COMPUTATIONAL MODELLING OF SYNAPTIC FUNCTION

MÁTÉ LENGYEL

Computational and Biological Learning Lab
Department of Engineering
University of Cambridge

COMPUTATIONAL MODELLING OF SYNAPTIC PLASTICITY

MÁTÉ LENGYEL

Computational and Biological Learning Lab
Department of Engineering
University of Cambridge

SYNAPTIC PLASTICITY

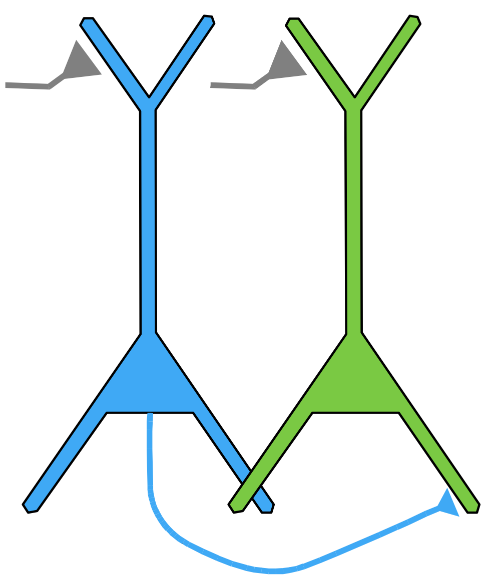
Synapses are computational devices that not only transmit action potential-encoded information, but also transform it. Neuronal information is often encoded by bursts or trains of action potentials. Synapses process such action potential bursts or trains in a synapse-specific manner that involves use-dependent changes in neurotransmitter release during the burst or train (referred to as short-term plasticity). In addition, synapses experience use-dependent long-term changes in synaptic transmission that adjust the "gain" of a synapse, and operate either pre- and/or postsynaptically (referred to as long-term plasticity)

Südhof, 2012

SYNAPTIC PLASTICITY

Synapses are computational devices that not only transmit action potential-encoded information, but also transform it. Neuronal information is often encoded by bursts or trains of action potentials. Synapses process such action potential bursts or trains in a synapse-specific manner that involves use-dependent changes in neurotransmitter release during the burst or train (referred to as short-term plasticity). In addition, synapses experience use-dependent long-term changes in synaptic transmission that adjust the “gain” of a synapse, and operate either pre- and/or postsynaptically (referred to as long-term plasticity)

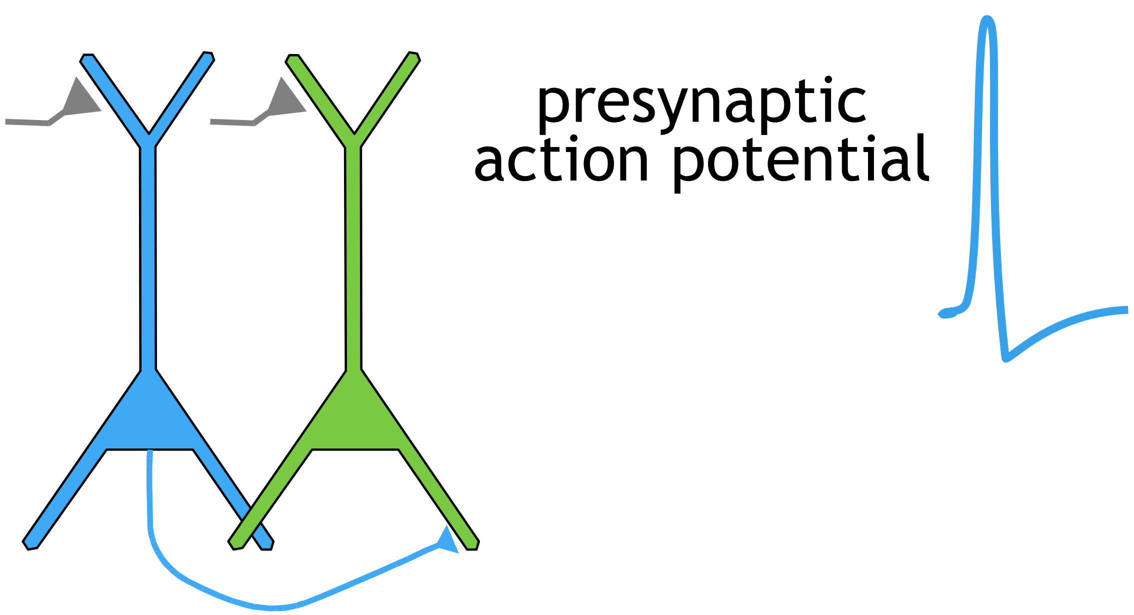
Südhof, 2012



SYNAPTIC PLASTICITY

Synapses are computational devices that not only transmit action potential-encoded information, but also transform it. Neuronal information is often encoded by bursts or trains of action potentials. Synapses process such action potential bursts or trains in a synapse-specific manner that involves use-dependent changes in neurotransmitter release during the burst or train (referred to as short-term plasticity). In addition, synapses experience use-dependent long-term changes in synaptic transmission that adjust the "gain" of a synapse, and operate either pre- and/or postsynaptically (referred to as long-term plasticity)

