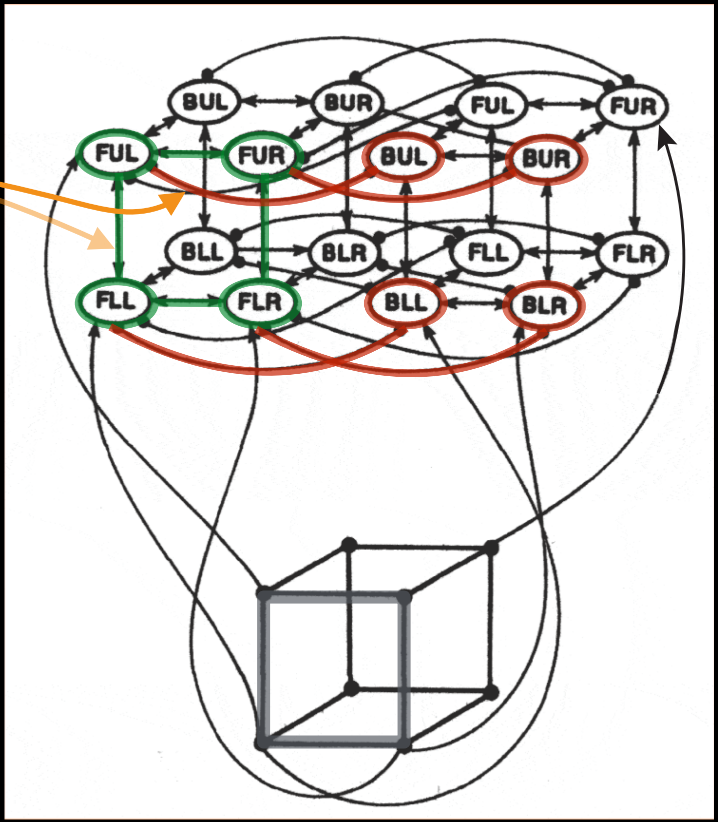
Constraint Satisfaction Network

- ***Constraints*** = connections between units



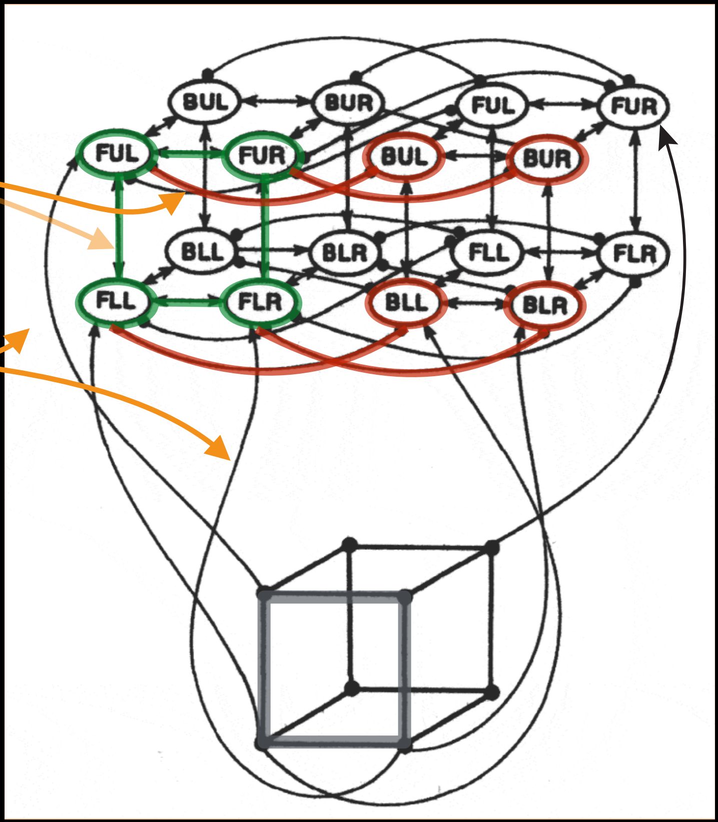
Constraint Satisfaction Network

- ***Importance of constraint*** = weight of connection

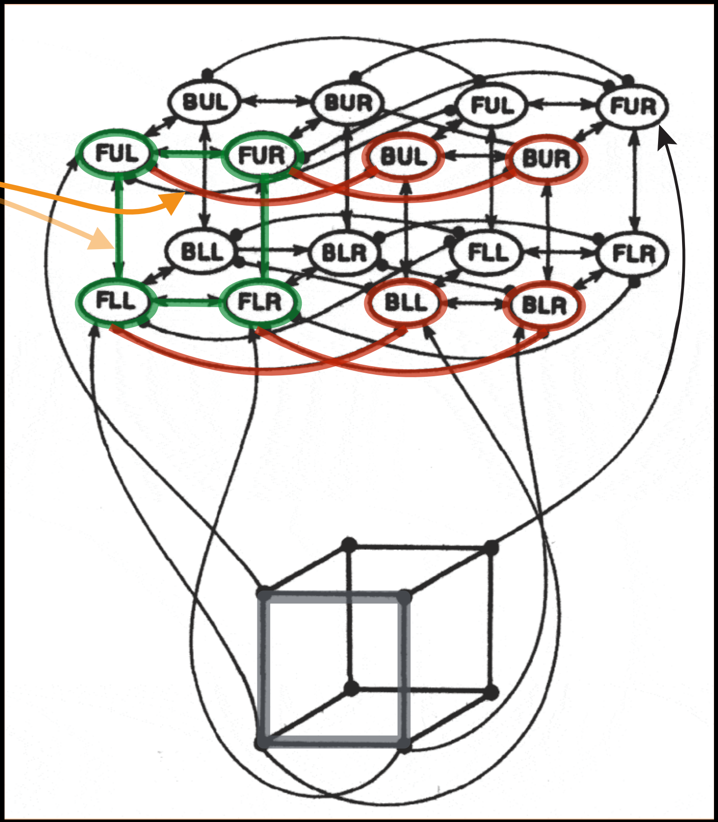


Constraint Satisfaction Network

- ***Evidence*** (for a given hypothesis) = external input

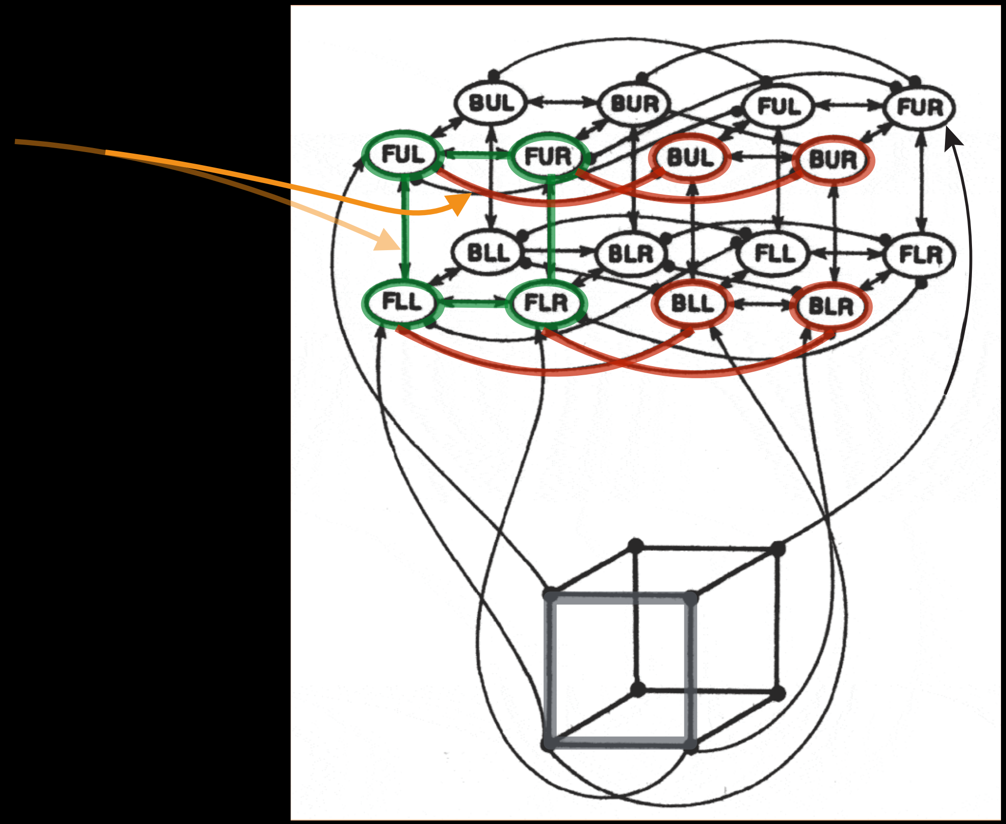


Constraint Satisfaction Network



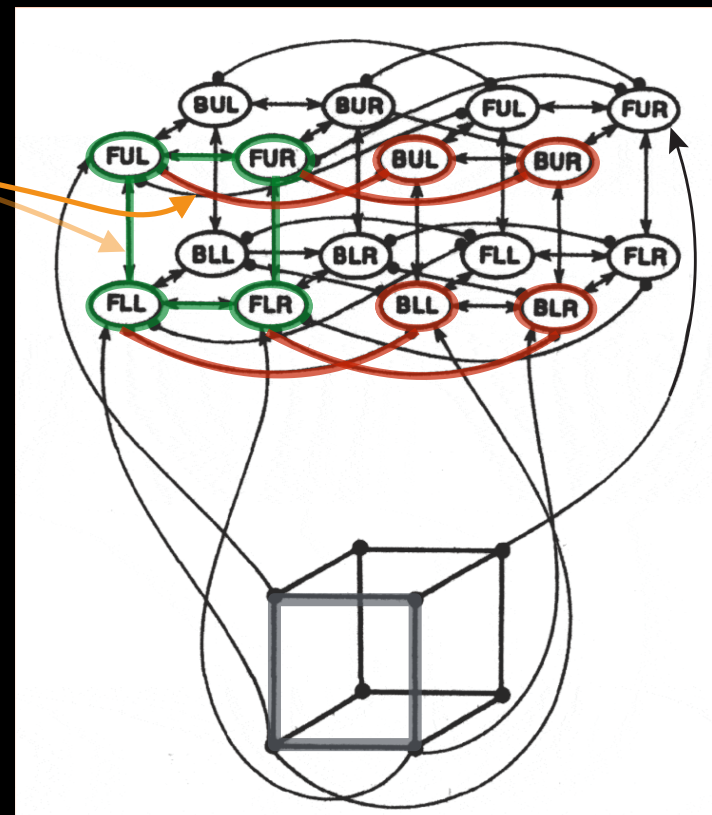
- ***A priori probability*** (for a given hypothesis) = biases

