Spiking neural networks

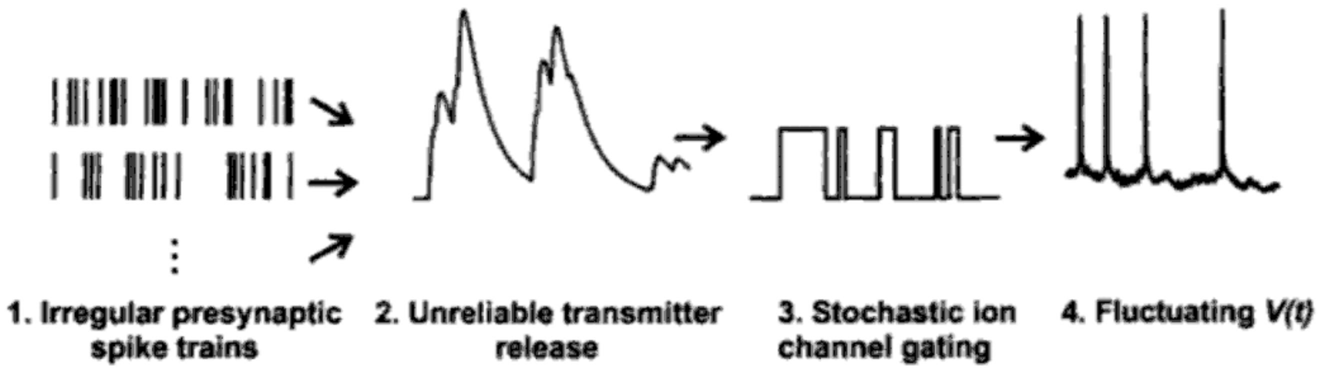
Computational Approaches to Neuroscience (NSCI 850)
Gunnar Blohm

Outline

- Neuronal firing variability
  - Spike time variability
  - Efficient coding hypothesis
- Spiking networks
  - Phase oscillators and synaptic coupling
  - Synchronization and phase locking
  - Example applications
- Hebbian learning
  - Associative memory
  - Synaptic plasticity
  - Mathematical formulation of Hebbian learning
- Discussion

Neuronal firing variability

- Biophysical basis

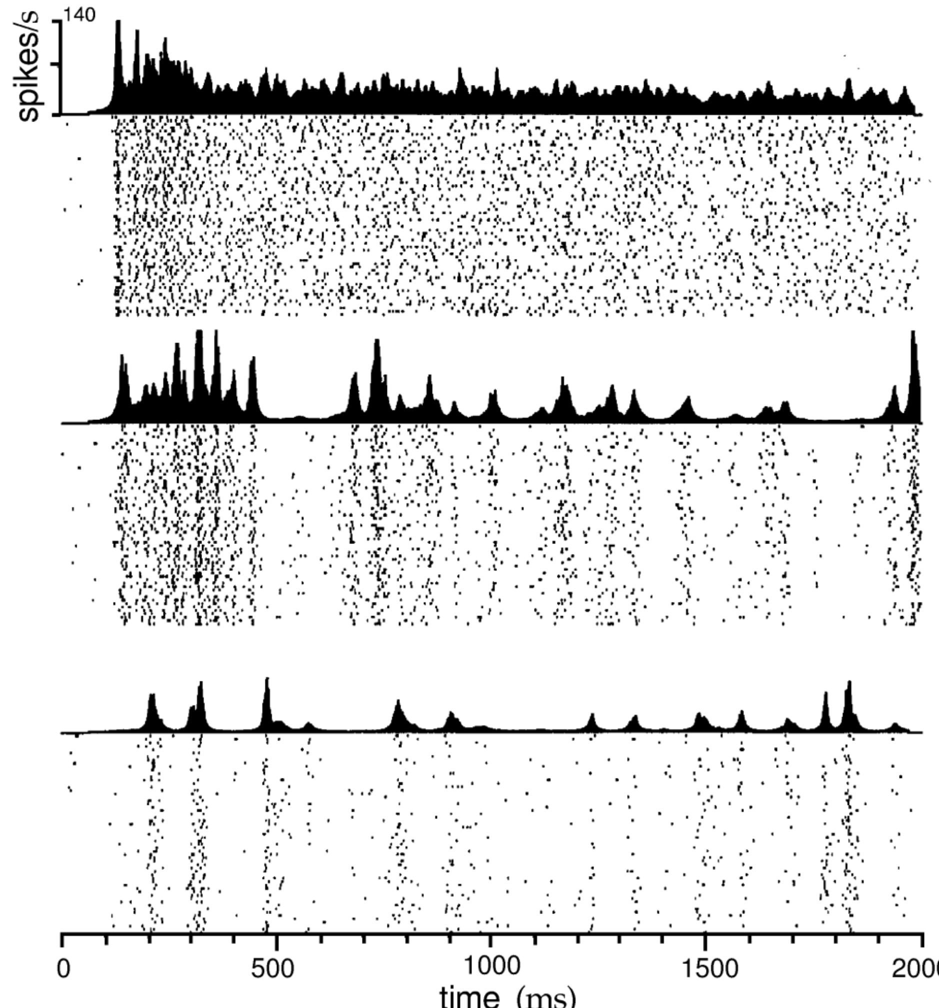


Feng 2004

  - Stochastic pre-synaptic spike trains converge on ~10^4 synaptic terminals distributed over the dendritic tree
  - Highly unreliable and stochastic process of transmitter release at each terminal
  - Transmitters gate ion channels behaving like stochastic processes
  - Induced membrane potential changes gate other voltage-dependent stochastic ion channels…

Spike count variability

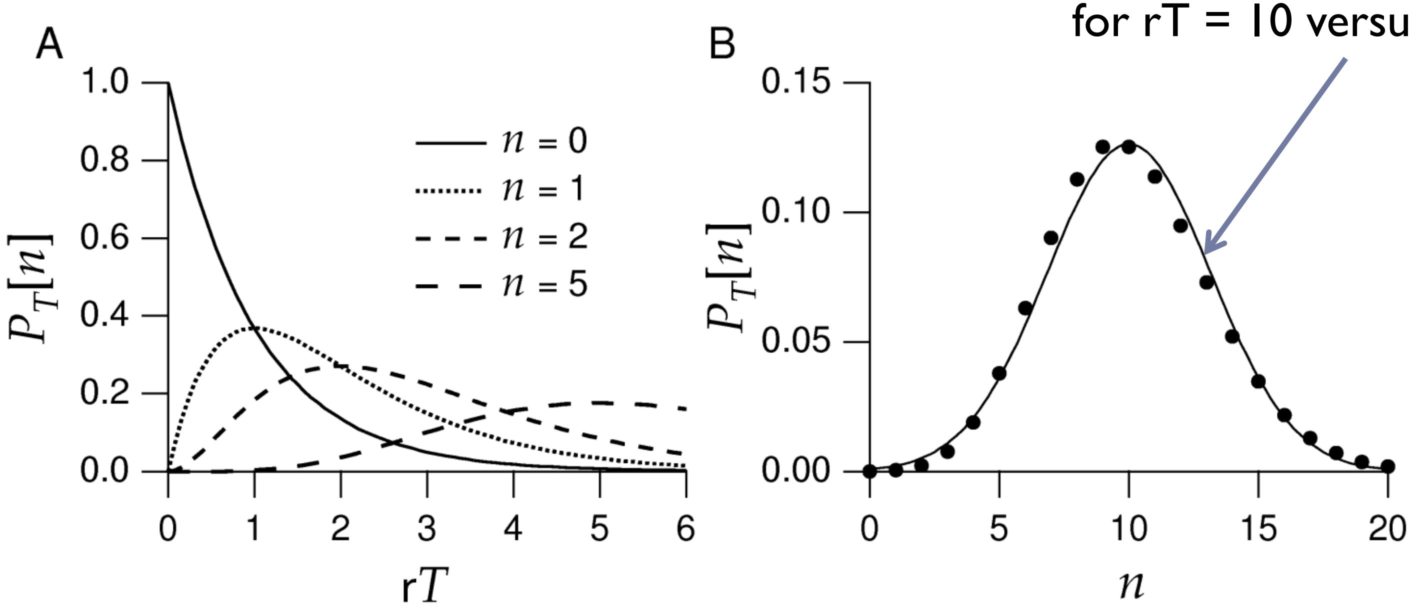
- Spike timings show variability
- If the time between 2 spikes does not depend on the spike history, then we have a Poisson process
  - all events are statistically independent
  - E.g. radioactive decay, web site page view requests…