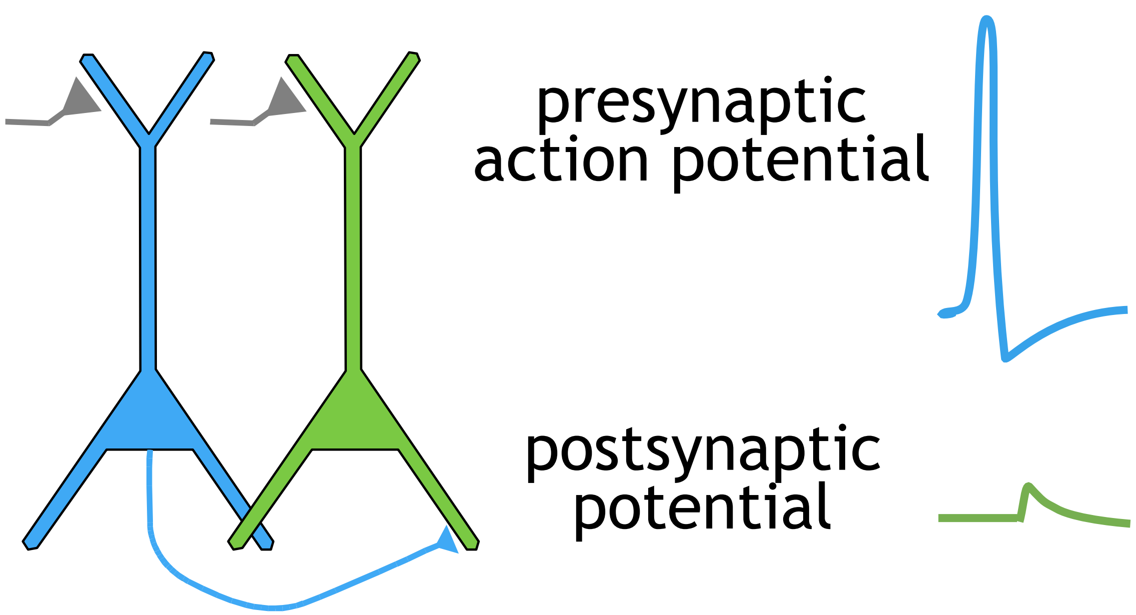
Südhof, 2012



SYNAPTIC PLASTICITY

Synapses are computational devices that not only transmit action potential-encoded information, but also transform it. Neuronal information is often encoded by bursts or trains of action potentials. Synapses process such action potential bursts or trains in a synapse-specific manner that involves use-dependent changes in neurotransmitter release during the burst or train (referred to as short-term plasticity). In addition, synapses experience use-dependent long-term changes in synaptic transmission that adjust the “gain” of a synapse, and operate either pre- and/or postsynaptically (referred to as long-term plasticity)

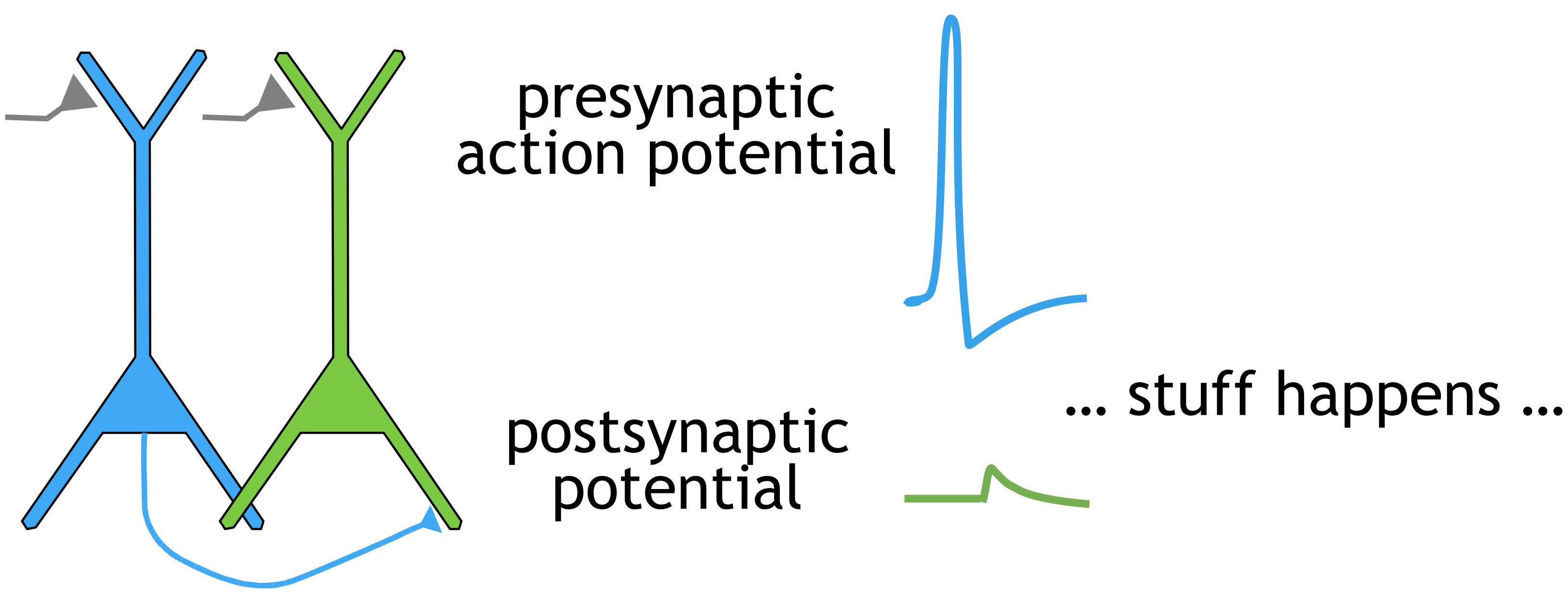
Südhof, 2012



SYNAPTIC PLASTICITY

Synapses are computational devices that not only transmit action potential-encoded information, but also transform it. Neuronal information is often encoded by bursts or trains of action potentials. Synapses process such action potential bursts or trains in a synapse-specific manner that involves use-dependent changes in neurotransmitter release during the burst or train (referred to as short-term plasticity). In addition, synapses experience use-dependent long-term changes in synaptic transmission that adjust the "gain" of a synapse, and operate either pre- and/or postsynaptically (referred to as long-term plasticity)

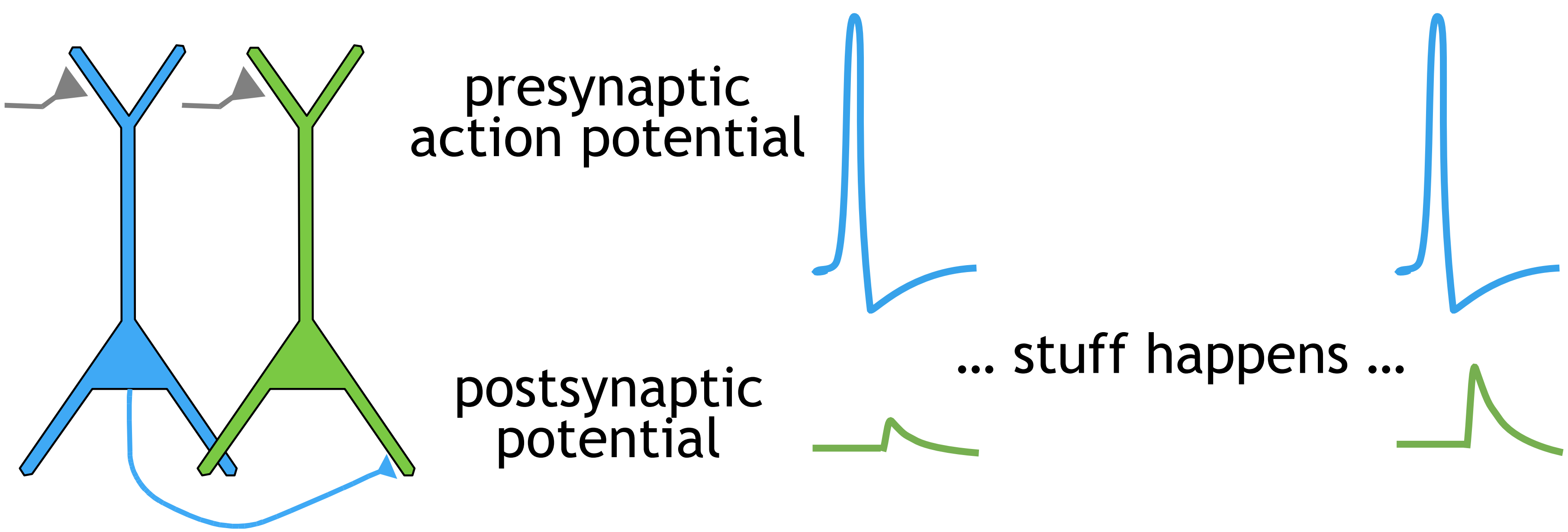
Südhof, 2012



SYNAPTIC PLASTICITY

Synapses are computational devices that not only transmit action potential-encoded information, but also transform it. Neuronal information is often encoded by bursts or trains of action potentials. Synapses process such action potential bursts or trains in a synapse-specific manner that involves use-dependent changes in neurotransmitter release during the burst or train (referred to as short-term plasticity). In addition, synapses experience use-dependent long-term changes in synaptic transmission that adjust the "gain" of a synapse, and operate either pre- and/or postsynaptically (referred to as long-term plasticity)