Constraint Satisfaction Network



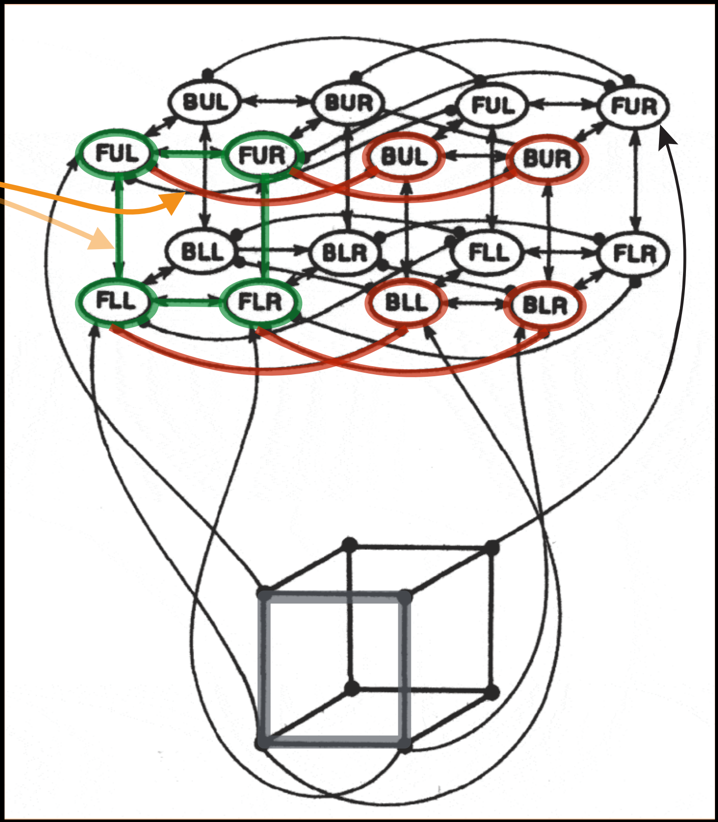
- ***Inference*** = settling process

Constraint Satisfaction Network



- ***Success*** = goodness of fit

Constraint Satisfaction Network



- ***Success*** = goodness of fit

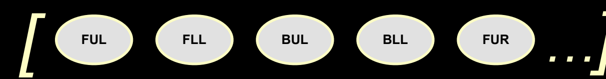
- **How can we formalize this?**

Hopfield Networks, Energy & "State Space" Dynamics

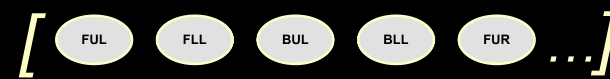
Hopfield Networks, Energy & "State Space" Dynamics

- **State: vector of unit activation values**

  (state space = range of all possible vector values)



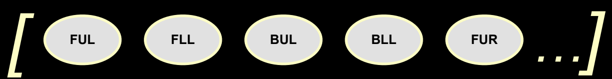
Hopfield Networks, Energy & "State Space" Dynamics



- **Energy of each state:** $E = -\dfrac{\Sigma_{ij} w_{ij} a_i a_j}{2}$

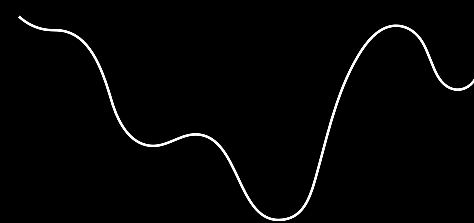
  (opposite of Goodness)

Hopfield Networks, Energy & "State Space" Dynamics

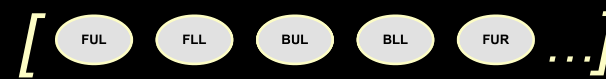


$$E = -\frac{\Sigma_{ij} w_{ij} a_i a_j}{2}$$

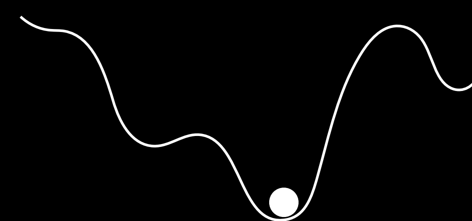
- **Energy surface:** **plot of energy for every state**



Hopfield Networks, Energy & "State Space" Dynamics

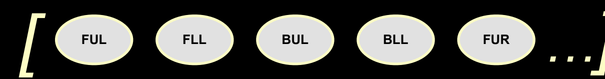


$$E = -\frac{\Sigma_{ij} w_{ij} a_i a_j}{2}$$

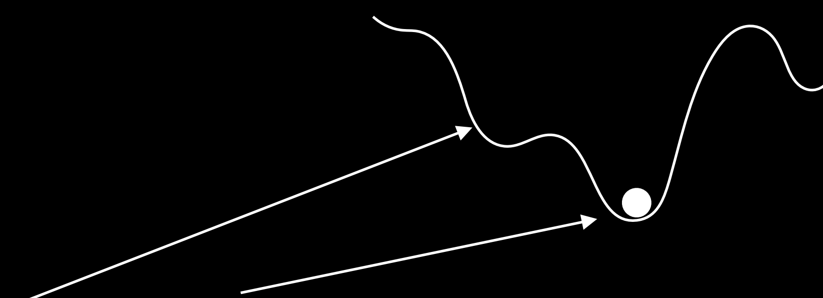


- **Dynamics:** traversal of energy surface

Hopfield Networks, Energy & "State Space" Dynamics

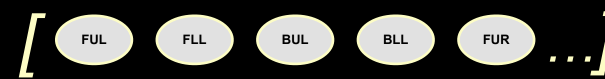


$$E = -\frac{\Sigma_{ij} w_{ij} a_i a_j}{2}$$

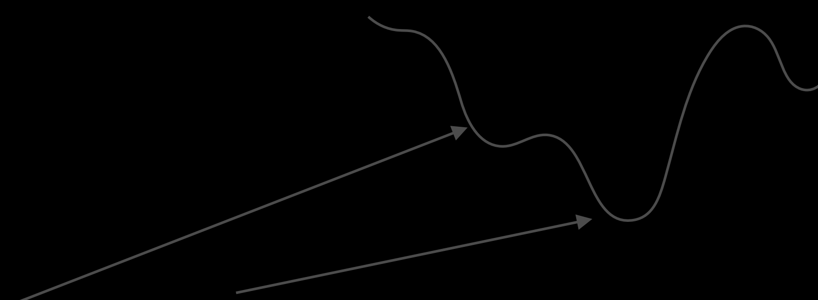


- **Minima:** points of lowest energy (local & global)

Hopfield Networks, Energy & "State Space" Dynamics

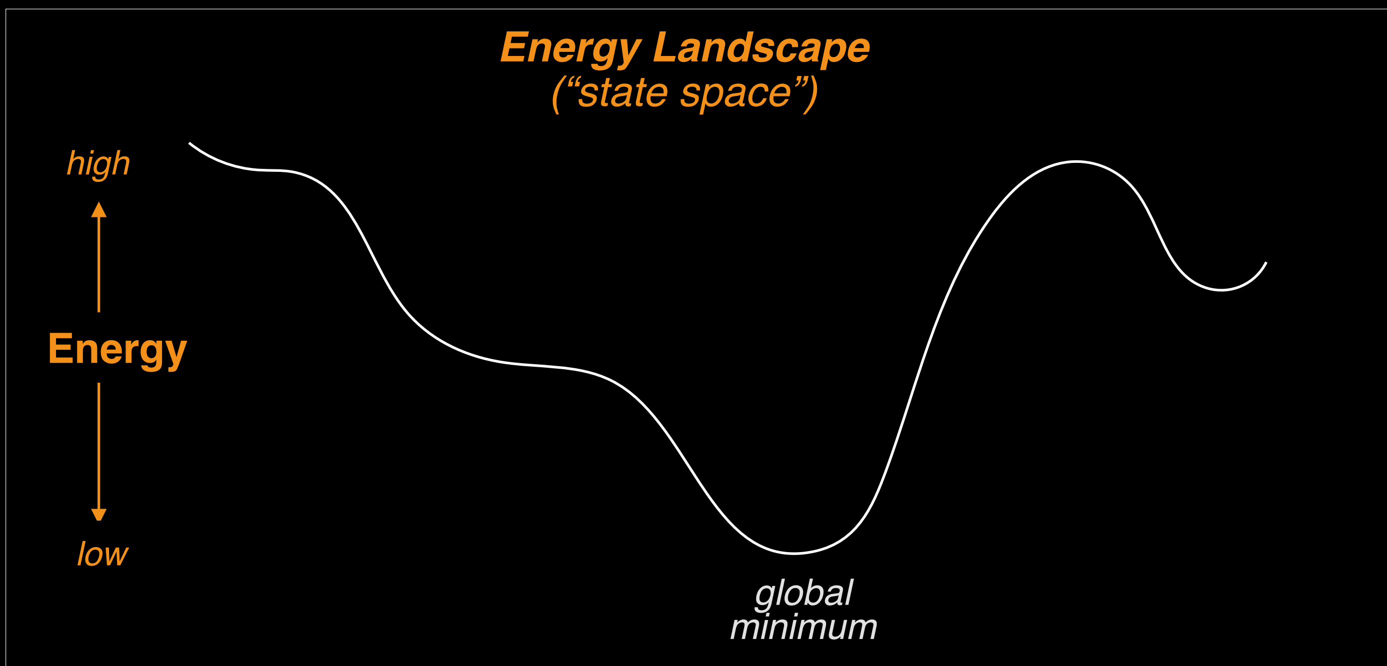


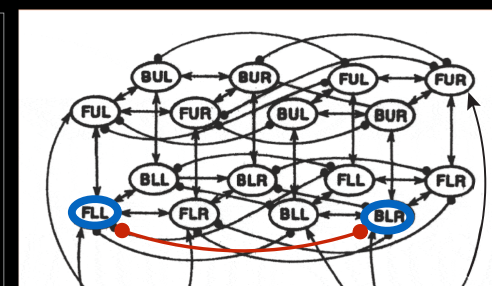
$$E = -\frac{\Sigma_{ij} w_{ij} a_i a_j}{2}$$