Dayan & Abbott, 2001

Spike count variability

- Poisson distribution $$P_T[n] = \frac{(rT)^n}{n!} e^{-rT}$$
  - T: considered time interval during which spikes can occur
  - r: spike rate
  - n: number of spikes occurring



Probability to find n spikes for rT = 10 versus Gaussian

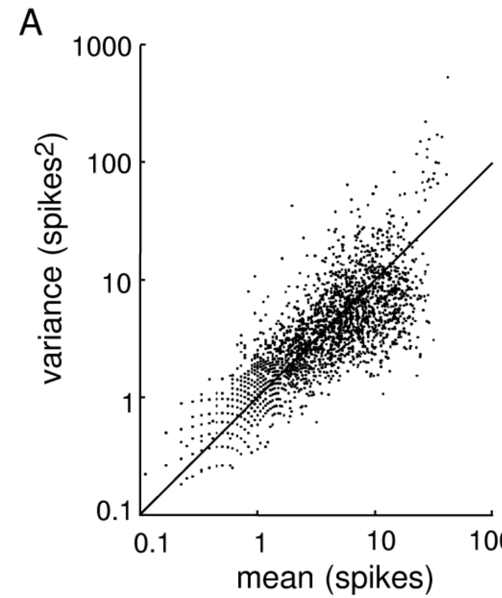
Dayan & Abbott, 2001

Spike count variability

- Properties of Poisson distribution
  - Spike variance and spike count are equal

$$\mu = \sigma^2$$

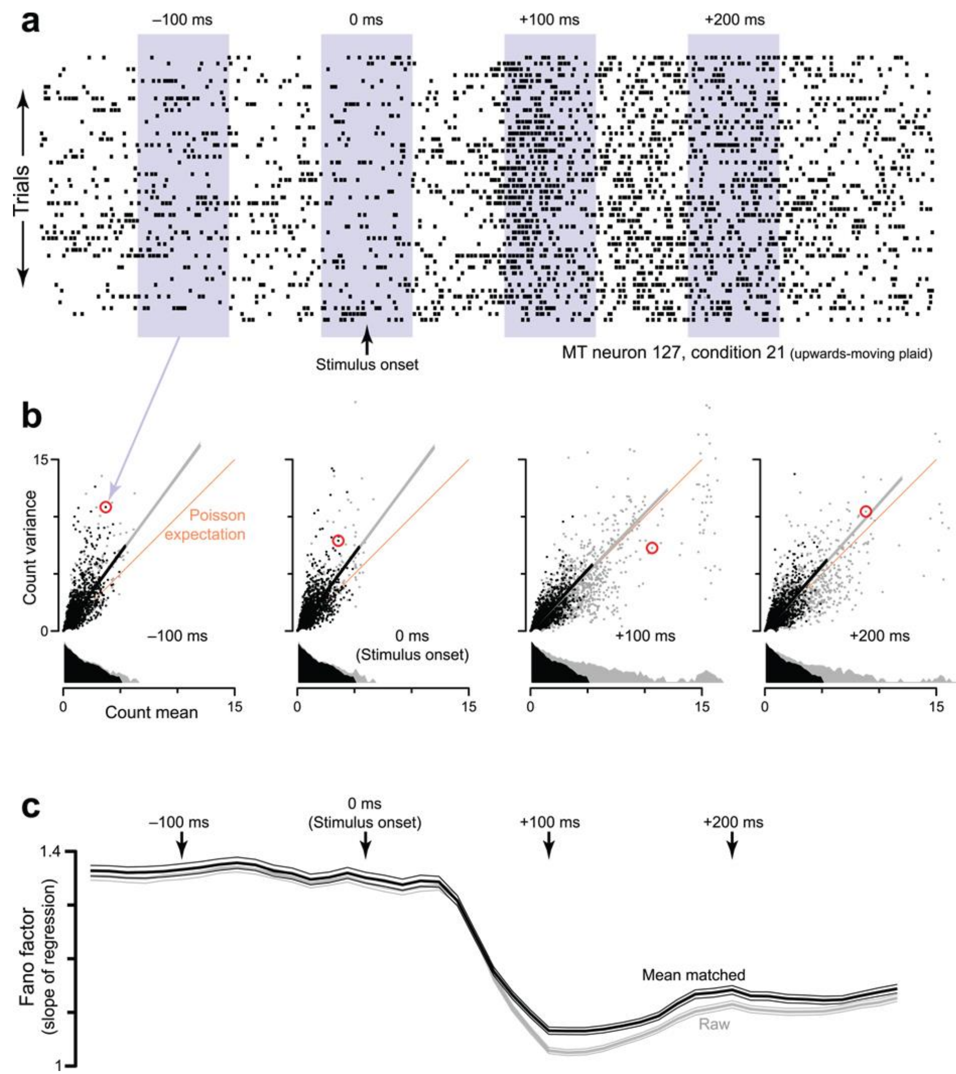
Fano Factor: $C_V = \dfrac{\sigma^2}{\mu}$



  - But this is not always true
  - A linear relationship with slope < 1: Poisson-like

Spike count variability

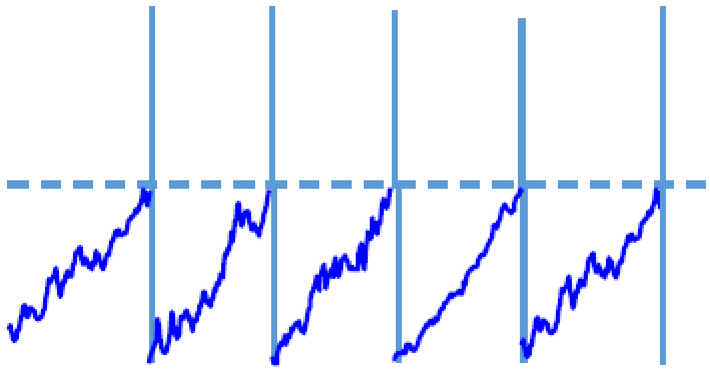
- Spike count variability changes during tasks



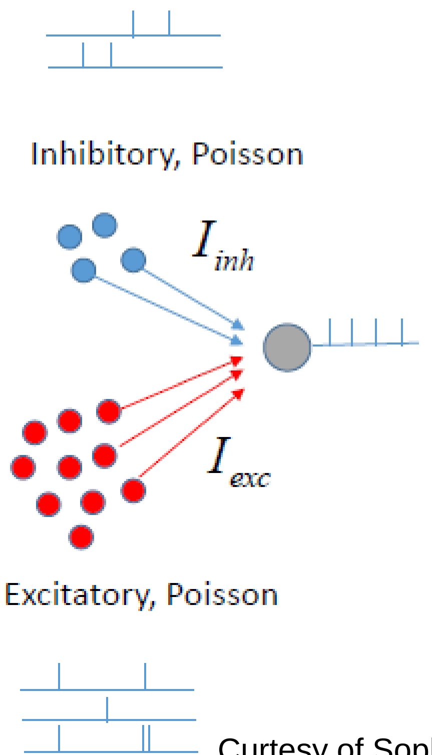
Churchland, et al., 2010

Spike count variability

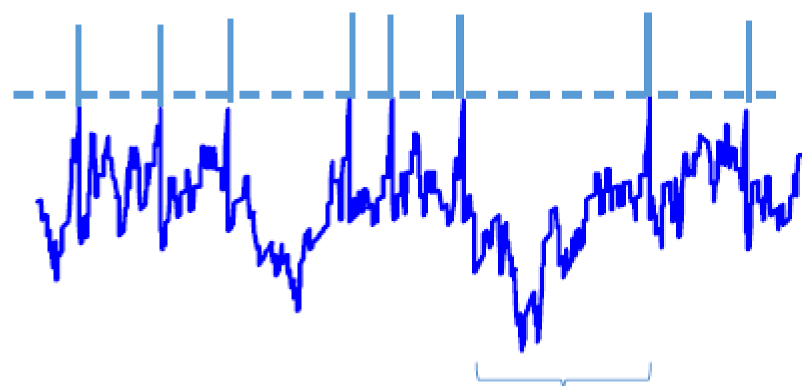
- What is the source of this variability?
  - E/I network: output more regular than input for integrate-and-fire neurons



$$\tau\dot{V} = -V + I_{exc} + I_{inh}$$



  - Balanced E/I: variability is conserved!
  - Balanced networks generate their own variability