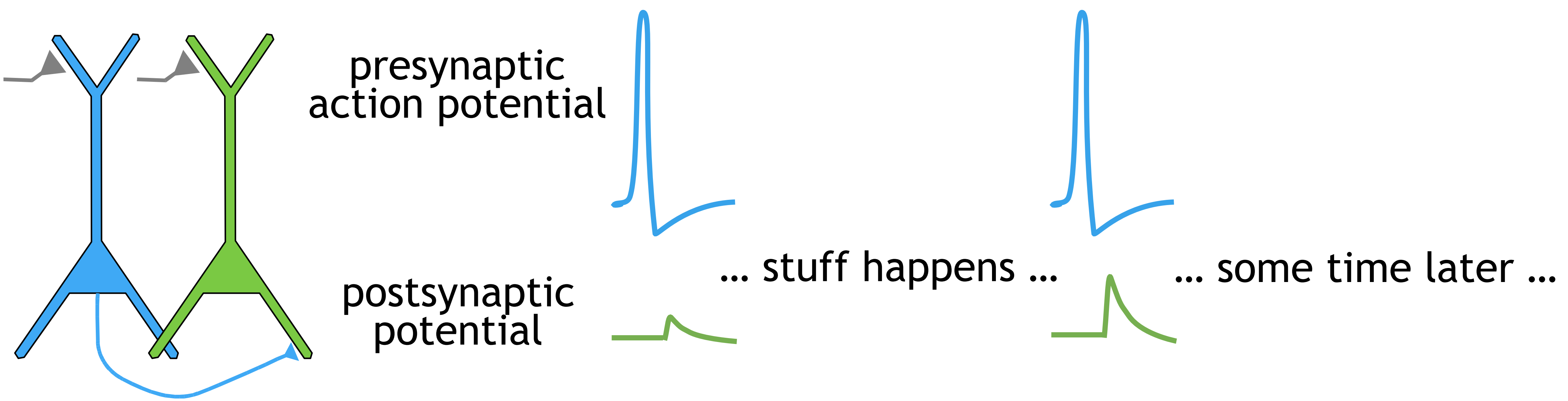
Südhof, 2012



SYNAPTIC PLASTICITY

Synapses are computational devices that not only transmit action potential-encoded information, but also transform it. Neuronal information is often encoded by bursts or trains of action potentials. Synapses process such action potential bursts or trains in a synapse-specific manner that involves use-dependent changes in neurotransmitter release during the burst or train (referred to as short-term plasticity). In addition, synapses experience use-dependent long-term changes in synaptic transmission that adjust the "gain" of a synapse, and operate either pre- and/or postsynaptically (referred to as long-term plasticity)

Südhof, 2012



SYNAPTIC PLASTICITY

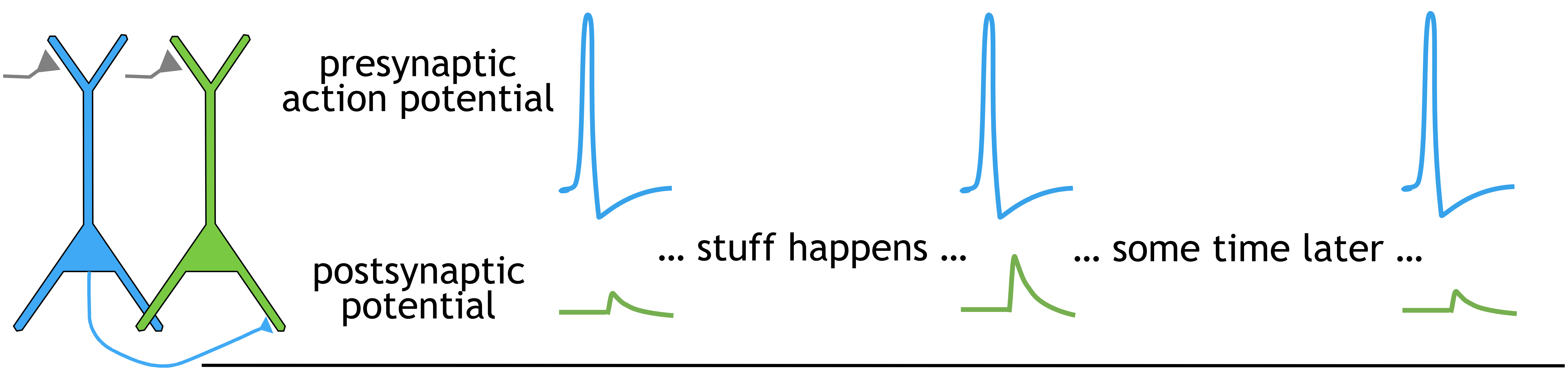
Synapses are computational devices that not only transmit action potential-encoded information, but also transform it. Neuronal information is often encoded by bursts or trains of action potentials. Synapses process such action potential bursts or trains in a synapse-specific manner that involves use-dependent changes in neurotransmitter release during the burst or train (referred to as short-term plasticity). In addition, synapses experience use-dependent long-term changes in synaptic transmission that adjust the "gain" of a synapse, and operate either pre- and/or postsynaptically (referred to as long-term plasticity)

Südhof, 2012

SYNAPTIC PLASTICITY

Synapses are computational devices that not only transmit action potential-encoded information, but also transform it. Neuronal information is often encoded by bursts or trains of action potentials. Synapses process such action potential bursts or trains in a synapse-specific manner that involves use-dependent changes in neurotransmitter release during the burst or train (referred to as short-term plasticity). In addition, synapses experience use-dependent long-term changes in synaptic transmission that adjust the "gain" of a synapse, and operate either pre- and/or postsynaptically (referred to as long-term plasticity)