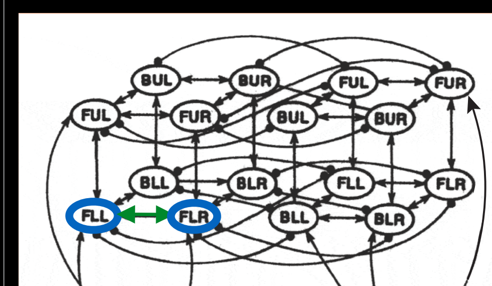


- Under proper assumptions, can prove that system will flow down hill

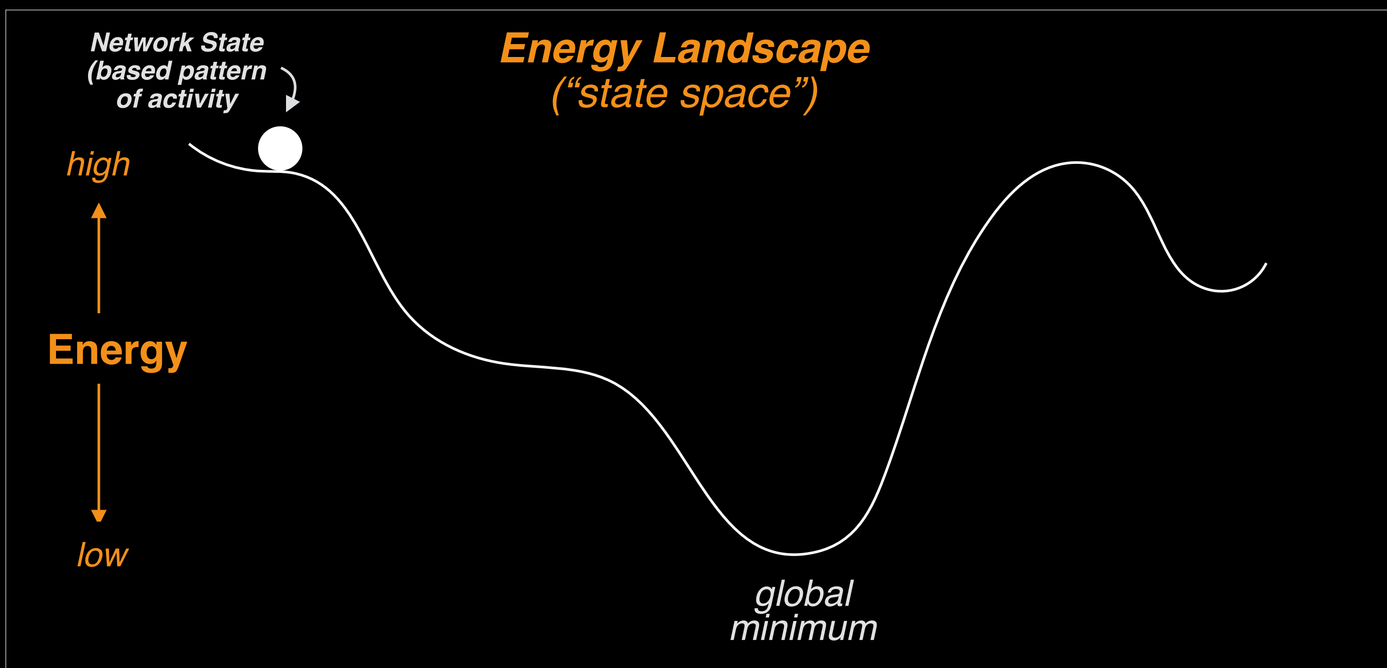
Engergy Landscape

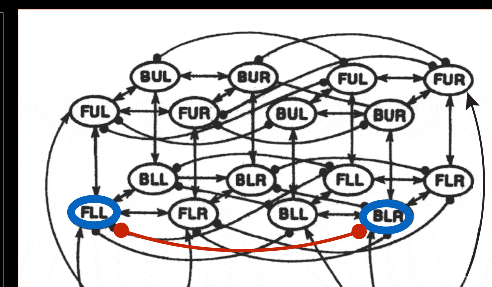


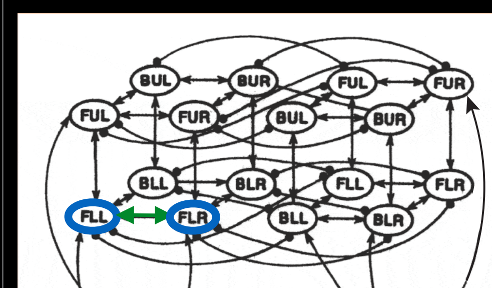




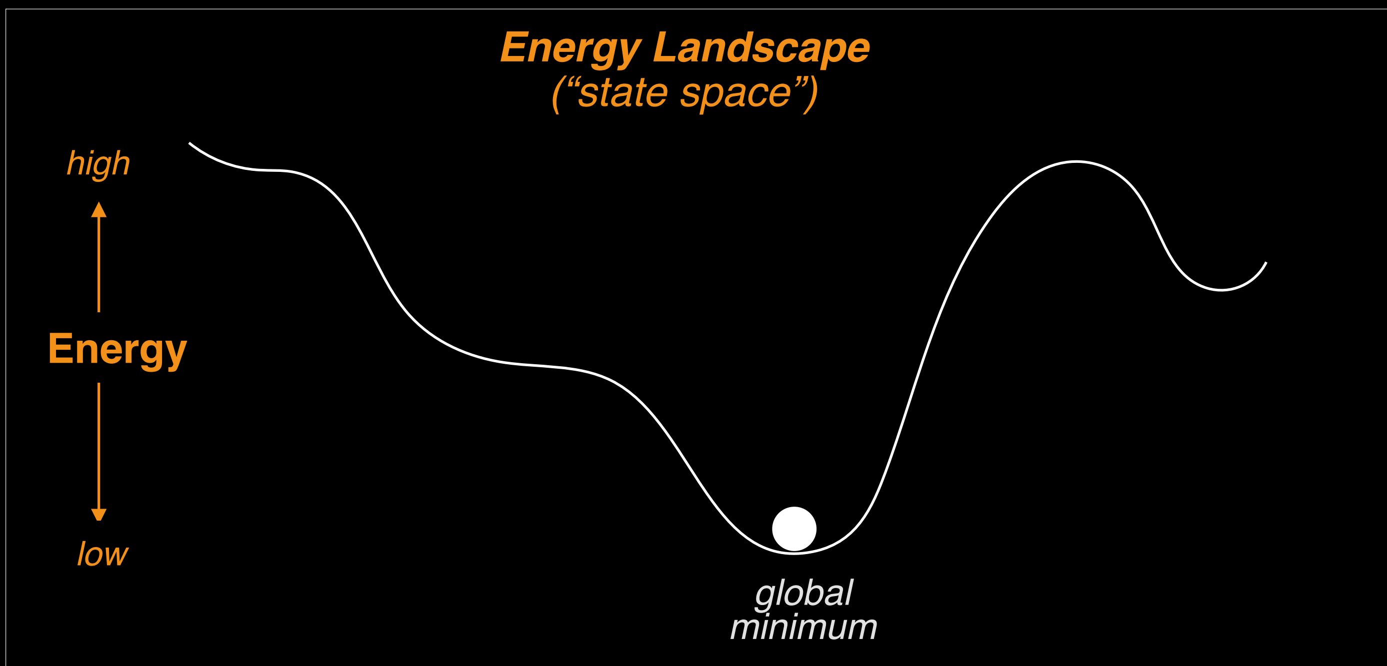
Engergy Landscape

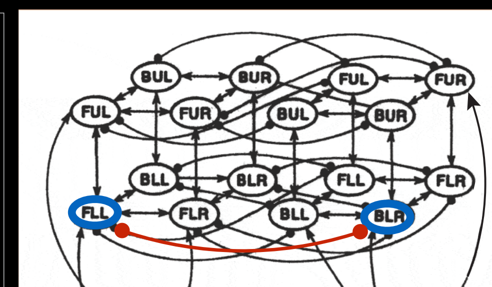


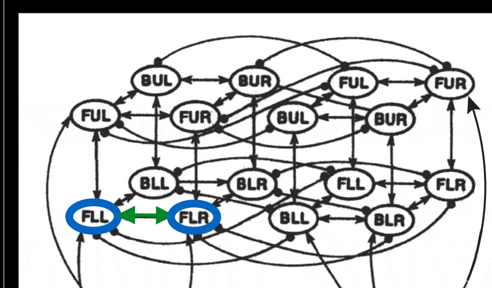




Engergy Landscape



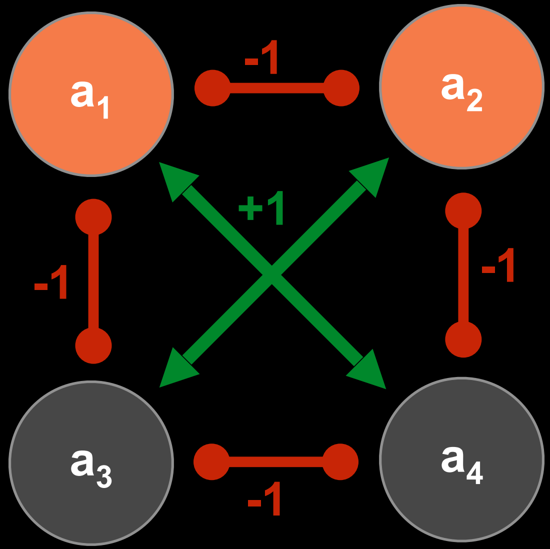




An Example

$$E = -\frac{\Sigma_{ij} w_{ij} a_i a_j}{2}$$

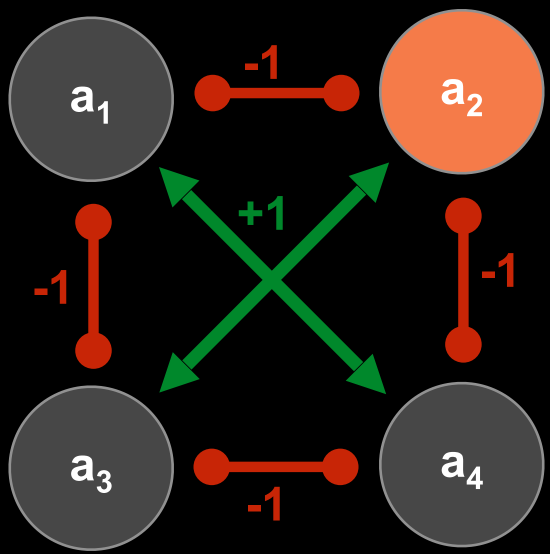
Update	a_1	a_2	a_3	a_4	Energy
Initial State	1	1	0	0	+0.5



An Example

$$E = -\frac{\Sigma_{ij} w_{ij} a_i a_j}{2}$$

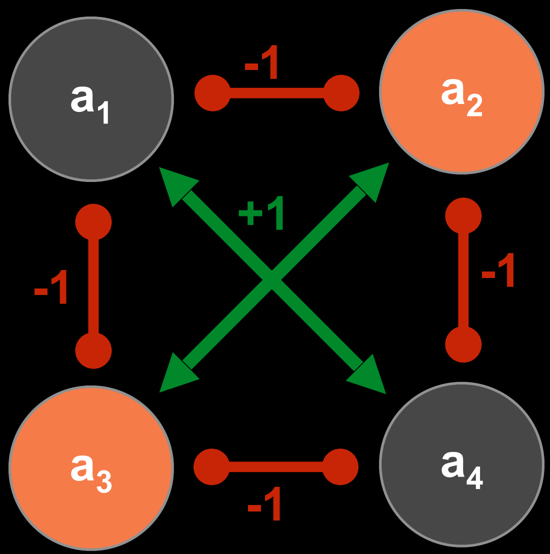
Update	a_1	a_2	a_3	a_4	Energy
a_1	0	1	0	0	0