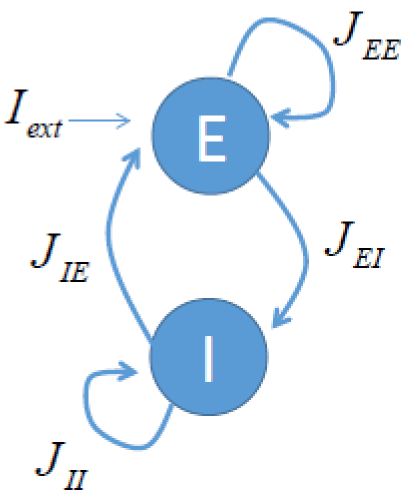


Curtesy of Sophie Denève

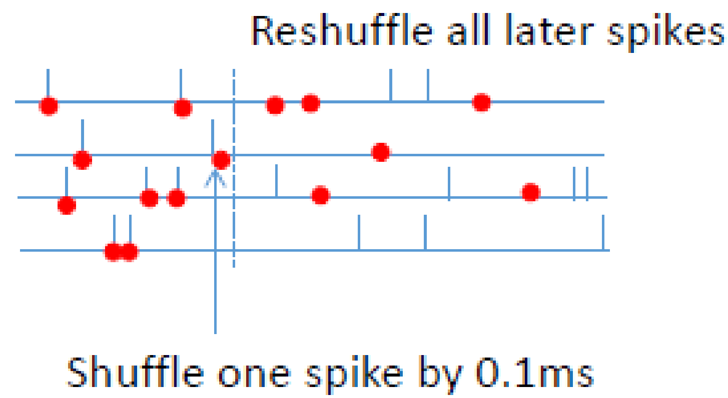
Shadlen & Newsome, 1996

Spike count variability

- Balanced networks generate their own variability
  - Spike timing is not robust
  - Recurrent networks are highly sensitive to initial conditions, even when deterministic



$$J_{EE}v_E - J_{IE}v_I = 0$$
$$J_{EI}v_E - J_{II}v_I = 0$$



- Only firing rates are reproducible from trial to trial!
- The brain relies on large population of redundant unreliable neurons!?

Population coding

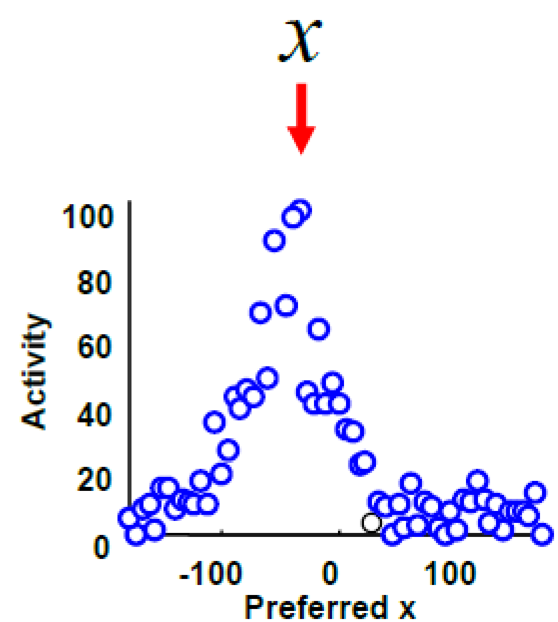
- Many unreliable neurons
  - Decoding: $p(s \mid x) = \prod_i p(s_i \mid x)$
  - Poisson noise:

$$p(s_i \mid x) = \frac{f_i(x)^{s_i} \exp(-f_i(x))}{s_i!}$$

  - Log likelihood function

$$\log(p(s|x)) = \sum_i \log(f_i(x)) s_i - \sum_i f_i(x)$$

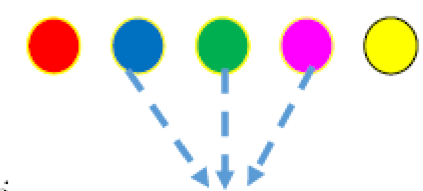
Ma et al. 2006



- Why many unreliable neurons?
  - Efficient population coding!

$$(\mathbf{r}, \Gamma) = \arg\min_{\mathbf{r}^*, \Gamma^*} \left(\|\mathbf{x} - \hat{\mathbf{x}}\|^2 + \mathrm{Cost}(\mathbf{r}^*) \right)$$

$$\hat{x}_i = \sum_j \Gamma_{ij} r_j$$



$\hat{x}$

Linear Decoder

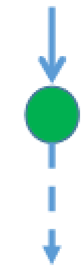
Efficient population coding with spikes

- How do neurons fire at the right time?
- Single neuron
  - When to spike?

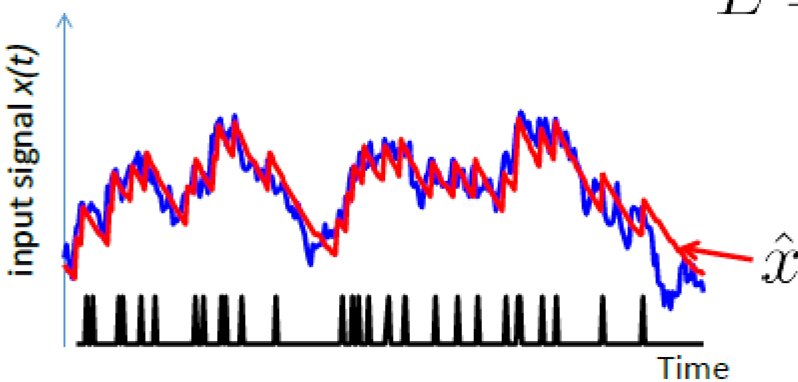
Minimize:

$$E = (x - \hat{x})^2$$

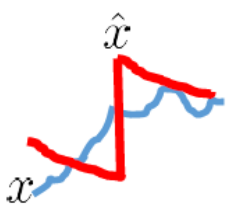
x

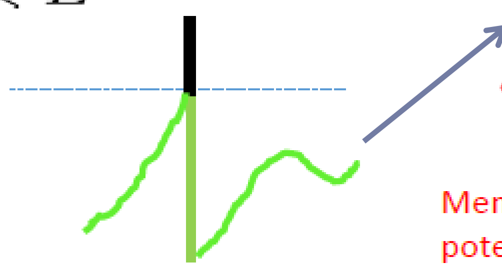


$$\hat{x} = \Gamma r$$



  - Spike when: $E^{\text{spike}} < E^{\text{nospike}}$





$$V = x - \hat{x} > \frac{\Gamma}{2}$$

Membrane potential

Decoding error

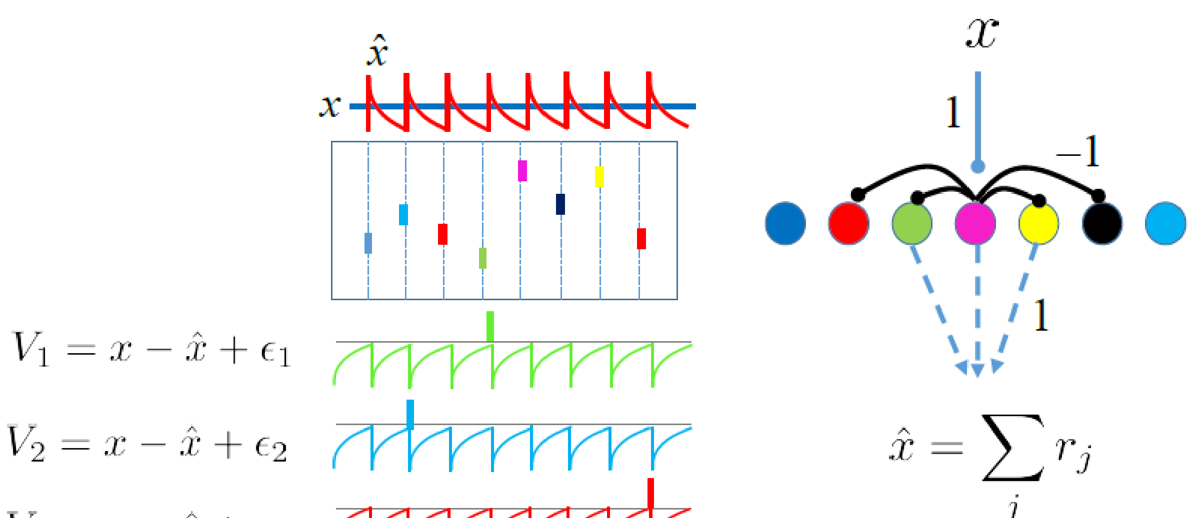
Threshold

Boerlin, Machens, Deneve, 2013

Efficient population coding with spikes

- How do neurons fire at the right time?
- Network of neurons:

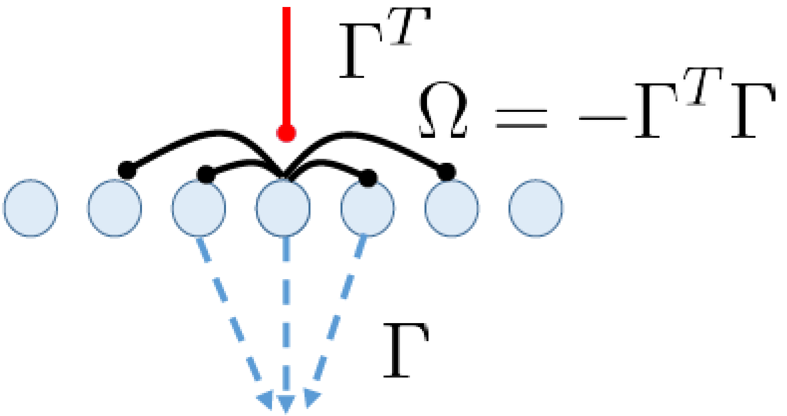
Homogeneous Network



Efficient population coding with spikes

- How do neurons fire at the right time?
- Network of neurons: general case

$$\mathbf{x} = [x_1, \ldots, x_J]$$



$$\hat{x}_i = \sum_j \Gamma_{ij} r_j$$

Minimize:

$$E = \|\mathbf{x} - \hat{\mathbf{x}}\|^2 + \mathrm{Cost}(\mathbf{r})$$

Greedy spike rule:

$$E^{\text{spike } j} < E^{\text{no spike j}}$$

$$\Gamma^T(\mathbf{x} - \hat{\mathbf{x}}) - C(\mathbf{r}) > \frac{\|\Gamma\|^2}{2}$$

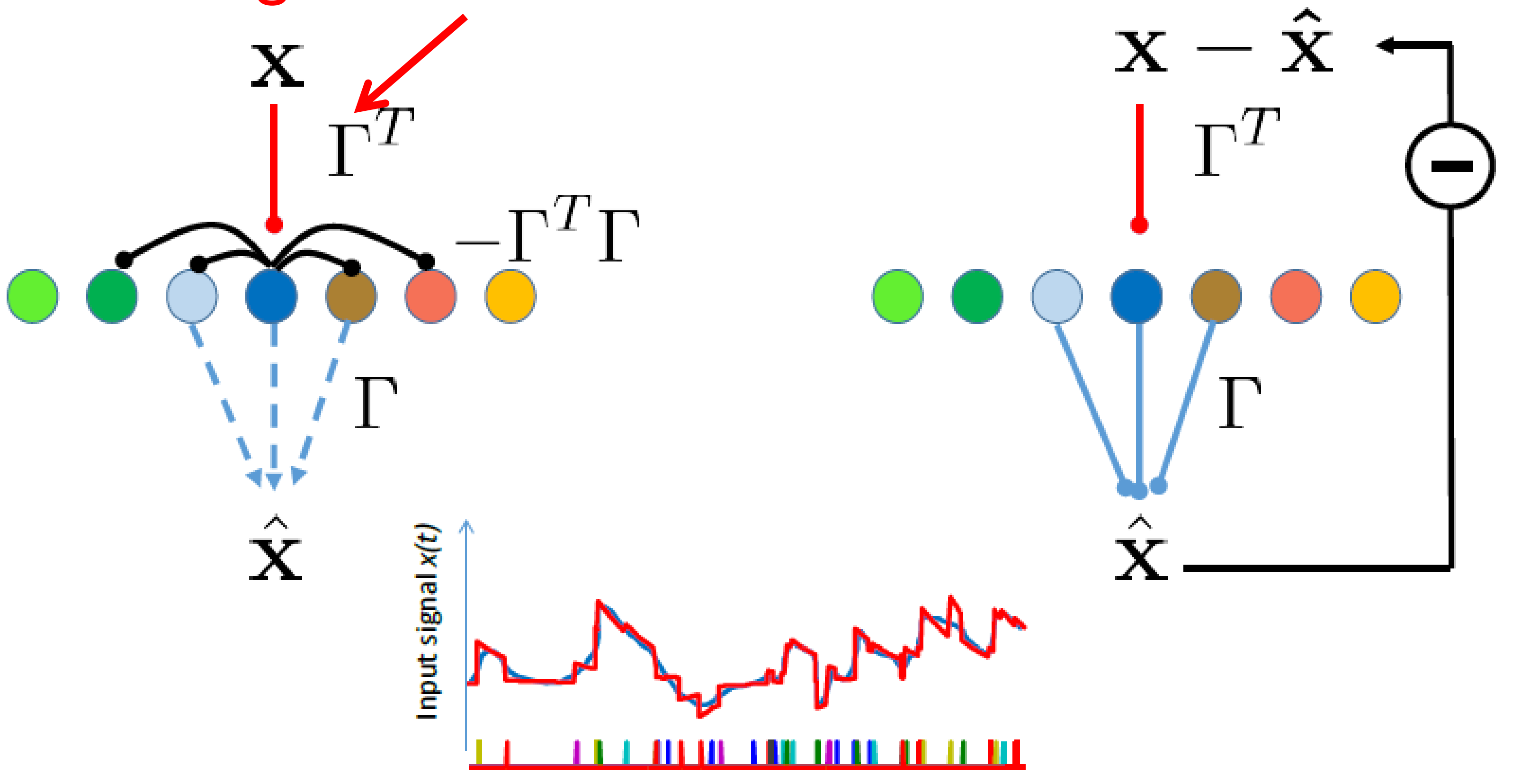
$$V = \Gamma^T(\mathbf{x} - \hat{\mathbf{x}}) - C(\mathbf{r})$$

$$= \Gamma^T\mathbf{x} - \Gamma^T\Gamma\mathbf{r} - C(\mathbf{r})$$

Efficient population coding with spikes

Equivalent to predictive encoder

Optimal weights can also be learned…



Spiking networks

Spiking networks

- Connecting multiple spiking neurons together
  - Coupling through excitatory or inhibitory synapses
  - Interaction between non-linear oscillators
  - Extremely difficulty to analyze

  - Cohen (1982): for weak coupling between oscillators, only their relative phase needs to be considered. Amplitude and waveform of spikes will remain unaffected, but phase relations and frequencies can change.

Spiking networks

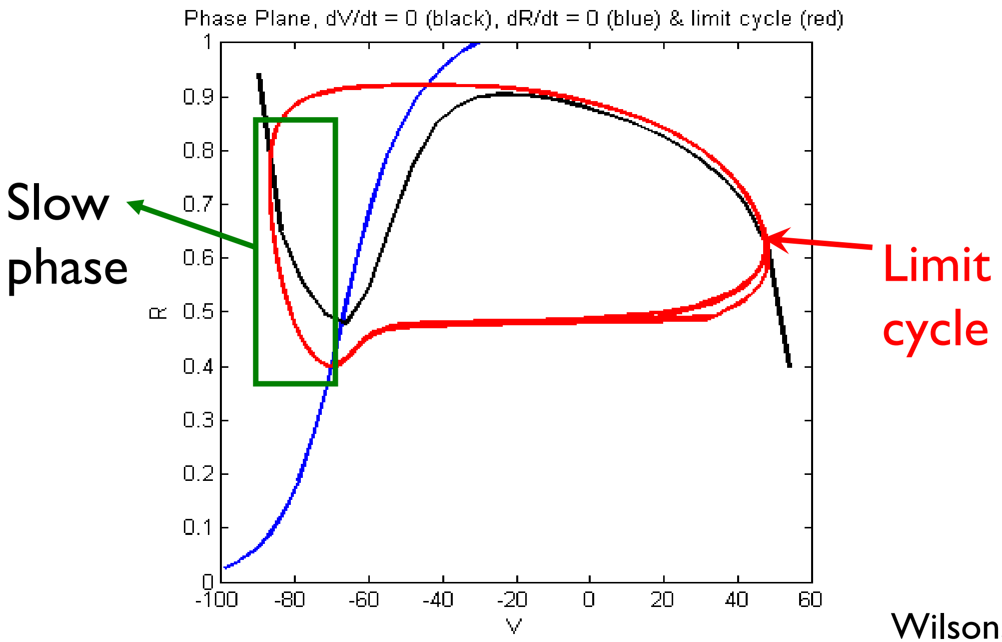
- **Neural oscillations**
  - See Stiefel & Ermentrout (2016) for a good review