Südhof, 2012



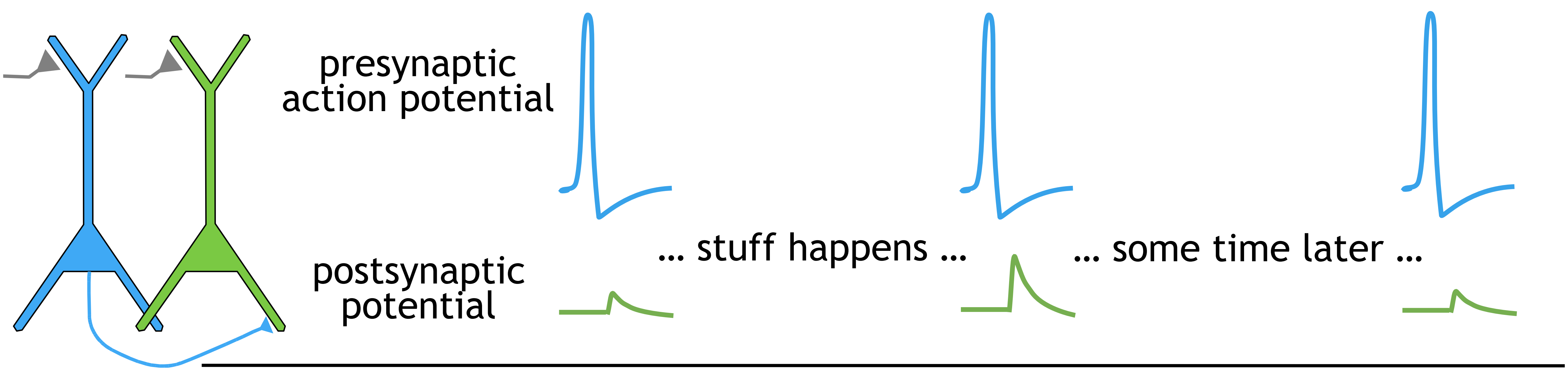
short-term plasticity

long-term plasticity

SYNAPTIC PLASTICITY

Synapses are computational devices that not only transmit action potential-encoded information, but also transform it. Neuronal information is often encoded by bursts or trains of action potentials. Synapses process such action potential bursts or trains in a synapse-specific manner that involves use-dependent changes in neurotransmitter release during the burst or train (referred to as short-term plasticity). In addition, synapses experience use-dependent long-term changes in synaptic transmission that adjust the "gain" of a synapse, and operate either pre- and/or postsynaptically (referred to as long-term plasticity)

Südhof, 2012

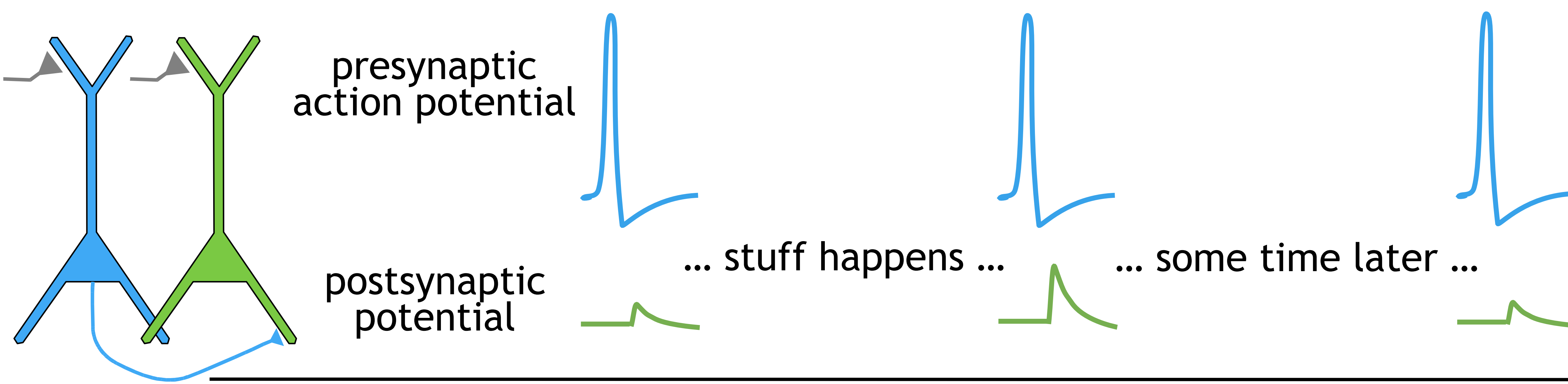


short-term plasticity	*10 — 100 ms*
long-term plasticity	*10 min —*

SYNAPTIC PLASTICITY

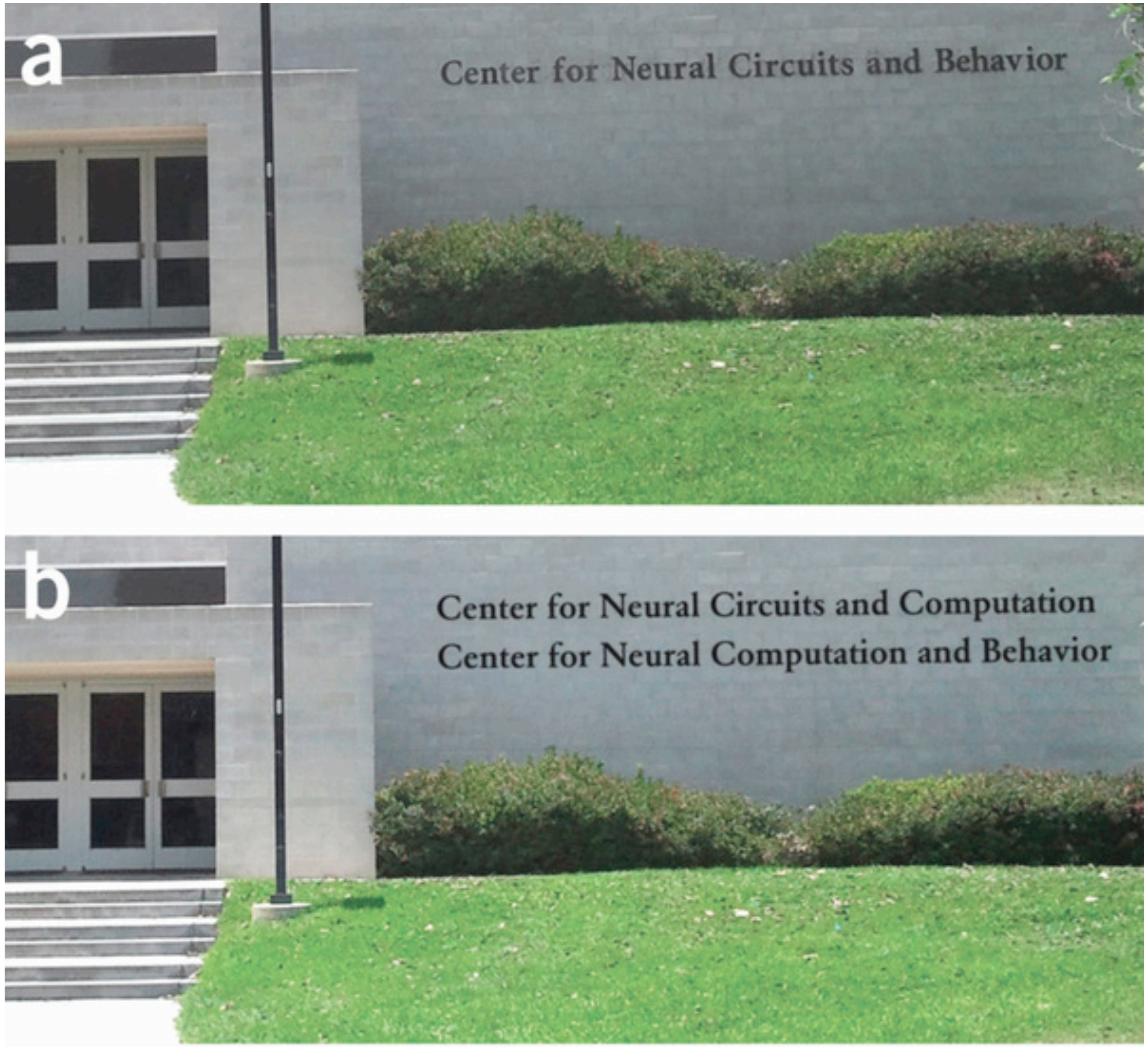
Synapses are computational devices that not only transmit action potential-encoded information, but also transform it. Neuronal information is often encoded by bursts or trains of action potentials. Synapses process such action potential bursts or trains in a synapse-specific manner that involves use-dependent changes in neurotransmitter release during the burst or train (referred to as short-term plasticity). In addition, synapses experience use-dependent long-term changes in synaptic transmission that adjust the "gain" of a synapse, and operate either pre- and/or postsynaptically (referred to as long-term plasticity)

