From connectivity to dynamics and function in recurrent networks

Srdjan Ostojic
srdjan.ostojic@ens.fr
Group for Neural Theory
Ecole Normale Superieure Paris
Neural activity in V1

orientation selectivity

[Hubel and Wiesel 1959, 1962]
Sensory systems: Receptive fields

Linear-nonlinear model

\[ L(t) = \int_0^\infty d\tau \int dx dy \, D(x, y, \tau)s(x, y, t - \tau) \]

“receptive field”

stimulus

Stephen David’s lecture
V1 Receptive fields

[Hubel and Wiesel 1959]
V1 Receptive fields

orientation selectivity
V1 Receptive fields: Gabor functions

function: efficient coding of natural stimuli

\[ D_s(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp \left( -\frac{x^2}{2\sigma_x^2} - \frac{y^2}{2\sigma_y^2} \right) \cos(kx - \phi). \]
V1 circuitry

which connections generate orientation tuning?

[Ferster and Miller 2000]
Thalamic inputs: LGN cells

Receptive Fields

On-center, Off-surround
Off-center, On-surround

[center surround][Hartline, Kuffler]
Feedforward model

orientation selectivity from LGN inputs

[Hubel and Wiesel 1962]
Feed-forward model: Hubel and Wiesel

(a)  
(b)  
(c)  

mechanistic explanation

[Hubel and Wiesel 1962]
Evidence for the feed-forward model

[Alonso and Reid 1995]
Evidence for the feed-forward model

[Alonso and Reid 1995]
- single cell activity has a clear functional role
- this activity can be understood mechanistically in terms of connectivity

Connectivity ↔ Neural activity

Behavior/Function
role of recurrent connectivity?
1. The Garden of Eden: network models of pure selectivity

2. Facing the Jungle: training random recurrent networks
1. The Garden of Eden: network models of pure selectivity
   - working memory and persistent activity
   - decision making
   - working memory and decision making

2. Facing the Jungle: training random recurrent networks
Delay match to sample task

how is information maintained in working memory?

[Miyashita and Chang 1988]
Persistant mnemonic activity

example cell

function: working memory

[Miyashita and Chang 1988]
Persistant mnemonic activity

underlying mechanism?

[Miyashita and Chang 1988]
Cortical Networks

role of recurrent connectivity?
Mechanism for persistent activity: positive feedback


strong recurrent excitation

Mechanism for persistent activity: positive feedback

Dynamical mechanism: bistability

general mechanism for memory
Understanding the mechanism: rate model
Understanding the mechanism: rate model

\[ \frac{d}{dt} x = -x + JF(x) + I \]
Understanding the mechanism: rate model
Multiple item working memory

Brunel & Wang 2001
Multiple item working memory

Brunel & Wang 2001
1. The Garden of Eden: network models of pure selectivity
   - working memory and persistent activity
   - decision making
   - working memory and decision making

2. Facing the Jungle: training random recurrent networks
Decision making

stimulus → Sensory Area → Association Area → Motor Area → action

visual system: Movshon, Newsome, Shadlen
somato-sensory: Romo
Random-dot task

[Gold and Shadlen 2007]
Random-dot task

[Gold and Shadlen 2007]
Decision making

stimulus → Sensory Area → Association Area → Motor Area → action

MT cortex

[Albright 1984]
Activity in LIP

select T1

select T2

motion sac

51.2%

6.4%

Spikes/s

500 ms

[Gold and Shadlen 2007]
Activity in LIP

**Function:** accumulation of information and decision making

[Gold and Shadlen 2007]
Attractor model of decision making

[Wang 2002]
Attractor model of decision making

Population A

Population B

[Wang 2002]
Attractor model of decision making

[A] Trial 1

[B] Trial 2

[Wang 2002]
Attractor model of decision making

[Diagram showing decision space with points A and B, and a line indicating winner-take-all dynamics.]