$$C \cdot \frac{dV}{dt} = -g_{Na} M(V)^3 (1-R)(V - E_{Na})$$
$$- g_K R^4 (V - E_K) - g_{leak}(V - E_{leak}) + I$$

$$\frac{dR}{dt} = \frac{1}{\tau_R(V)}(-R + G(V))$$

Rinzel's approximation of Hodgkin & Huxley:

$$m \approx m_{eq}(V)$$
$$h \approx 1 - n$$



Wilson, 1999

Recovery variable:
K+ channel opening and Na+ channel closing

Spiking networks

- Simplified oscillator

$$\frac{d\theta}{dt} = \omega \quad so \quad \theta(t) = \omega t \ (\text{mod } 2\pi)$$

- 2 oscillators with synaptic coupling (H)

$$\frac{d\theta_1}{dt} = \omega_1 + H_1(\theta_2 - \theta_1)$$

$$\frac{d\theta_2}{dt} = \omega_2 + H_2(\theta_1 - \theta_2)$$

  - H must be periodic with period 2π

Spiking networks

- Differential oscillation: $\phi=\theta_2-\theta_1$

$$\frac{d\phi}{dt}=\omega_2-\omega_1+H_2(-\phi)-H_1(\phi)$$

  - Phase locking: $\phi= \phi(t)$=constant (e.g. 1:1, 2:1, 3:1 phase locking, …)
  - Synchronization: phase locking with zero phase difference

- Phase locking $$\frac{d\phi}{dt}=0=\omega_2-\omega_1+H_2(-\phi)-H_1(\phi)$$

- Asymptotic stability $$\frac{d}{d\phi}\left[H_2(-\phi)-H_1(\phi)\right]_{\phi_{eq}}<0$$
  - (Jacobian)

Spiking networks

- Simplest example (Cohen 1982) $H_i(\phi) = a_i \cdot \sin(\phi + \sigma)$

$$\frac{d\phi}{dt} = 0 = \omega_2 - \omega_1 - a_1 \cdot \sin(\phi + \sigma) - a_2 \cdot \sin(\phi - \sigma)$$

$$\Leftrightarrow 0 = \omega_2 - \omega_1 - (a_1 + a_2) \cdot \cos(\sigma)\sin(\phi) + (a_2 - a_1) \cdot \sin(\sigma)\cos(\phi)$$

$$\& \frac{d}{d\phi}\left[H_2(-\phi) - H_1(\phi)\right]_{\phi_{eq}} = -(a_1 + a_2) \cdot \cos(\sigma)\cos(\phi) - (a_2 - a_1) \cdot \sin(\sigma)\sin(\phi) < 0$$

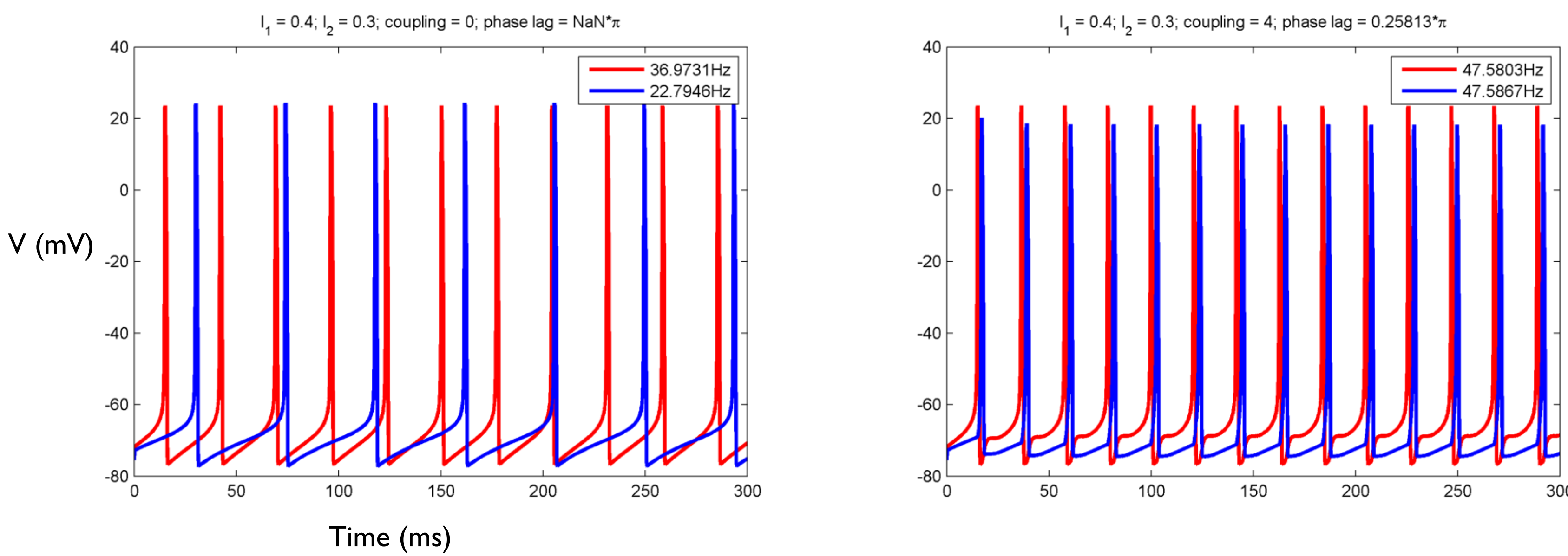
- Special case $\omega_2 = \omega_1 = \omega$
  - Solution $\phi = \arctan\left(\frac{(a_2 - a_1)\sin(\sigma)}{(a_2 + a_1)\cos(\sigma)}\right)$

Spiking networks

- Special case $\omega_2 = \omega_1 = \omega$
  - Solution $\phi = \arctan\left(\dfrac{(a_2 - a_1)\sin(\sigma)}{(a_2 + a_1)\cos(\sigma)}\right)$
  - Assuming
    - synaptic delay: $0 < \sigma < \pi/2$
    - And excitatory synapses: $a_i > 0$
  - 3 particular solutions
    - $\Phi = 0, \pi$ for $a_1 = a_2$
    - $\Phi > 0, \pi$ for $a_1 < a_2$
    - $\Phi < 0, \pi$ for $a_1 > a_2$
  - for $a_1 = a_2$, asymptotic stability condition is $-(a_2 + a_1)\cos(\sigma) < 0$
    - Asymptotically stable steady state; synchronized oscillators