Südhof, 2012



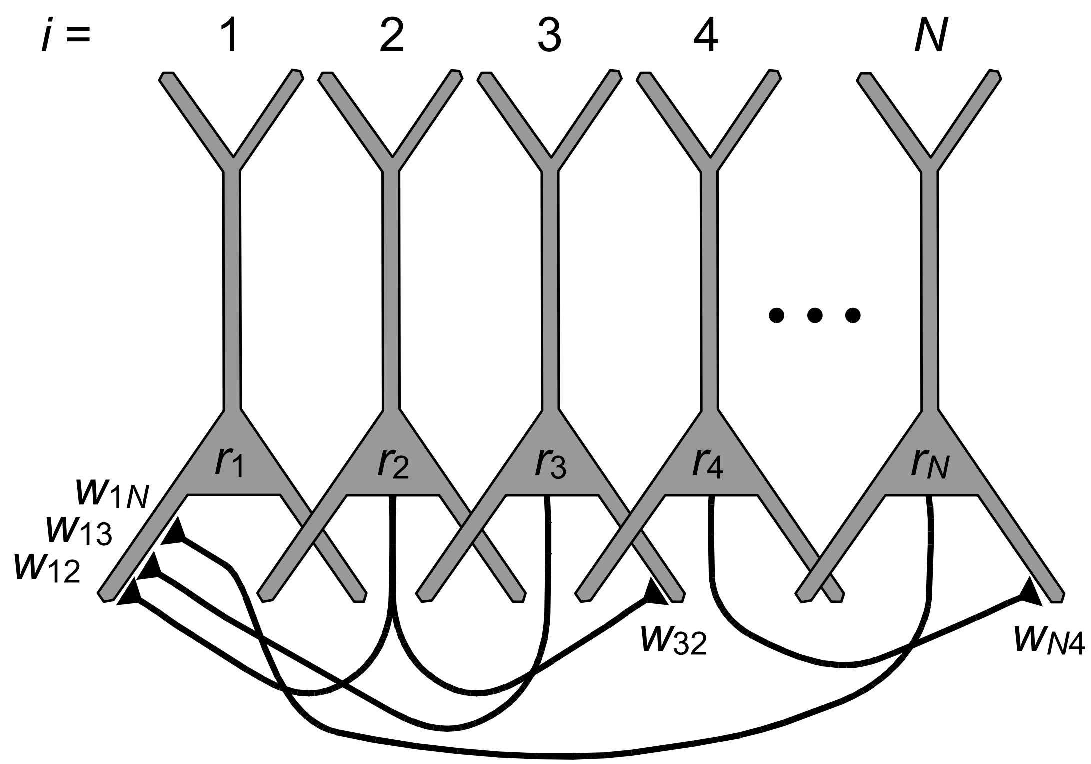
short-term plasticity	*presynaptically*	*10 — 100 ms*
long-term plasticity	*pre- and postsynaptically*	*10 min —*