An Example

$$E = -\frac{\Sigma_{ij} w_{ij} a_i a_j}{2}$$

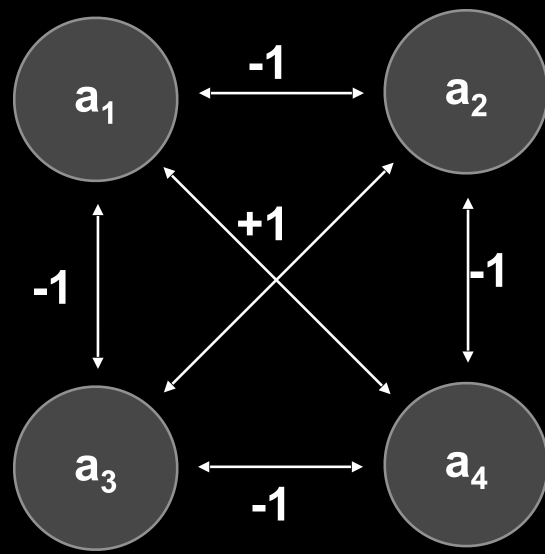
Update	a_1	a_2	a_3	a_4	Energy
	1	1	0	0	+0.5
a_1	0	1	0	0	0
a_3	0	1	1	0	-0.5



An Example

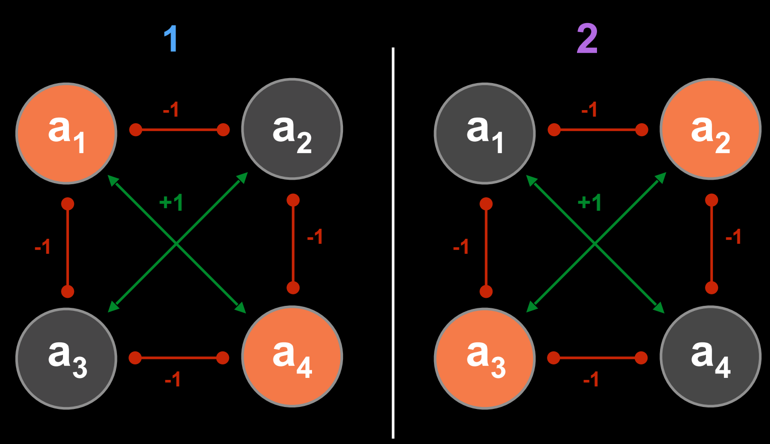
$$E = -\frac{\Sigma_{ij} w_{ij} a_i a_j}{2}$$

Update	a_1	a_2	a_3	a_4	Energy
Initial state	1	1	0	0	+0.5
a_1	0	1	0	0	0
a_2	0	1	1	0	-0.5



Two stable states — percepts *(or memories)*

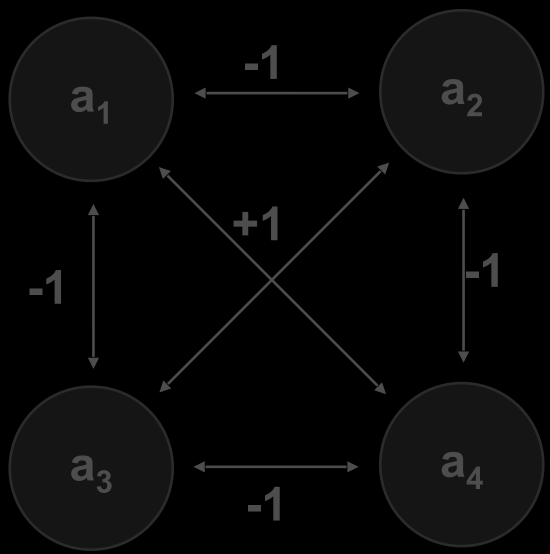
State	a_1	a_2	a_3	a_4	Energy
1	1	0	0	1	-0.5
2	0	1	1	0	-0.5



An Example

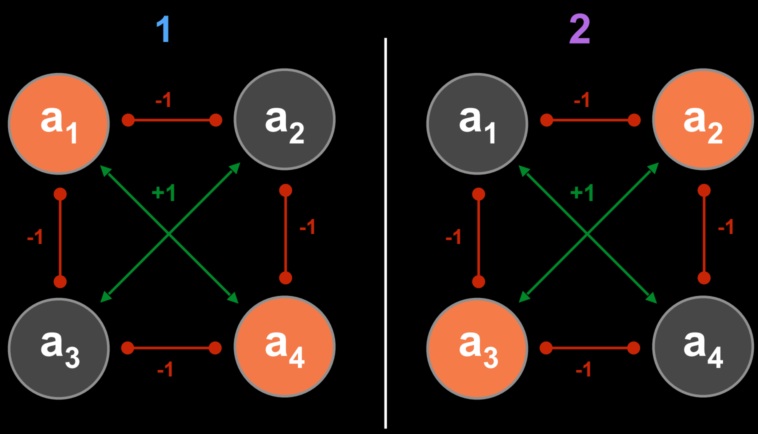
$$E = -\frac{\Sigma_{ij} w_{ij} a_i a_j}{2}$$

Update	a_1	a_2	a_3	a_4	Energy
Initial state	1	1	0	0	+0.5
a_1	0	1	0	0	0
a_2	0	1	1	0	-0.5



Two stable states — percepts *(or memories)*

State	a_1	a_2	a_3	a_4	Energy
1	1	0	0	1	-0.5
2	0	1	1	0	-0.5



Energy Landscapes

Energy Landscapes

- A surface that is the energy of the network as a function of the activity of its units

Energy Landscapes

- The state of the system is a point on this surface

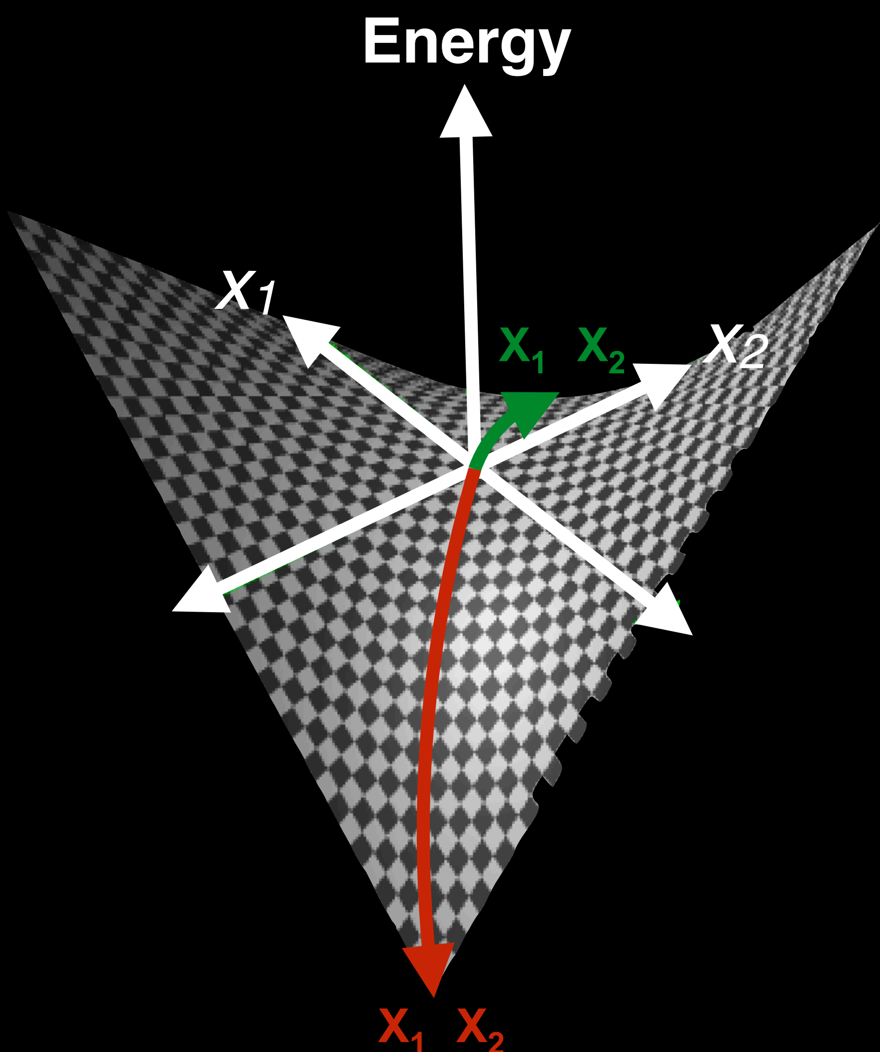
Energy Landscapes

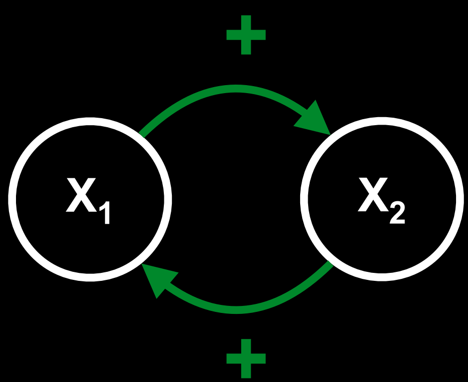
- The settling process of the network is the downhill trajectory of this network along this surface