winner-take-all dynamics

[Wang 2002]
Dynamical mechanism

[Wong & Wang 2006]
Understanding the mechanism: rate model

winner-take-all dynamics

[Wong & Wang 2006]
Understanding the mechanism: rate model

\[
\frac{d}{dt} x_1 = -x_1 - JF(x_2) + I_1 \\
\frac{d}{dt} x_2 = -x_2 - JF(x_1) + I_2
\]
Outline

1. The Garden of Eden: network models of pure selectivity
   - working memory and persistent activity
   - decision making
   - working memory and decision making

2. Facing the Jungle: training random recurrent networks
A parametric working memory task

[Romo et al 1999]
Neural activity

function: working memory + decision

[Machens, Romo, Brody 2005]
Dynamical mechanism

[Machens, Romo, Brody 2005]
Rate model

[Machens, Romo, Brody 2005]
Rate model
Rate model

[Machens, Romo, Brody 2005]
1. The Garden of Eden: network models of pure selectivity
   - working memory and persistent activity
   - decision making
   - working memory and decision making

2. Facing the Jungle: training random recurrent networks
Looking at the full population

[Machens, Romo, Brody 2010]
Looking at the full population

[Image: Graph showing interactions between neuron 1, neuron 2, and neuron 3.

References: Machens, Romo, Brody 2010]
Looking at the full population

Principal Component Analysis (PCA)

[Graph showing relative rate (Hz) over time (sec) with contributions of 57.3%, 17.4%, and 14.4%]
Looking at the full population

[Kobak et al 2016]
Link with network dynamics
Link with network dynamics
Randomly connected networks

\[ \dot{x}_i(t) = -x_i(t) + g \sum_{j=1}^{N} \chi_{ij} \phi(x_j(t)) \]

\[ \phi(x) = \tanh(x) \]

\[ \chi_{ij} \sim \mathcal{N}(0, \frac{1}{\sqrt{N}}) \]

[Sompolinsky, Crisanti and Sommers 1988]
Dynamics

\[ g < 1 \]

\[ g > 1 \]

[Sompolinsky, Crisanti and Sommers 1988]
Spontaneous dynamics

\[ g < 1 \]

\[ g > 1 \]

computational “reservoir”
Liquid state / Echo state / Reservoir computing
Reservoir computing

\[ Z(t) = \sum_{j=1}^{N} w_j \phi(x_j) \]

- Input
- Reservoir
- Readout
Reservoir computing

Input �ightharpoonup Reservoir �ightharpoonup Readout

\[ Z(t) = \sum_{j=1}^{N} w_j \phi(x_j) \]

Activation \( x_i(t) \)

Reservoir computing

\[ Z(t) = \sum_{j=1}^{N} w_j \phi(x_j) \]

Input \rightarrow \text{Reservoir (basis functions)} \rightarrow \text{Readout}
Reservoir computing

Input \rightarrow \text{Reservoir} \rightarrow \text{Readout}

\text{TRAIN}

cost function:
\[
\sum_{\text{time, trials}} \left[ \sum_{j=1}^{N} w_j \phi(x_j(t)) - Z_{\text{target}}(t) \right]^2
\]

minimise over readout weights \( w \)

= linear regression problem

target output: \( Z_{\text{target}} \)
Reservoir computing

Input → Reservoir → Readout

**TRAIN**

**cost function:**

\[
\sum_{\text{time, trials}} \left[ \sum_{j=1}^{N} w_j \phi(x_j(t)) - Z_{target}(t) \right]^2
\]

\[
= \| W \Phi - Z_{target} \|^2
\]

minimise over readout weights \(w\)

= linear regression problem

\[
W = (\Phi^T \Phi)^{-1} \Phi^T Z_{target}
\]
Implementing a task

[Barak et al 2013]
Implementing a task

[Barak et al 2013]
Implementing a task

Examples of activity

![Activity plots](image)

dPCA

![dPCA plots](image)