COMPUTATION: BETWEEN CIRCUITS AND BEHAVIOUR

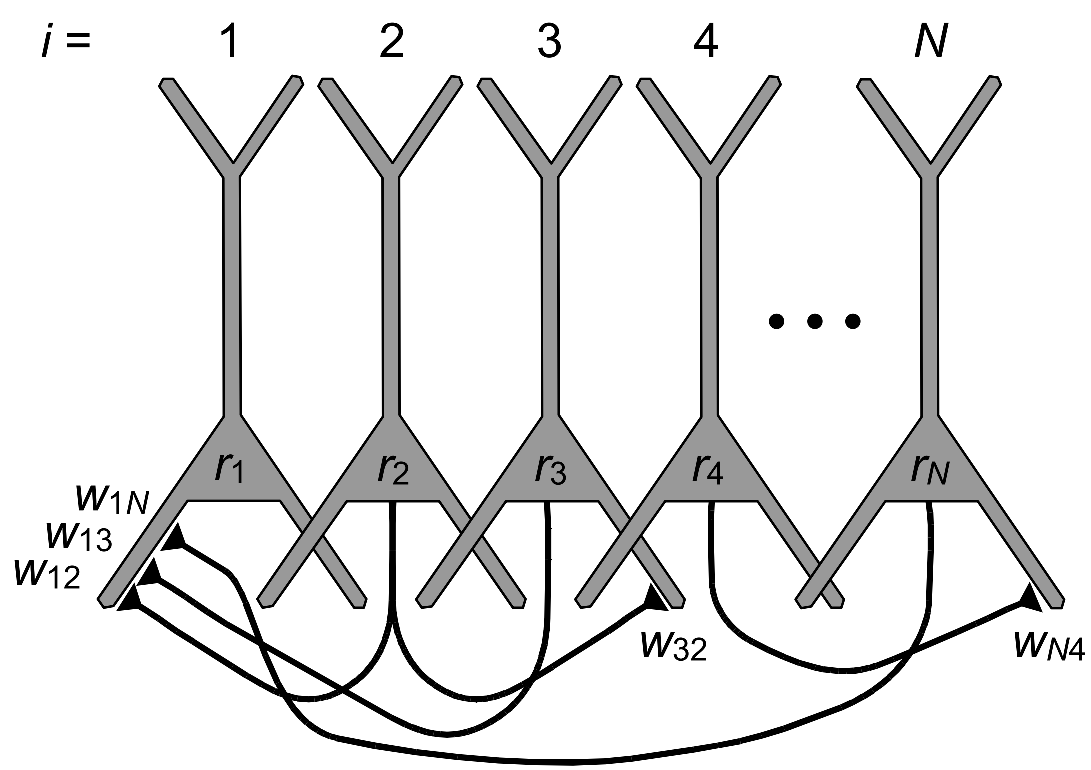


Carandini, 2012

TWO LEVELS OF NETWORK DYNAMICS



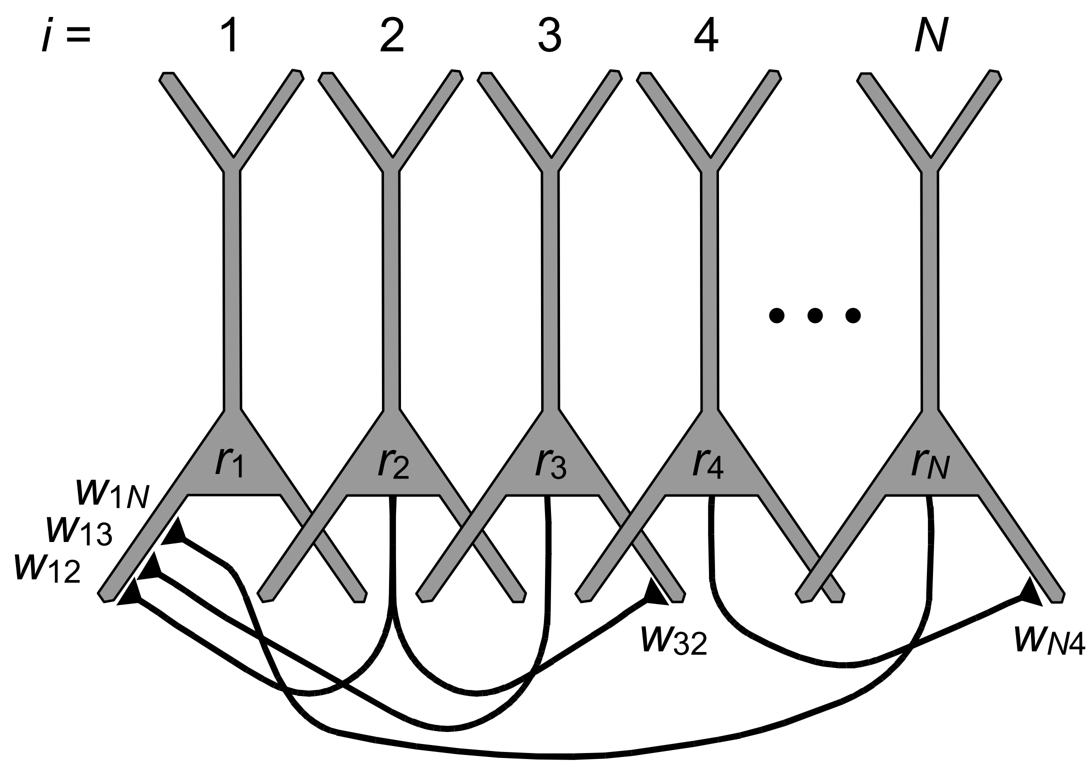
TWO LEVELS OF NETWORK DYNAMICS

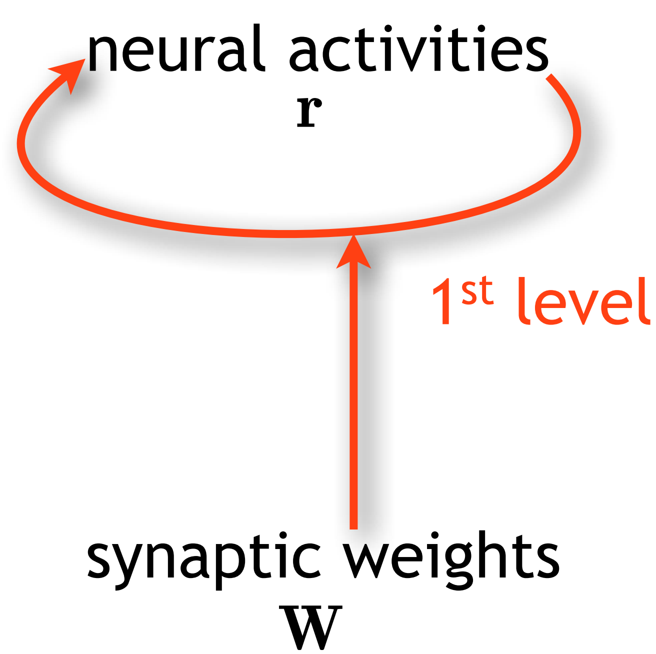


neural activities
$\mathbf{r}$

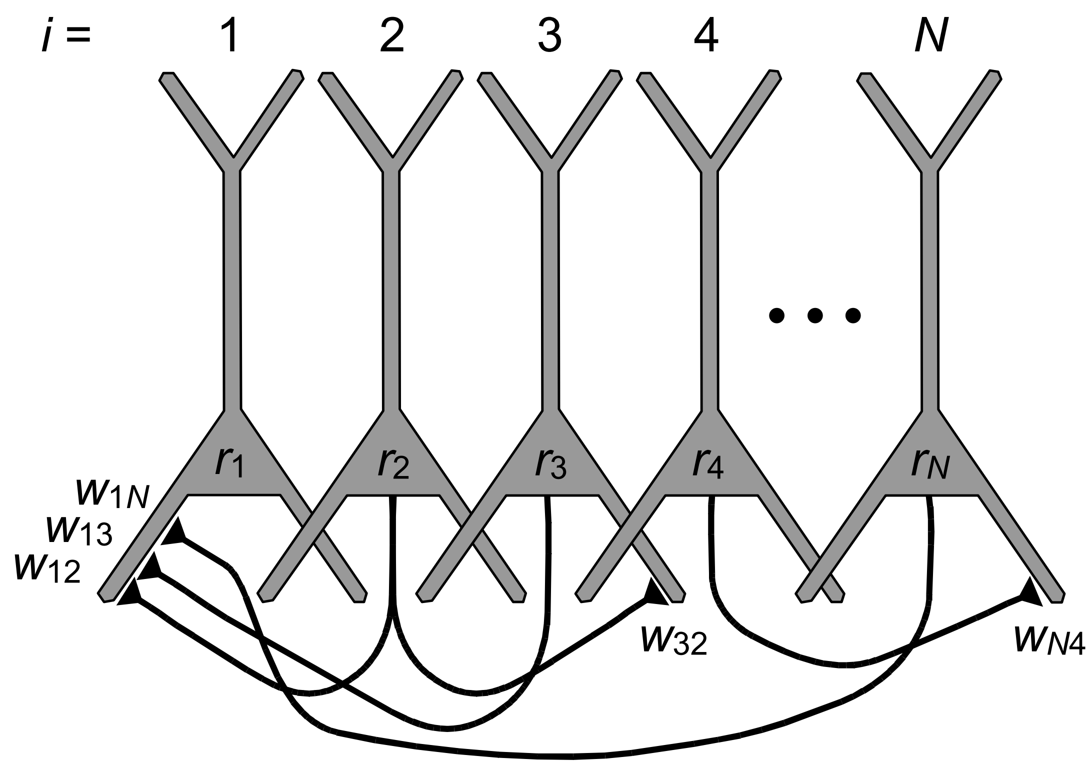
synaptic weights
$\mathbf{W}$

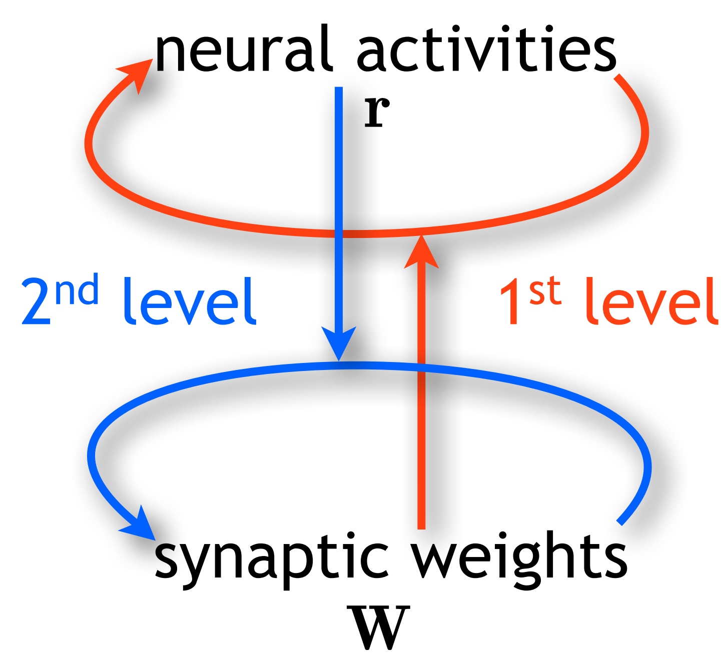
TWO LEVELS OF NETWORK DYNAMICS





TWO LEVELS OF NETWORK DYNAMICS





AUTOASSOCIATIVE MEMORY: AN EXAMPLE

I raised to my lips a spoonful of the tea in which I had soaked a morsel of the cake. ... And suddenly the memory returns. The taste was that of the little crumb of madeleine which on Sunday mornings at Combray, when I went to say good day to her in her bedroom, my aunt Léonie used to give me, dipping it first in her own cup of real or of lime-flower tea.

Marcel Proust: À la recherche du temps perdu

AUTOASSOCIATIVE MEMORY: AN EXAMPLE

I raised to my lips a spoonful of the tea in which I had soaked a morsel of the cake. ... And suddenly the memory returns. The taste was that of the little crumb of madeleine which on Sunday mornings at Combray, when I went to say good day to her in her bedroom, my aunt Léonie used to give me, dipping it first in her own cup of real or of lime-flower tea.

Marcel Proust: À la recherche du temps perdu

HOW DOES THIS HAPPEN?