Energy Landscapes

- Hard to visualize in high dimensions, so stick to 2.5-D…

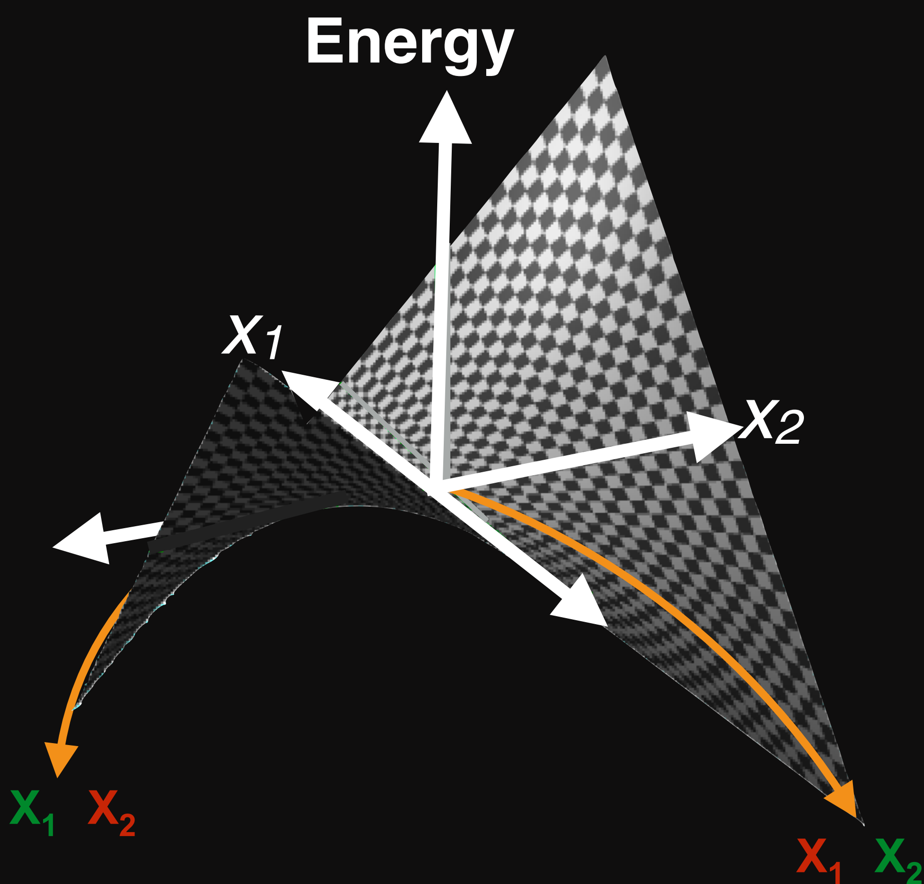
Energy Landscapes

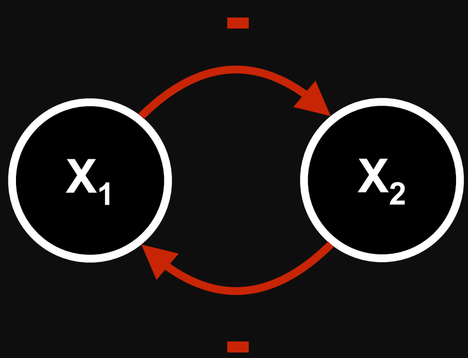




Mutually Excitatory units

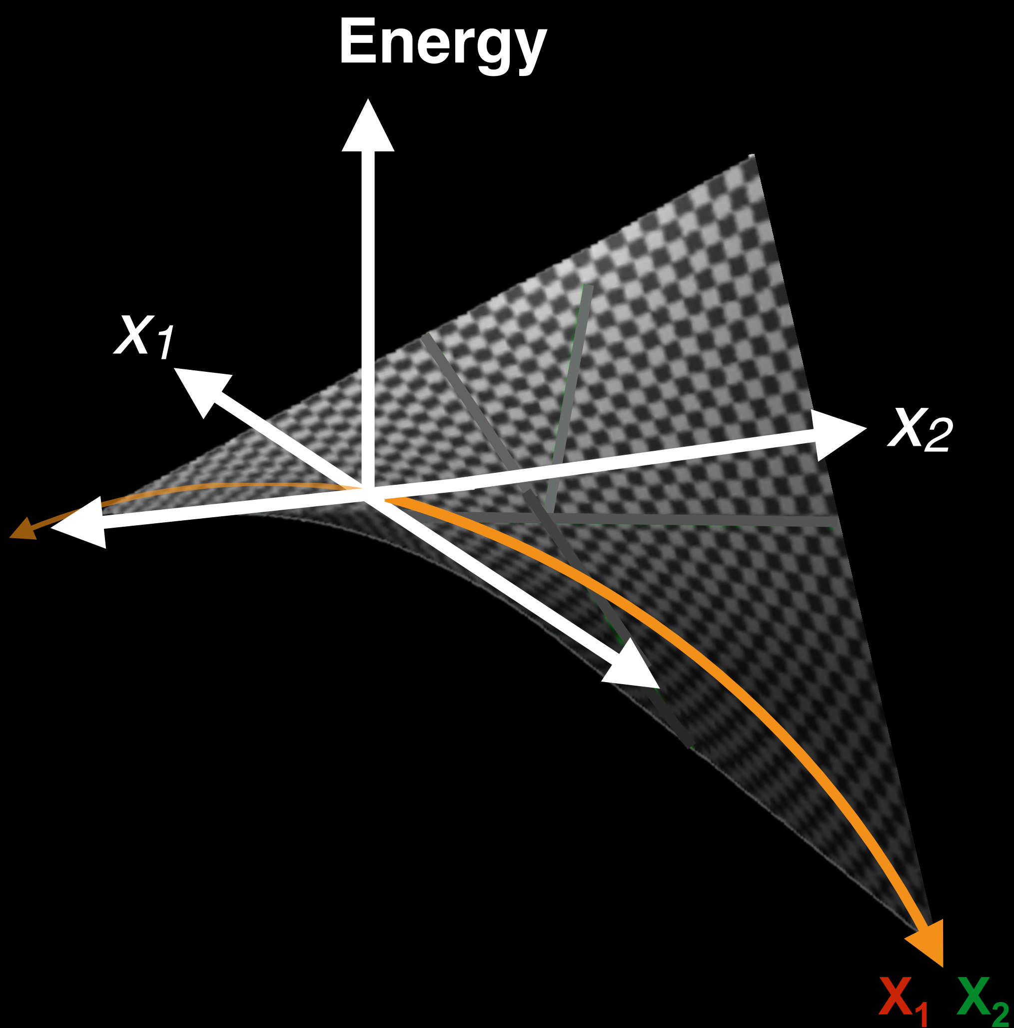
Energy Landscapes

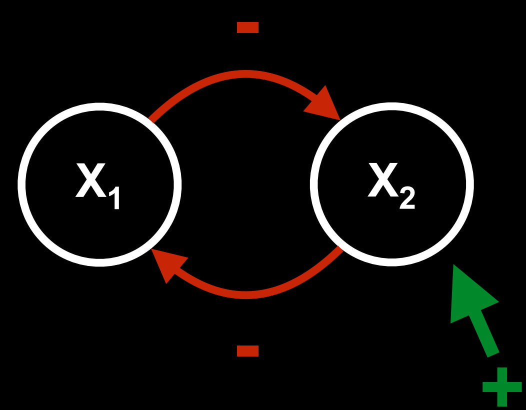




Mutually Inhibitory units

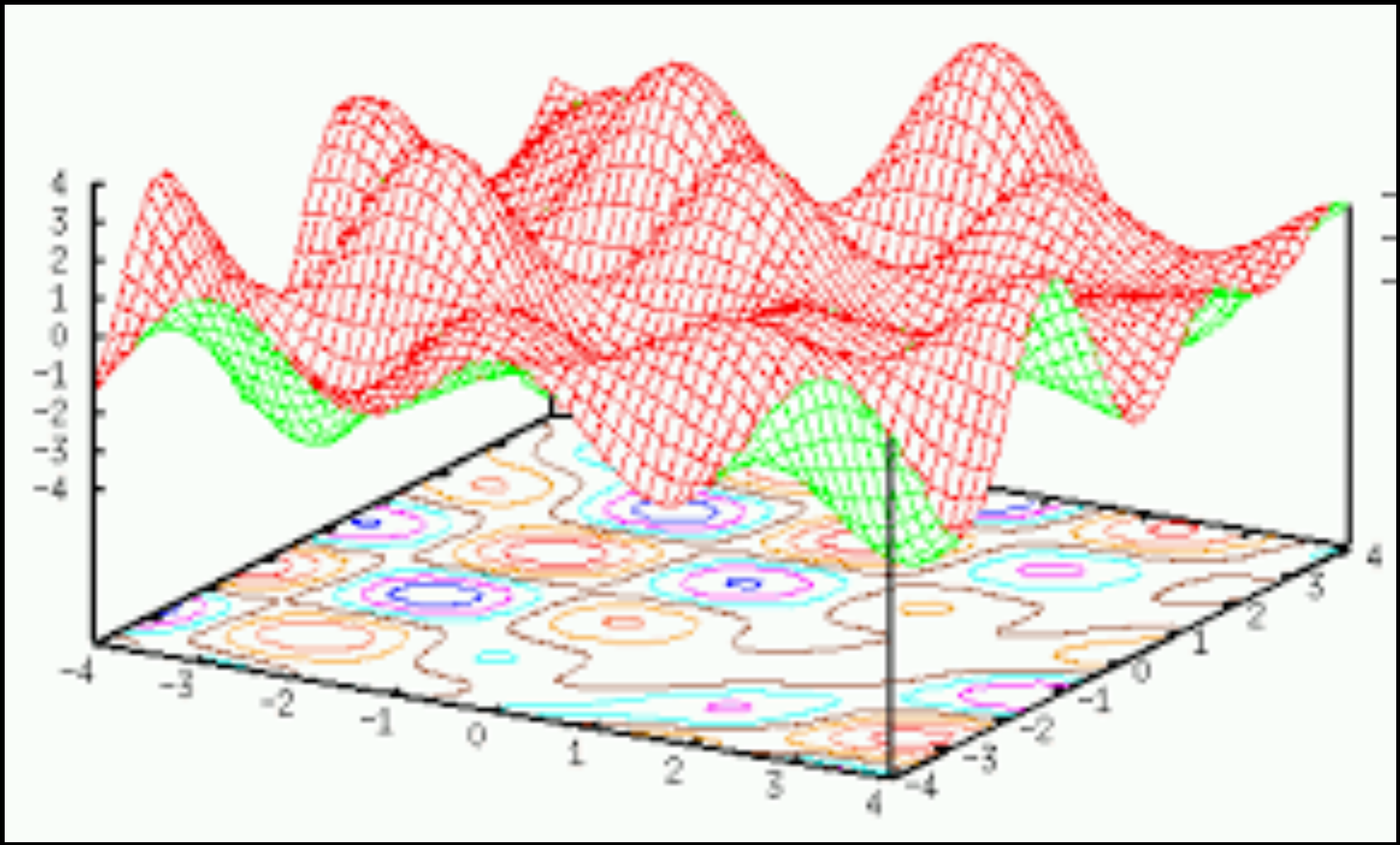
Energy Landscapes





Mutually Inhibitory units
and strong input to x_2

Energy Landscapes



Minima

Minima

- Minima define stable states of the system: attractors
  - "wells" in the energy landscape

Minima

- Minima define stable states of the system: attractors
  - "wells" in the energy landscape
- System will head to the nearest well: settling
  - ball will roll downhill to the nearest well and stay there

Minima

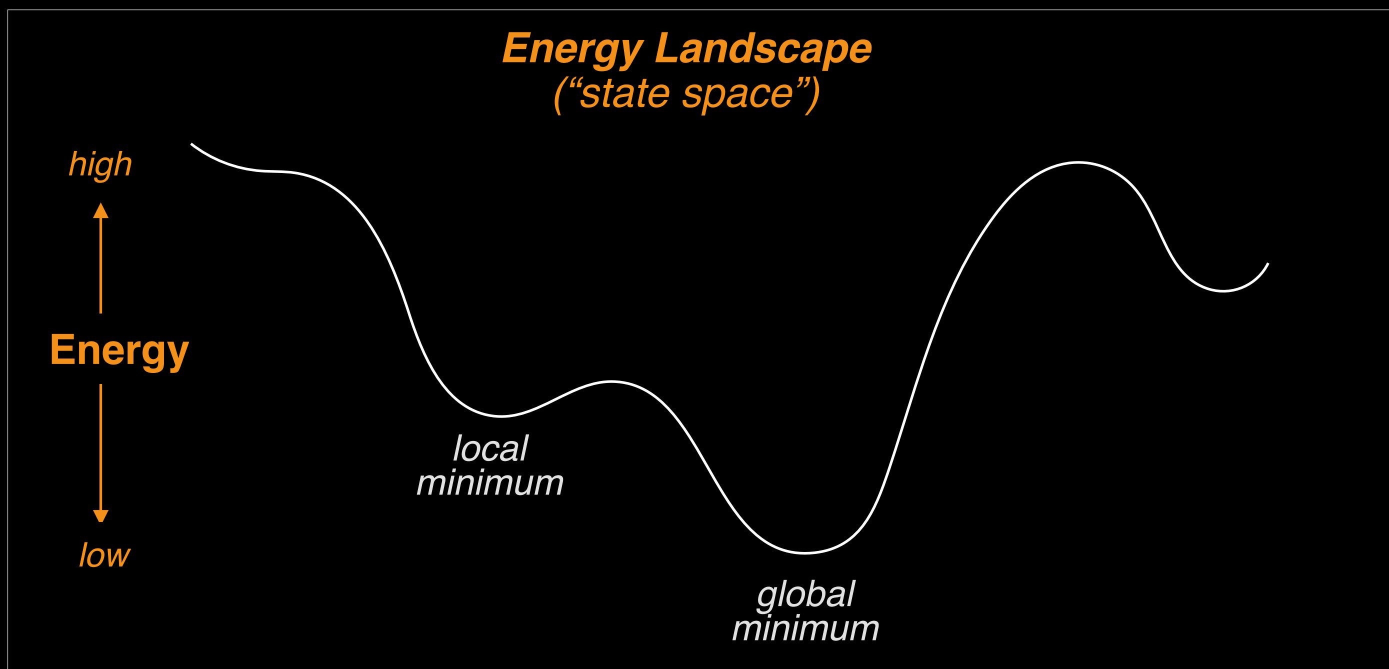
- Minima define stable states of the system: attractors
  - "wells" in the energy landscape
- System will head to the nearest well: settling
  - ball will roll downhill to the nearest well and stay there
- **Settling = perception** *(or retrieval)*
  - **values of the units in that state reflect the properties of the perceptual interpretation** *(or retrieved memory)*