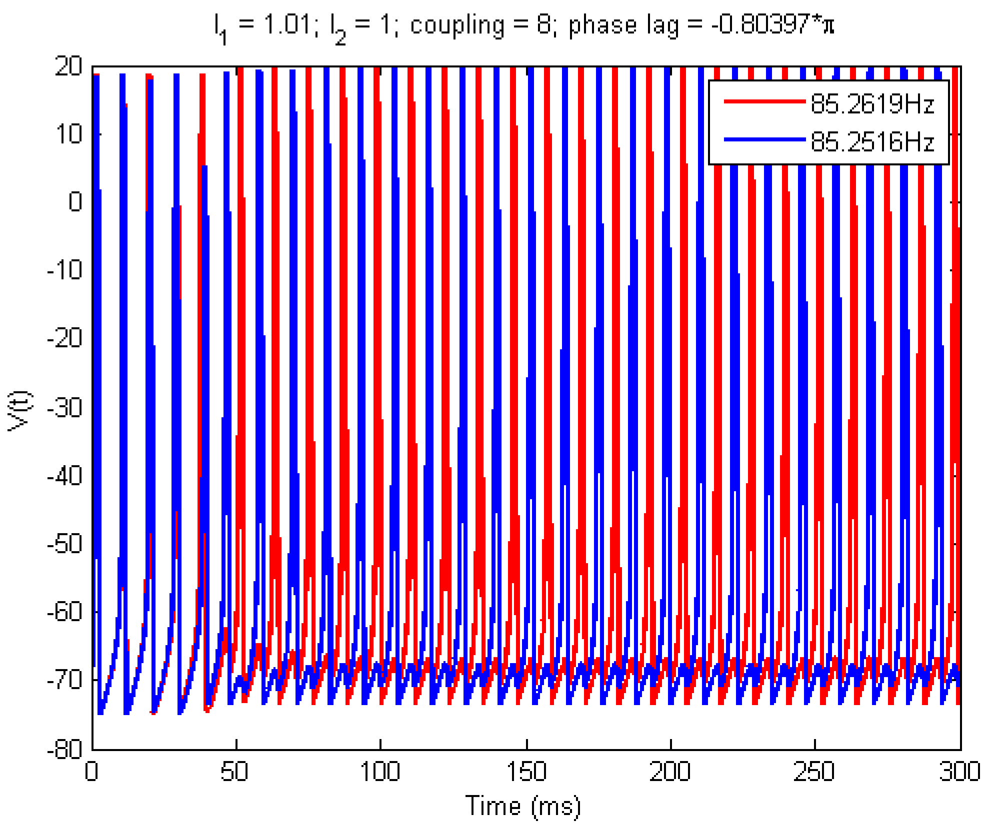
Spiking networks

- Example: 2 excitatory coupled Rinzel neurons



Spiking networks

- Example: 2 inhibitory coupled Rinzel neurons

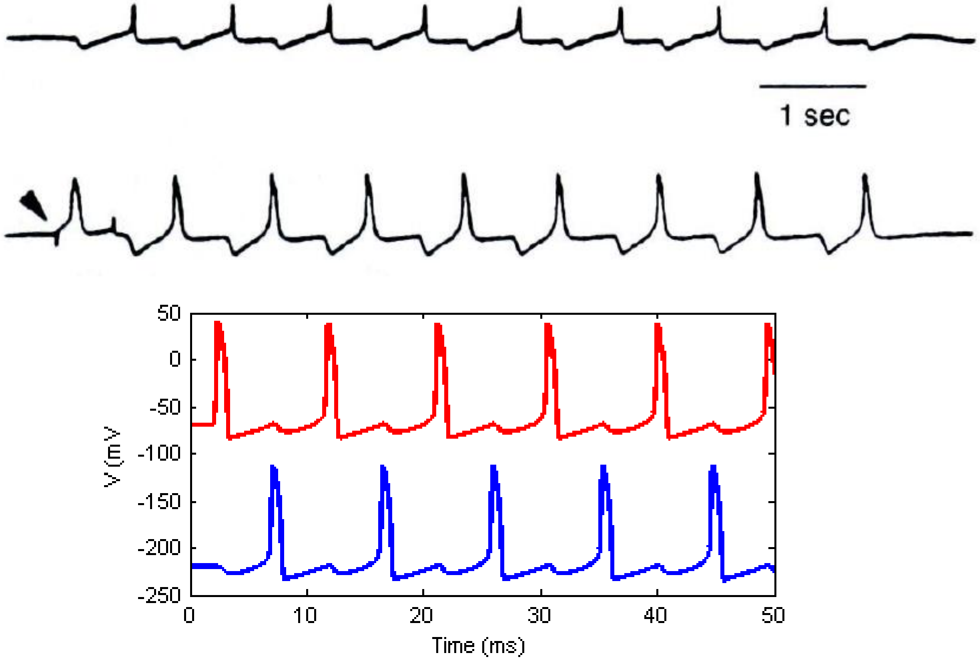


Spiking networks

- Application 1: Clione Limacina swimming

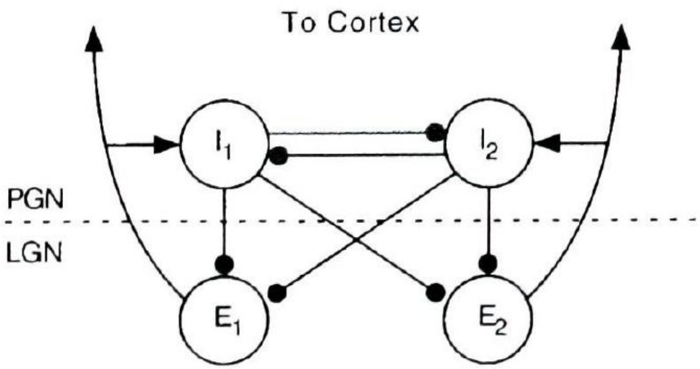
Swimming wing

Swimming controlled by only 2 neurons for dorsal / ventral flexion (Satterlie, 1985).

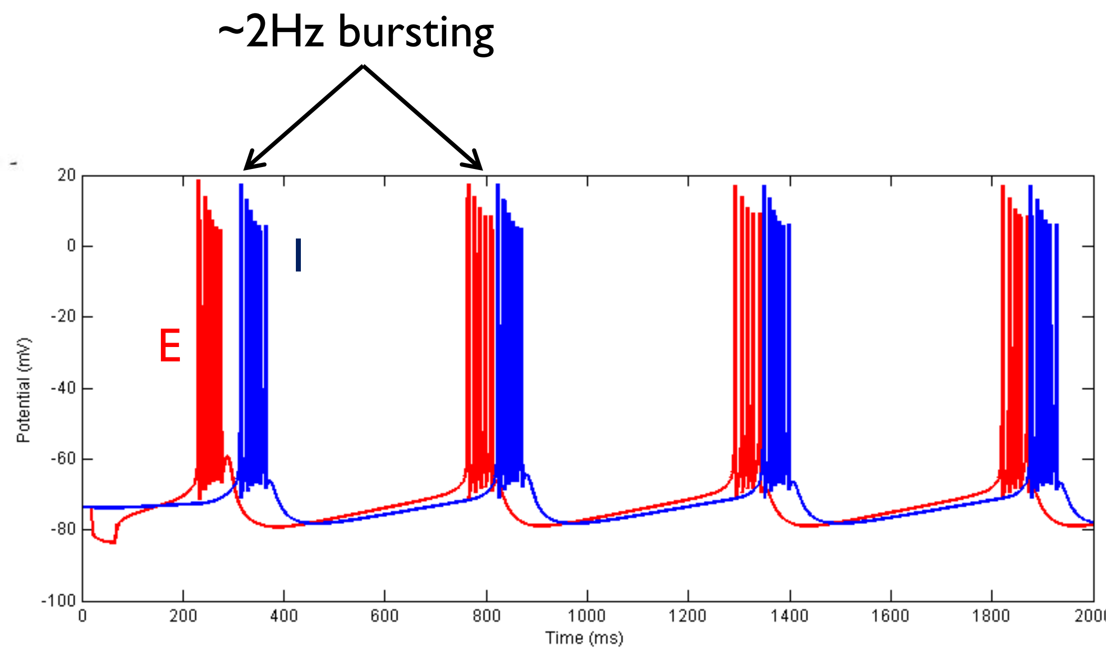


Spiking networks

- Application 2: Thalamic synchronization
  - Bursts of activity during sleep or certain forms of epilepsy
  - LGN-PGN interaction (Lateral Geniculate Nucleus, Peri-Geniculate Nucleus)