[Barak et al 2013]
Training a recurrent network

cost function:

\[ \sum_{\text{time, trials}} \left[ \sum_{j=1}^{N} w_j \phi(x_j(t)) - Z_{\text{target}}(t) \right]^2 \]

minimise over all connection weights:
difficult!
Training a recurrent network

A simplified approach: adding feedback

[Jaeger and Haas 2004, Sussillo and Abbott 2009]
Networks with feedback

\[ \tau \dot{x}_i(t) = -x_i(t) + g \sum_{j=1}^{N} \chi_{ij} \phi(x_j(t)) + u_i Z(t) \]

\[ Z(t) = \sum_{j=1}^{N} w_j \phi(x_j) \]

[Jaeger and Haas 2004, Sussillo and Abbott 2009]
Networks with feedback

\[ \tau \dot{x}_i(t) = -x_i(t) + g \sum_{j=1}^{N} \chi_{ij} \phi(x_j(t)) + u_i Z(t) \]

\[ Z(t) = \sum_{j=1}^{N} w_j \phi(x_j) \]

\[ \tau \dot{x}_i(t) = -x_i(t) + \sum_{j=1}^{N} (g \chi_{ij} + u_i w_j) \phi(x_j(t)) \]

[Jaeger and Haas 2004, Sussillo and Abbott 2009]

rank 1 perturbation of random connectivity
Training a recurrent network

cost function: \[ \sum_{\text{time, trials}} \left[ \sum_{j=1}^{N} w_j \phi(x_j(t)) - Z_{\text{target}}(t) \right]^2 \]

other approaches: backpropagation + tricks

[Barak 2017]
Implementing a task

[Barak et al 2013]
Comparing approaches

<table>
<thead>
<tr>
<th>Model</th>
<th>Architecture</th>
<th>Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>LA (Line attractor)</td>
<td><img src="LA.png" alt="Diagram" /></td>
<td><img src="LA.png" alt="Performance" /></td>
</tr>
<tr>
<td>RN (Chaotic random)</td>
<td><img src="RN.png" alt="Diagram" /></td>
<td><img src="RN.png" alt="Performance" /></td>
</tr>
<tr>
<td>TRAIN (HF Training)</td>
<td><img src="TRAIN.png" alt="Diagram" /></td>
<td><img src="TRAIN.png" alt="Performance" /></td>
</tr>
</tbody>
</table>

[Barak et al 2013]
Comparing approaches

<table>
<thead>
<tr>
<th>Model</th>
<th>A1</th>
<th>A2</th>
</tr>
</thead>
<tbody>
<tr>
<td>LA (Line attractor)</td>
<td><img src="image1" alt="Graph" /></td>
<td><img src="image2" alt="Graph" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model</th>
<th>B1</th>
<th>B2</th>
</tr>
</thead>
<tbody>
<tr>
<td>RN (Chaotic random)</td>
<td><img src="image3" alt="Graph" /></td>
<td><img src="image4" alt="Graph" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model</th>
<th>C1</th>
<th>C2</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRAIN (HF Training)</td>
<td><img src="image5" alt="Graph" /></td>
<td><img src="image6" alt="Graph" /></td>
</tr>
</tbody>
</table>

[Barak et al 2013]
Comparing approaches

dPCA

[Barak et al, 2013]
Training recurrent networks

Cost function:

$$\sum_{\text{time, trials}} \left[ \sum_{j=1}^{N} w_j \phi(x_j(t)) - Z_{\text{target}}(t) \right]^2$$

- popular approach for building network implementations

[Barak et al 2017]
Reverse engineering trained networks

[Sussillo & Barak 2013]
Reverse engineering trained networks

- Dynamics determined by fixed points and saddles

[Sussillo & Barak 2013]
More complex task: finding new mechanisms

dynamics determined by fixed points and saddles

[Mante, Sussillo et al 2013]
Reverse engineering the trained network

Dynamics determined by structure of attractors

[Mante, Sussillo et al 2013]
Comparing approaches

[Barak 2017]
Summary: recurrent networks

Positive feedback $\rightarrow$ bi-stability, memory, persistent activity

Mutual inhibition $\rightarrow$ competition, winner-take-all

Trained recurrent networks $\rightarrow$ searching for new mechanisms