Minima

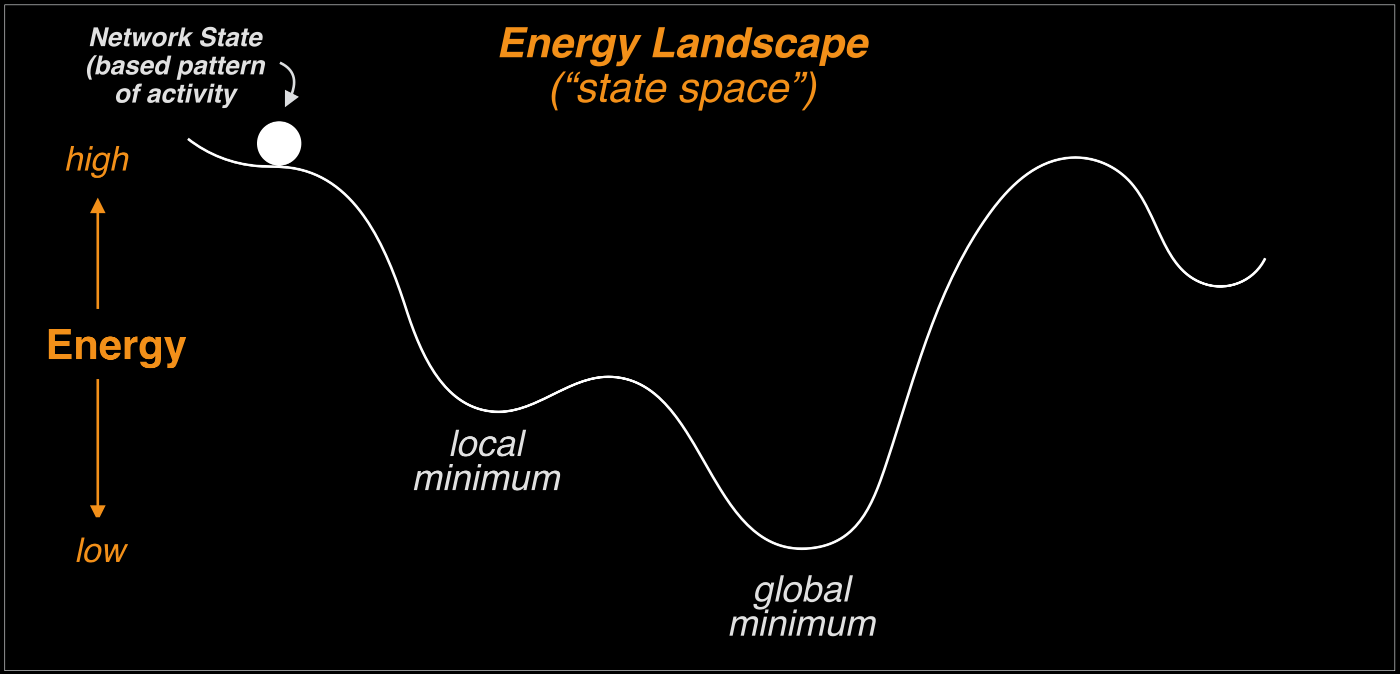
- Minima define stable states of the system: attractors
  - "wells" in the energy landscape
- System will head to the nearest well: settling
  - ball will roll downhill to the nearest well and stay there
- Settling = perception *(or retrieval)*
  - values of the units in that state reflect the properties of the perceptual interpretation *(or retrieved memory)*
- No guarantee that nearest minimum is the best: local minima
  - ball can get stuck in a shallow well before finding deepest one…

Local Minima

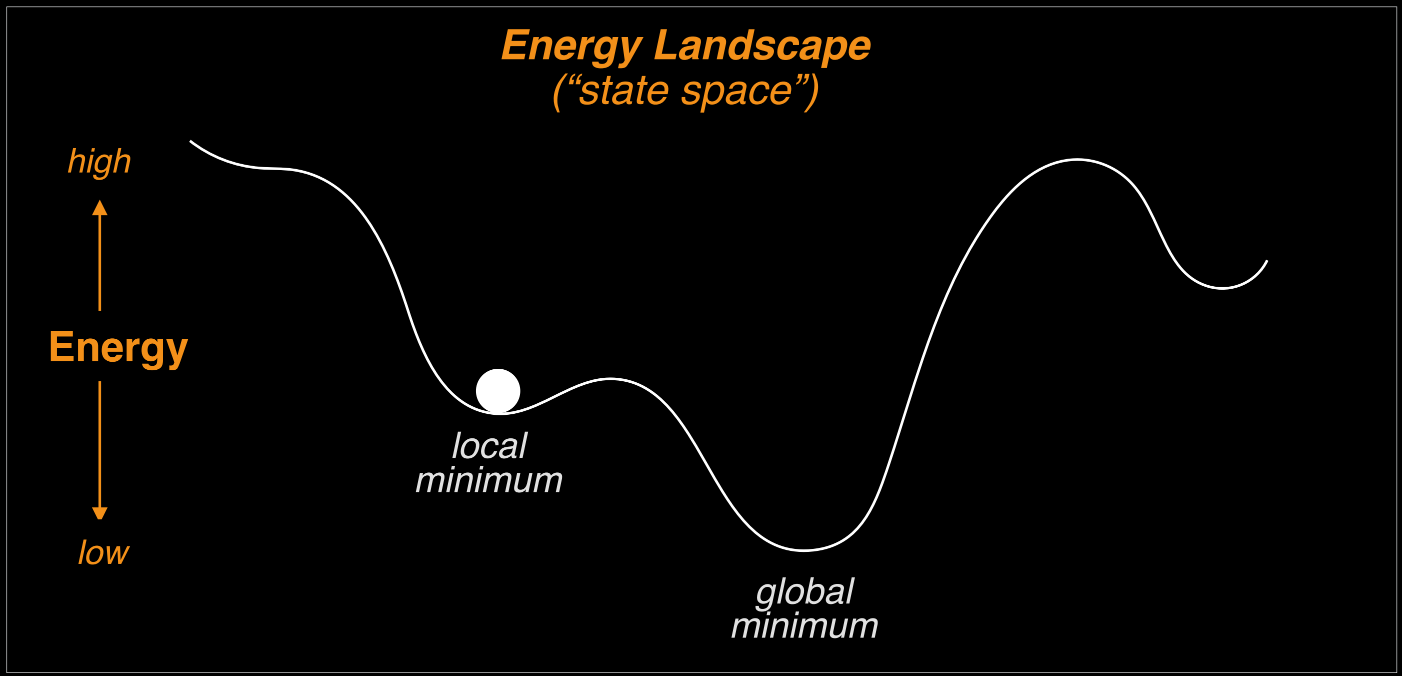
Local Minima



Local Minima



Local Minima



Local Minima

Local Minima

- Predisposing factors to getting stuck in local minima:
  - – Neighborhood effects (coalitions)
  - – Binary units or large changes in activation (rapid updating)

Local Minima

- **Predisposing factors to getting stuck in local minima:**
  - **Neighborhood effects (coalitions)**
  - **Binary units or large changes in activation (rapid updating)**

- **Avoiding local minima** *("annealing")*:
  - **Continuous activation values**

    small adjustments in activity in each update prevents units from "committing themselves" before they feel more distant (global) influences

Local Minima

- **Predisposing factors to getting stuck in local minima:**
  - **Neighborhood effects (coalitions)**
  - **Binary units or large changes in activation (rapid updating)**

- **Avoiding local minima** *(“annealing”)*:
  - **Continuous activation values**

    small adjustments in activity in each update prevents units from “committing themselves” before they feel more distant (global) influences

    but it takes longer to settle

Local Minima

- **Predisposing factors to getting stuck in local minima:**
  - **Neighborhood effects (coalitions)**
  - **Binary units or large changes in activation (rapid updating)**

- **Avoiding local minima** *("annealing")***:**
  - **Continuous activation values**

    small adjustments in activity in each update prevents units from "committing themselves" before they feel more distant (global) influences

    but it takes longer to settle

  - **Stochastic activation (noisy input)**

Local Minima

- **Predisposing factors to getting stuck in local minima:**
  - **Neighborhood effects (coalitions)**
  - **Binary units or large changes in activation (rapid updating)**

- **Avoiding local minima *("annealing")*:**
  - **Continuous activation values**

    small adjustments in activity in each update prevents units from "committing themselves" before they feel more distant (global) influences

    but it takes longer to settle

  - **Stochastic activation (noisy input)**

    allows the system to "correct" itself…
    can back out of blind alley, or literally "jump" out of a local minimum

Local Minima

- **Predisposing factors to getting stuck in local minima:**
  - **Neighborhood effects (coalitions)**
  - **Binary units or large changes in activation (rapid updating)**

- **Avoiding local minima** *("annealing")***:**
  - **Continuous activation values**

    small adjustments in activity in each update prevents units from "committing themselves" before they feel more distant (global) influences

    but it takes longer to settle

  - **Stochastic activation (noisy input)**

    allows the system to "correct" itself…
    can back out of blind alley, or literally "jump" out of a local minimum

    but less predictable

Local Minima

- **Predisposing factors to getting stuck in local minima:**
  - **Neighborhood effects (coalitions)**
  - **Binary units or large changes in activation (rapid updating)**