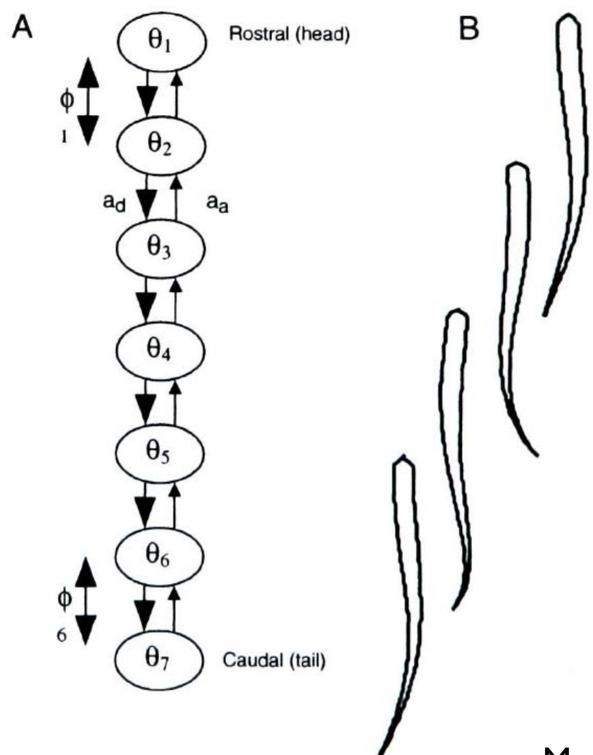


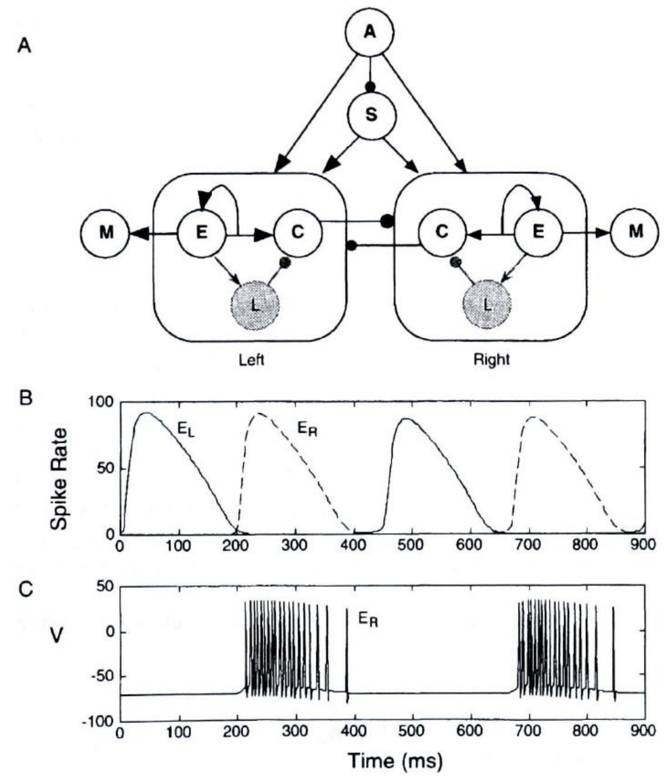
Inhibition > excitation



Spiking networks

- Application 3: swimming (Lamprey) and traveling waves





M: motoneurons
E: excitatory interneurons
C: crossed inhibitory interneurons

A: command neuron
S: serotonin-releasing neuron
L: lateral inhibitory interneurons

Spiking networks

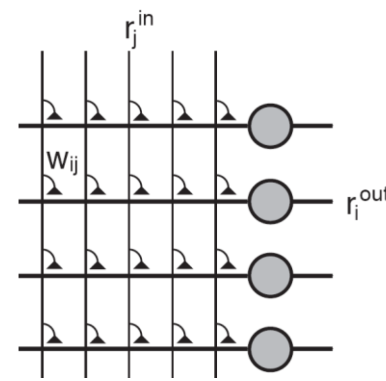
- Limitations on the "design" of spiking nets
  - Network size → unpredictable, emergent properties
  - Neuron types → complex interactions
  - Connectivity → delays, loops, structure…
  - Dynamics → difficulty of formal analysis
- Solution
  - Self-organization → adaptation, learning
  - Many open questions
    - Different concurrent mechanisms
    - Physiological realism
    - Cognitive vs. chemical control of learning…

Hebbian learning

Hebbian learning

- Associative memory
  - Neurons i and j connect with weights w_{ij}
  - Plasticity
    - Structural: change in connection topology of network
    - Functional: change in connection strength (weight)
  - Donald O. Hebb (1949)

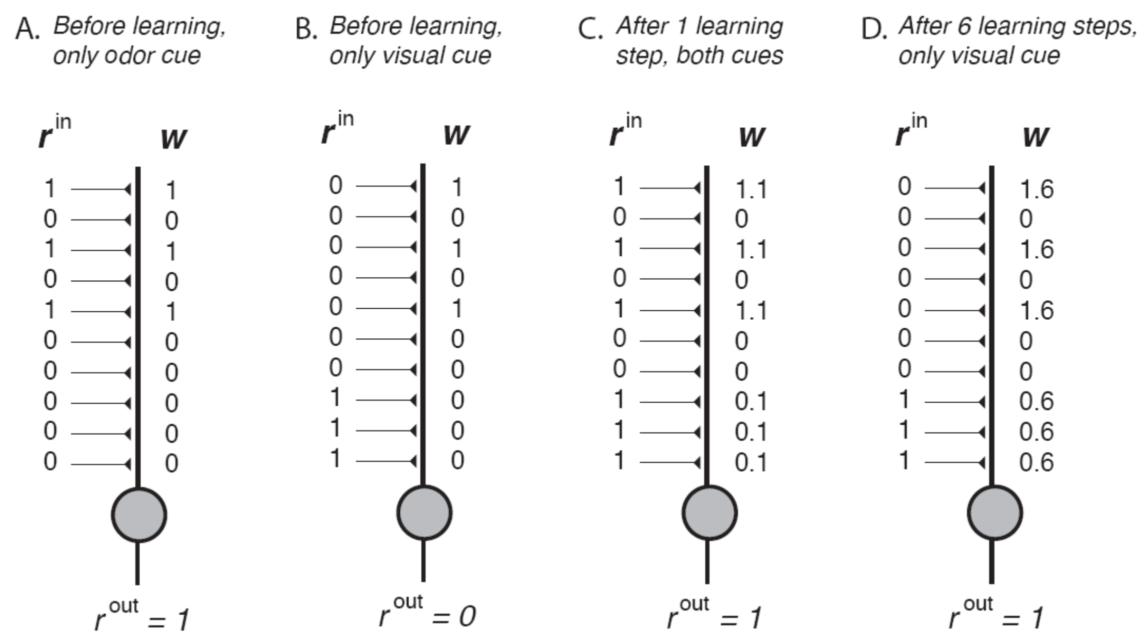
    "When an axon of a cell A is near enough to excite cell B or repeatedly or persistently takes part in firing it, some growth or metabolic change takes place in both cells such that A's efficiency, as one of the cells firing B, is increased."
  - Activity-dependent plasticity that depends on pre- and post-synaptic activity is called Hebbian plasticity / learning



Trappenberg 2010

Hebbian learning

- Associative memory (example)
  - Initially: only odor selective
  - Partial input sufficient (pattern completion)
  - Adaptation: increase w_i by $\Delta w = 0.1$ if a pre-synaptic firing is paired with a post-synaptic firing
  - Learning leads to association