- **Avoiding local minima *("annealing")*:**
  - **Continuous activation values**

    small adjustments in activity in each update prevents units from "committing themselves" before they feel more distant (global) influences

    but it takes longer to settle

  - **Stochastic activation (noisy input)**

    allows the system to "correct" itself…
    can back out of blind alley, or literally "jump" out of a local minimum

    but less predictable

  - **Temperature modulates the effects of both…**

Effects of Temperature

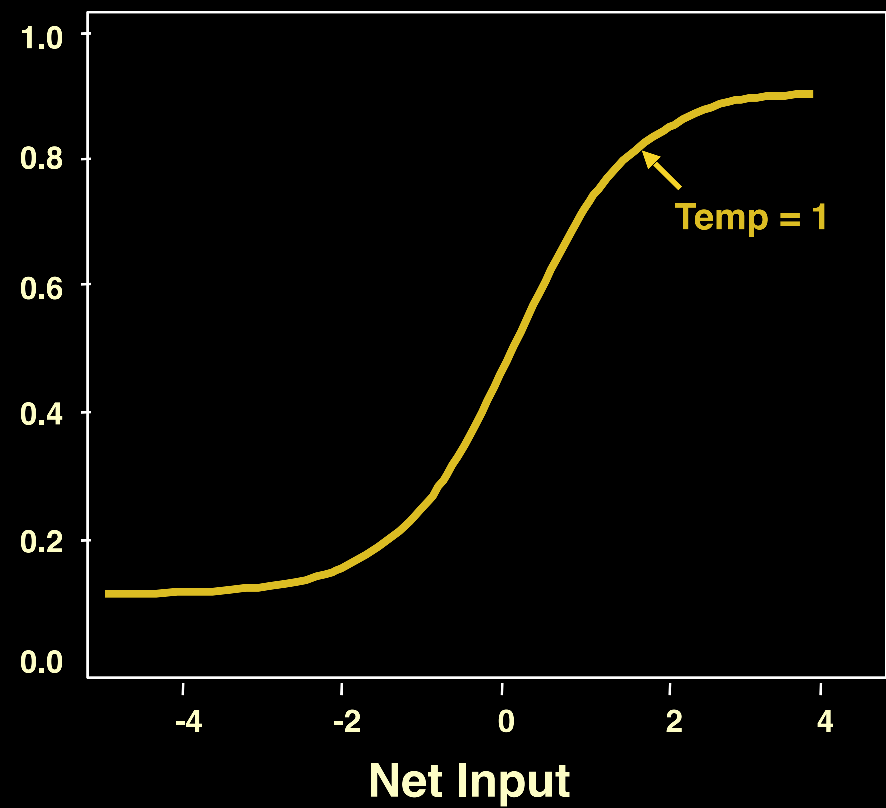
$$\frac{1}{1 + e^{-\frac{net\ input}{Temp}}}$$

Activity

(Hopfield Net, 1984)

Probability of firing

(Boltzmann Machine)



Effects of Temperature

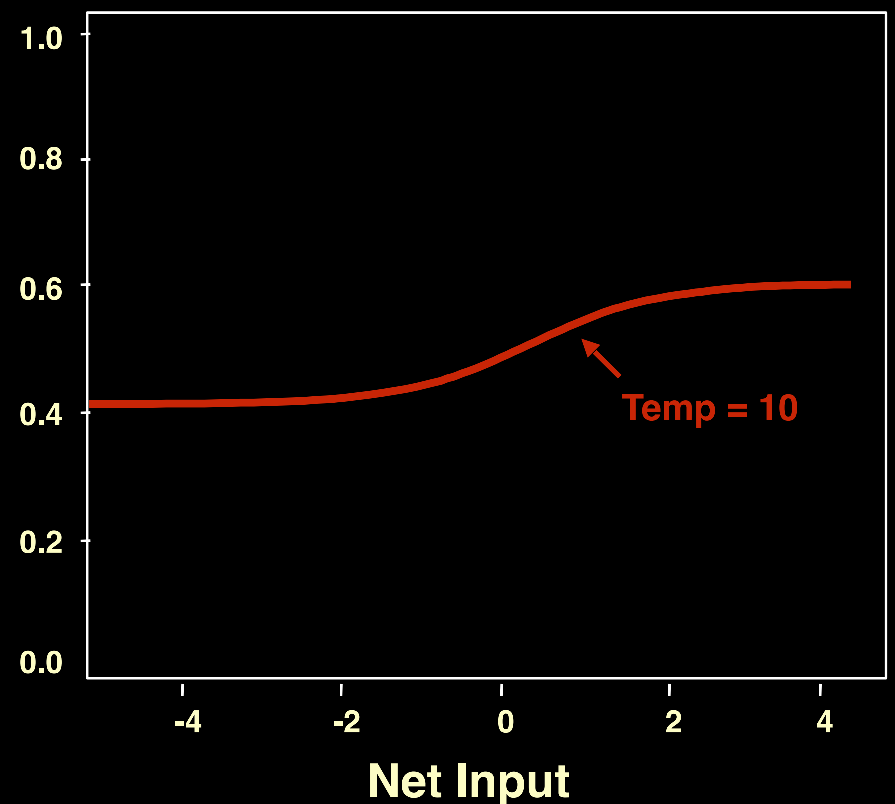
$$\frac{1}{1 + e^{-\frac{net\ input}{Temp}}}$$

Activity

(Hopfield Net, 1984)

Probability of firing

(Boltzmann Machine)

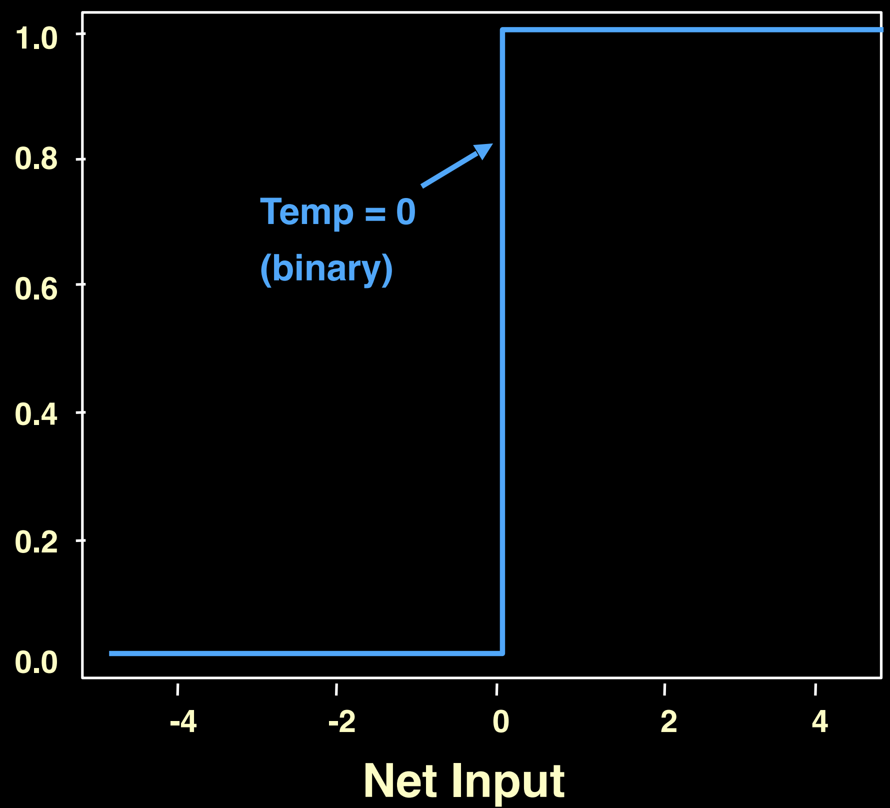


Effects of Temperature

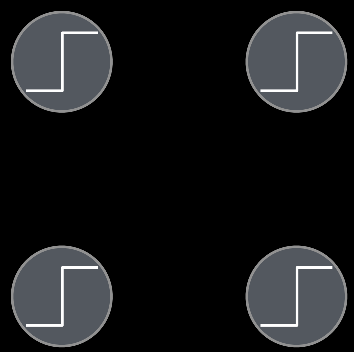
$$\frac{1}{1 + e^{-\frac{net\ input}{Temp}}}$$

Activity
(Hopfield Net, 1984)

Probability of firing
(Boltzmann Machine)

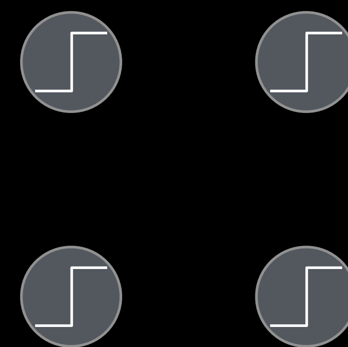


Hopfield (1982) Network



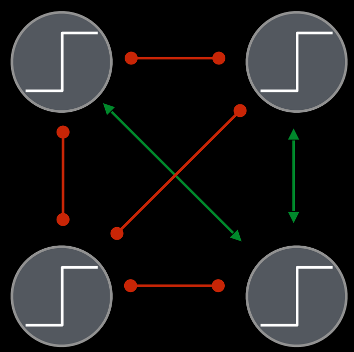
Hopfield (1982) Network

- **Critical assumptions**
  - **Non-linear (binary) units**
    - **forces them to make “decisions:”**
      - categorize an input as reflecting one memory or another;
      - vs. linear systems that represent *graded* blends of options



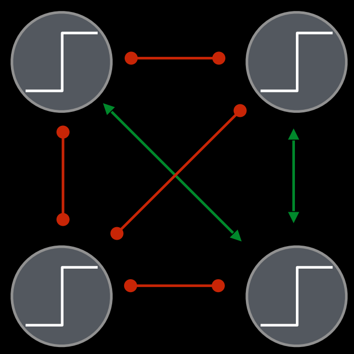
Hopfield (1982) Network

- Critical assumptions
  - Non-linear (binary) units
    - forces them to make "decisions:"
      - categorize an input as reflecting one memory or another;
      - vs. linear systems that represent *graded* blends of options
  - Recurrent connections
    - provides basis for settling dynamics and attractors
    - symmetric: required for analysis, but not critical for function



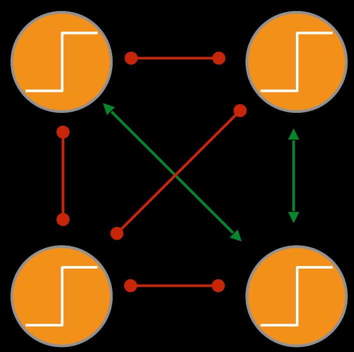
Hopfield (1982) Network

- Critical assumptions
  - Non-linear (binary) units
    - forces them to make "decisions:"
      - categorize an input as reflecting one memory or another;
      - vs. linear systems that represent *graded* blends of options
  - Recurrent connections
    - provides basis for settling dynamics and attractors
    - symmetric: required for analysis, but not critical for function
  - Asynchronous updating
    - biologically plausible
    - insures symmetry breaking, avoids "see-sawing" (pseudo-stochasticity)



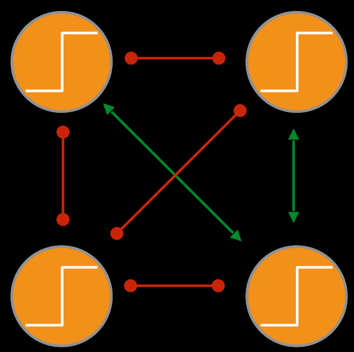
Hopfield (1982) Network

- Critical assumptions
  - Non-linear (binary) units
    - ♦ forces them to make "decisions:"
      - categorize an input as reflecting one memory or another;
      - vs. linear systems that represent *graded* blends of options
  - Recurrent connections
    - ♦ provides basis for settling dynamics and attractors
    - ♦ symmetric: required for analysis, but not critical for function
  - Asynchronous updating
    - ♦ biologically plausible
    - ♦ insures symmetry breaking, avoids "see-sawing" (pseudo-stochasticity)



Hopfield (1982) Network

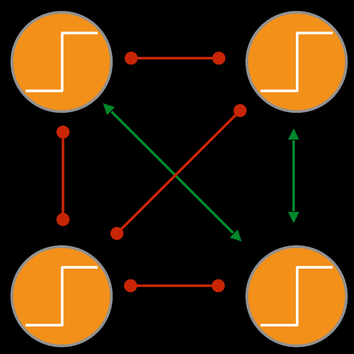
- Critical assumptions
  - Non-linear (binary) units
    - forces them to make “decisions:”
      - categorize an input as reflecting one memory or another;
      - vs. linear systems that represent *graded* blends of options
  - Recurrent connections
    - provides basis for settling dynamics and attractors
    - symmetric: required for analysis, but not critical for function
  - Asynchronous updating
    - biologically plausible
    - insures symmetry breaking, avoids “see-sawing” (pseudo-stochasticity)



- Additional assumptions
  - Deterministic
    - each unit does exactly what it is “told” by its neighbors (no noise)

Hopfield (1982) Network

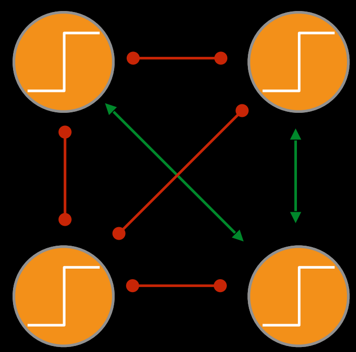
- Critical assumptions
  - Non-linear (binary) units
    - forces them to make “decisions:”
      - categorize an input as reflecting one memory or another;
      - vs. linear systems that represent *graded* blends of options
  - Recurrent connections
    - provides basis for settling dynamics and attractors
    - symmetric: required for analysis, but not critical for function
  - Asynchronous updating
    - biologically plausible
    - insures symmetry breaking, avoids “see-sawing” (pseudo-stochasticity)



- Additional assumptions
  - Deterministic
    - each unit does exactly what it is “told” by its neighbors (no noise)
  - No self-connections
    - no “memory; each unit governed entirely by sampling its neighbors

Hopfield (1982) Network

- Critical assumptions
  - Non-linear (binary) units
    - forces them to make "decisions:"
      - categorize an input as reflecting one memory or another;
      - vs. linear systems that represent *graded* blends of options
  - Recurrent connections
    - provides basis for settling dynamics and attractors
    - symmetric: required for analysis, but not critical for function
  - Asynchronous updating
    - biologically plausible
    - insures symmetry breaking, avoids "see-sawing" (pseudo-stochasticity)



- Additional assumptions
  - Deterministic
    - each unit does exactly what it is "told" by its neighbors (no noise)
  - No self-connections
    - no "memory; each unit governed entirely by sampling its neighbors
  - **Otherwise, fully interconnected**

Contributions

Contributions

- Demonstration of how a neural network can compute:
  - Biologically-inspired assumptions

Contributions

- **Demonstration of how a neural network can compute:**
  - **Biologically-inspired assumptions**
  - **Percepts** *(memories)* **can be represented** *(stored)* **as minima (attractors)**

Contributions

- **Demonstration of how a neural network can compute:**
  - **Biologically-inspired assumptions**
  - **Percepts** *(memories)* **can be represented** *(stored)* **as minima (attractors)**
  - **Algorithm for producing them using Hebbian learning**

Contributions

- **Demonstration of how a neural network can compute:**
  - **Biologically-inspired assumptions**
  - **Percepts** *(memories)* **can be represented** *(stored)* **as minima (attractors)**
  - **Algorithm for producing them using Hebbian learning**
  - **Emergent properties:**
    - Gestalt categorization
    - Content-addressability
    - Dynamics

Contributions

- **Demonstration of how a neural network can compute:**
  - **Biologically-inspired assumptions**
  - **Percepts** *(memories)* **can be represented** *(stored)* **as minima (attractors)**
  - **Algorithm for producing them using Hebbian learning**
  - **Emergent properties:**
    - Gestalt categorization
    - Content-addressability
    - Dynamics
- **Capacity:**
  - **roughly 15% no. of units, before minima become too narrow / shallow**

Contributions

- Demonstration of how a neural network can compute:
  - Biologically-inspired assumptions
  - Percepts *(memories)* can be represented *(stored)* as minima (attractors)
  - Algorithm for producing them using Hebbian learning
  - Emergent properties:
    - Gestalt categorization
    - Content-addressability
    - Dynamics
- Capacity:
  - roughly 15% no. of units, before minima become too narrow / shallow
- Connection to statistical mechanics *(Ising model)*:

Contributions

- **Demonstration of how a neural network can compute:**
  - **Biologically-inspired assumptions**
  - **Percepts** *(memories)* **can be represented** *(stored)* **as minima (attractors)**
  - **Algorithm for producing them using Hebbian learning**
  - **Emergent properties:**
    - Gestalt categorization
    - Content-addressability
    - Dynamics
- **Capacity:**
  - **roughly 15% no. of units, before minima become too narrow / shallow**
- **Connection to statistical mechanics** *(Ising model)***:**
  - **Can think about neural networks in terms of state-space (or "phase space") dynamics of energy minimization & annealing**

Contributions

- Demonstration of how a neural network can compute:
  - Biologically-inspired assumptions
  - Percepts *(memories)* can be represented *(stored)* as minima (attractors)
  - Algorithm for producing them using Hebbian learning
  - Emergent properties:
    - Gestalt categorization
    - Content-addressability
    - Dynamics
- Capacity:
  - roughly 15% no. of units, before minima become too narrow / shallow
- Connection to statistical mechanics *(Ising model)*:
  - Can think about neural networks in terms of state-space (or "phase space") dynamics of energy minimization & annealing
  - Energy landscapes describe network dynamics

Contributions

- **Demonstration of how a neural network can compute:**
  - **Biologically-inspired assumptions**
  - **Percepts** *(memories)* **can be represented** *(stored)* **as minima (attractors)**
  - **Algorithm for producing them using Hebbian learning**
  - **Emergent properties:**
    - Gestalt categorization
    - Content-addressability
    - Dynamics
- **Capacity:**
  - **roughly 15% no. of units, before minima become too narrow / shallow**
- **Connection to statistical mechanics** *(Ising model)*:
  - **Can think about neural networks in terms of state-space (or "phase space") dynamics of energy minimization & annealing**
  - **Energy landscapes describe network dynamics**
- **Connection to quantum computing** *(Hamiltonian dynamics)*

Comparison of Models

Model	Activation	Updating	Connections
Hopfield (1982)	Binary	Deterministic	Symmetric
Hopfield (1984)	Continuous	Deterministic	Symmetric
Bolzmann Machine	Binary	Stochastic	Asymmetric
Interactive Activation and Competition (IAC)	Continuous	Deterministic	Symmetric
Leaky Competing Accumulator (LCA)	Continuous	Stochastic	Asymetric

Comparison of Models

- To what extent are setting dynamics psychologically/ neurally plausible?

Comparison of Models

- How can such models be used to account for empirical data...

Interactive Activation Model of Word Perception

McClelland & Rumelhart, 1981

Interactive Activation Model of Word Perception

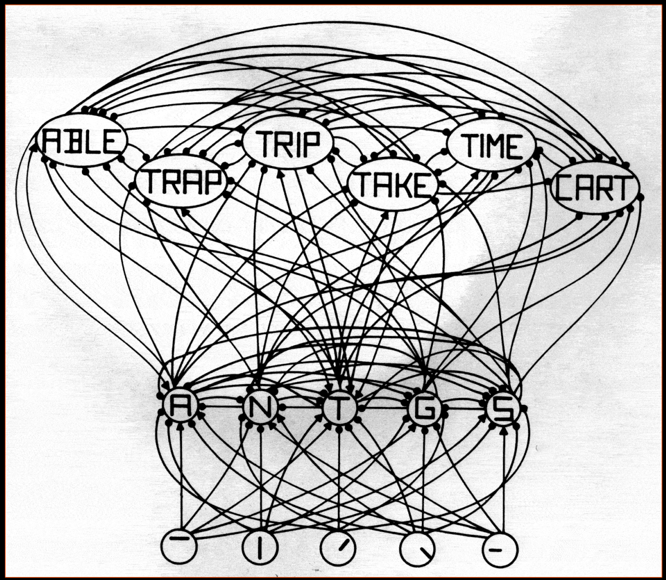
McClelland & Rumelhart, 1981

- Word superiority effect:
  - Faster to recognize a letter in the context of a word than alone

Interactive Activation Model of Word Perception

McClelland & Rumelhart, 1981

- Word superiority effect:
  - Faster to recognize a letter in the context of a word than alone

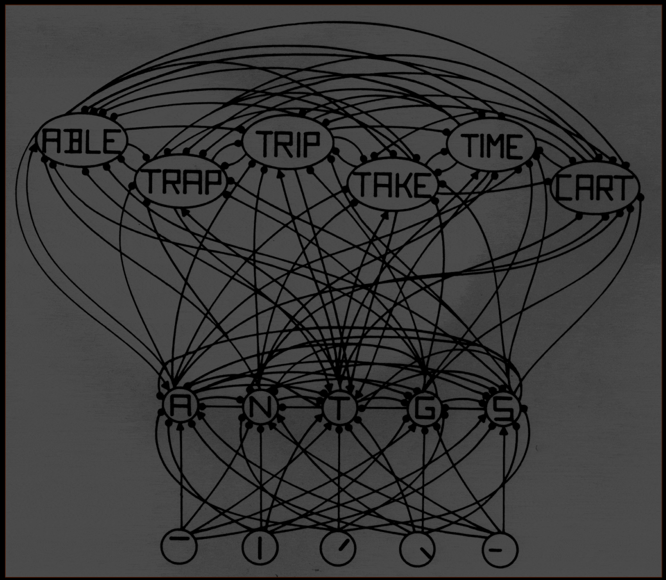


- IAC model:
  - Accounts for empirical findings regarding word perception:
    - frequency effects
    - neighborhood effects
    - word superiority effects
  - Predicted new perceptual phenomena
  - Landmark in formal modeling of complex psychological phenomena using connectionist architecture

Interactive Activation Model of Word Perception

McClelland & Rumelhart, 1981

- Word superiority effect:
  - Faster to recognize a letter in the context of a word than alone



- IAC model:
  - Accounts for empirical findings regarding word perception:
    - frequency effects
    - neighborhood effects
    - word superiority effects
  - Predicted new perceptual phenomena
  - Landmark in formal modeling of complex psychological phenomena using connectionist architecture

- Will come back to this under section on language