Trappenberg 2010

$$r^{out} = \left(\sum_i \omega_i r_i > \theta \right)$$

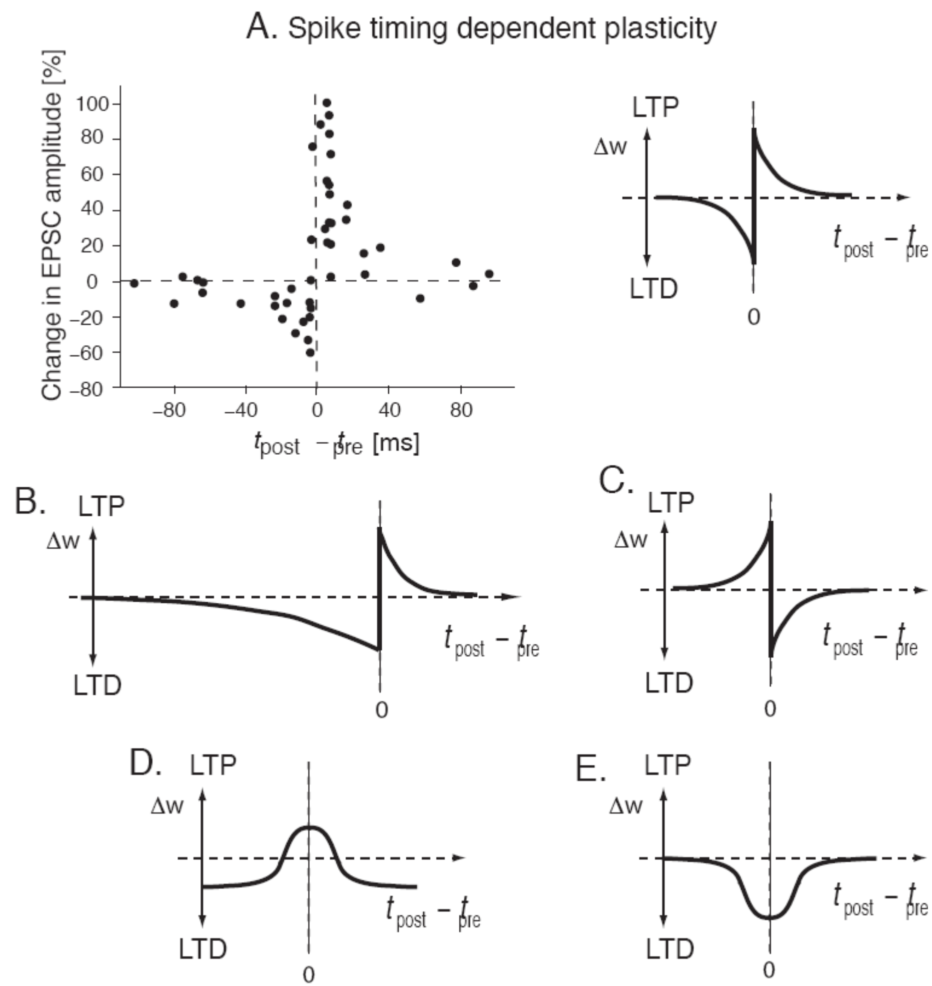
Threshold $\theta = 1.5$

Hebbian learning

- Associative memory features
  - Recall from partial input: pattern completion
    - Relies on distributed stimulus representation
    - Based on (at least) partial overlap (similarity) between stimulus and weight vectors: dot product $\mathbf{r} \cdot \mathbf{w}$ is a measure of similarity
    - Generalization: output node responds to all pattern similar to trained pattern
  - Extraction of central tendencies
    - Weight vector **w** represents average of training set = prototype
    - Noise reduction through the use of prototypes
  - Robustness to degradation
    - Loss or inaccuracy of parts (synapse, neuron) does not affect system
    - This fault tolerance is essential in biological systems

Hebbian learning

- Synaptic plasticity
  - Spike-timing dependent plasticity (STDP)
    - Role of time between pre- and post-synaptic spikes
    - Evokes post-synaptic current EPSC measured (voltage-clamp)
    - Critical time window for plasticity: $|\Delta t| \approx 40$ms
    - Asymmetrical form of Hebbian plasticity



Trappenberg 2010

Hebbian learning

- Mathematical formulation of Hebbian learning

$$w_{ij}(t+\Delta t) = w_{ij}(t) + \Delta w_{ij}(t_i^f, t_j^f; w_{ij})$$

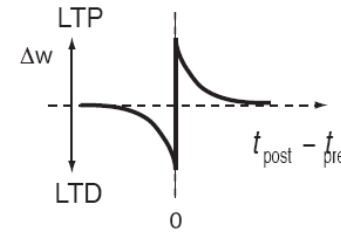
Discrete time step:
Exact time course of synaptic changes unknown…

Weight-dependent plasticity

Hebbian learning

- Mathematical formulation of Hebbian learning
  - Simplest rule for 2 spikes

$$\Delta w_{ij}^{\pm} = \varepsilon^{\pm}(w) \cdot K^{\pm}(t^{post} - t^{pre})$$



Weight-dependent rate
Restricts synaptic efficiencies from "exploding"

$$\varepsilon^{\pm} = \begin{cases} a^{\pm} & for\ w_{ij}^{\min} \le w_{ij} \le w_{ij}^{\max} \\ 0 & otherwise \end{cases}$$

$$\varepsilon^{+} = a^{+} \cdot (w^{\max} - w_{ij})$$

$$\varepsilon^{-} = a^{-} \cdot (w_{ij} - w^{\min})$$

Kernel function

$$K^{\pm}(t^{post} - t^{pre}) = e^{\mp\frac{t^{post} - t^{pre}}{\tau^{\pm}}} \cdot \Theta\left(\pm[t^{post} - t^{pre}]\right)$$

Threshold function restricting LTP and LTD to pos. / neg. domains

Hebbian learning

- Example: Izhikevich network with STDP

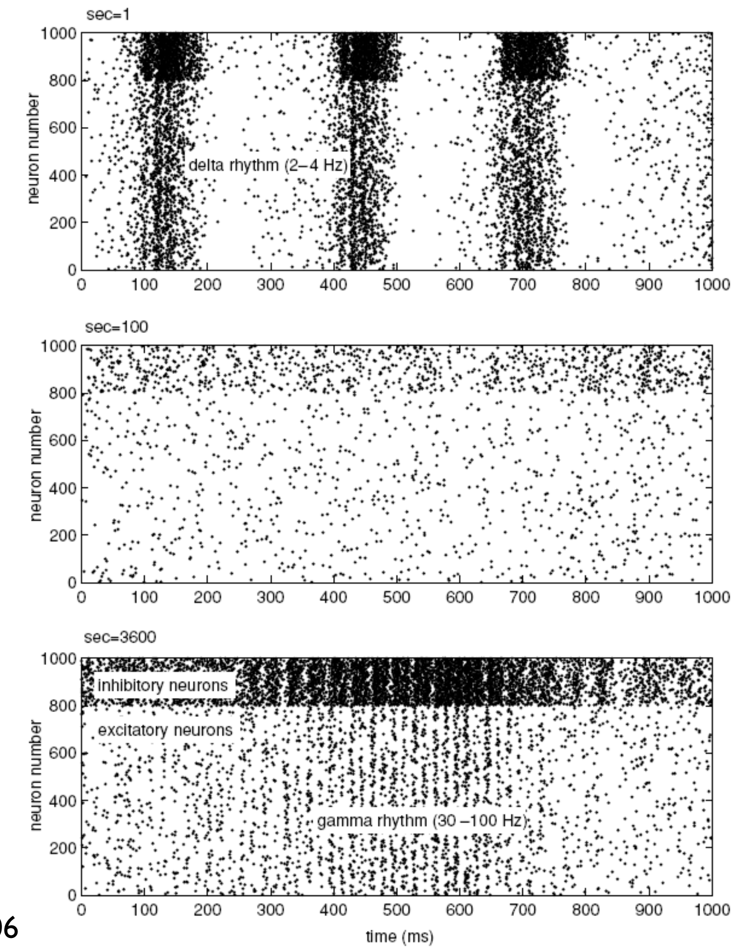
$$\dot{v} = 0.04 \cdot v^2 + 5 \cdot v + 140 - u + I$$

$$\dot{u} = a \cdot (bv - u)$$

$$v(v > 30) = c$$

$$u(v > 30) = u + d$$

$$I_i = \sum_j w_{ij}(v > 30)$$



Izhikevich 2006

Matlab exercises

- Compute the average firing rate of the network. To do so, convolve (use conv.m in Matlab) the spike trains with a narrow Gaussian Kernel (e.g. 2ms SD; use normpdf.m in Matlab) and then sum up the result.
- Compute the frequency content of the average firing rate (use fft.m in Matlab) and analyze how it changes over time, i.e. in 1000ms chunks.
- How do the synaptic weights change over time (from second to second)? Line 105 in spnet.m describes the update (sd) of the synaptic weight (s).
- How does STDP magnitude (line 67) influence the synaptic weight change?
- How do delays (D) in synaptic transmission (line 19) influence the results?
- Try using other neuron types. Examples of parameter values for different neuron types can be found here: http://www.izhikevich.org/publications/whichmod.htm.

Hebbian learning

- Example: Izhikevich network with STDP
  - 4:1 ratio of excitatory : inhibitory cells
  - Inhibitory: fast spiking neurons
  - Excitatory: regular spiking neurons
  - Excitatory neurons have conductance delays between 1-20ms (inhibitory neurons only have 1ms)
  - STDP only affects excitatory neuron synapses
  - STDP decays over time if not reinforced (time constant=20ms)
  - Synaptic depression is stronger than synaptic excitation (to ensure stability of network)

Izhikevich 2006

Further readings

- Feng, Computational Neuroscience: A comprehensive approach. Chapman & Hall/CRC Press, 2004
- Trappenberg, Fundamentals of computational neuroscience, Oxford University Press, 2010
- Wilson, Spikes, decisions and actions, Oxford University Press 1999