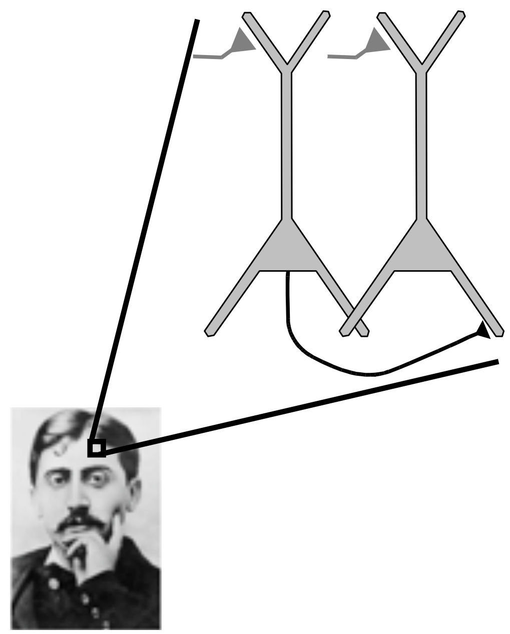
MEMORY PROCESSING IN NEURAL NETWORKS

MEMORY PROCESSING IN NEURAL NETWORKS

the Hebbian paradigm

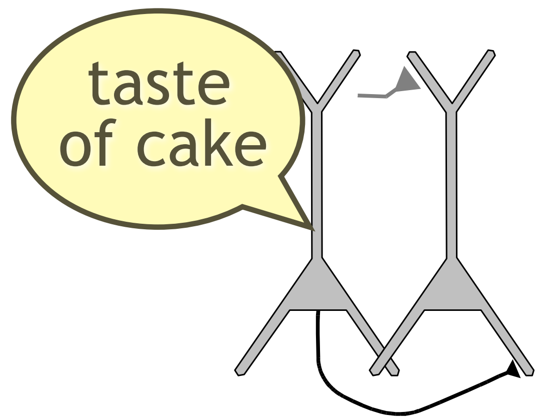
MEMORY PROCESSING IN NEURAL NETWORKS

the Hebbian paradigm



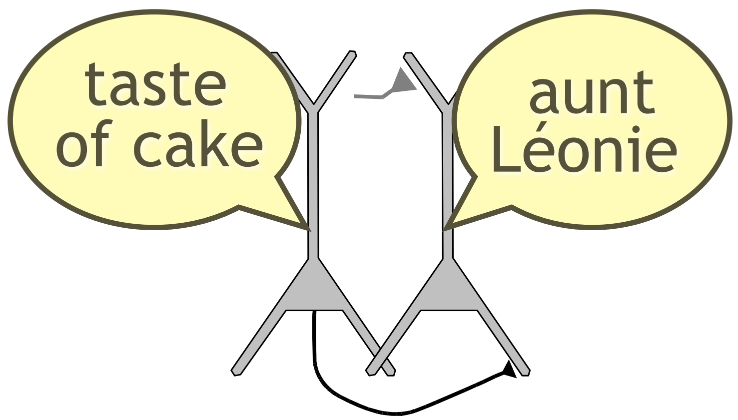
MEMORY PROCESSING IN NEURAL NETWORKS

the Hebbian paradigm



MEMORY PROCESSING IN NEURAL NETWORKS

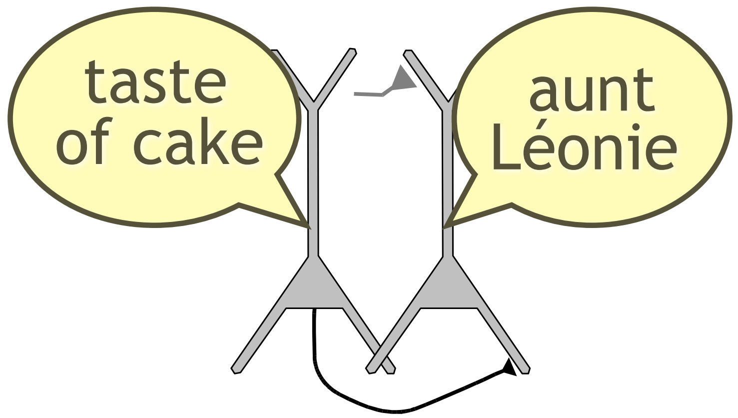
the Hebbian paradigm



MEMORY PROCESSING IN NEURAL NETWORKS

the Hebbian paradigm

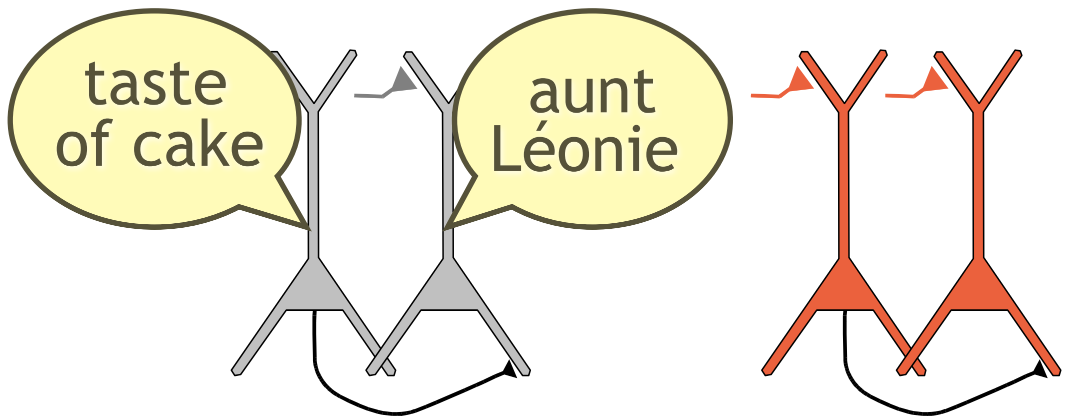
storage



MEMORY PROCESSING IN NEURAL NETWORKS

the Hebbian paradigm

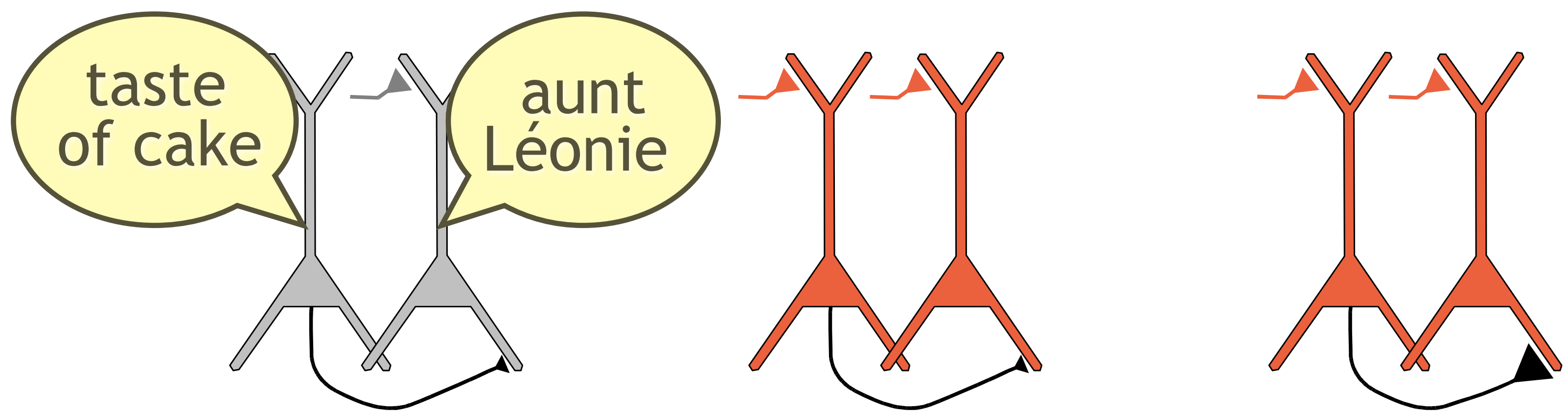
storage



MEMORY PROCESSING IN NEURAL NETWORKS

the Hebbian paradigm

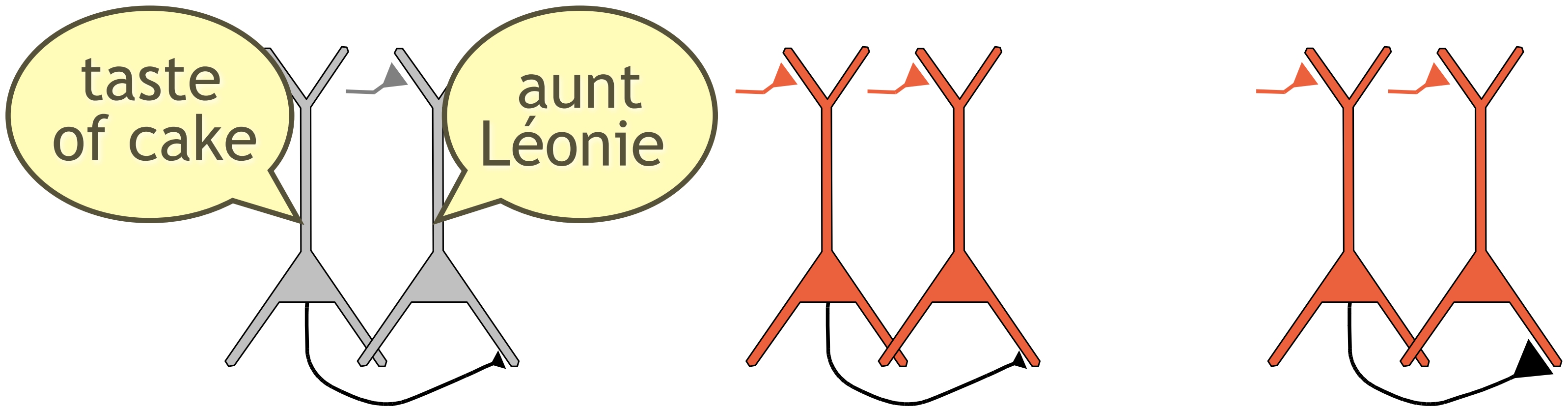
storage



MEMORY PROCESSING IN NEURAL NETWORKS

the Hebbian paradigm

storage

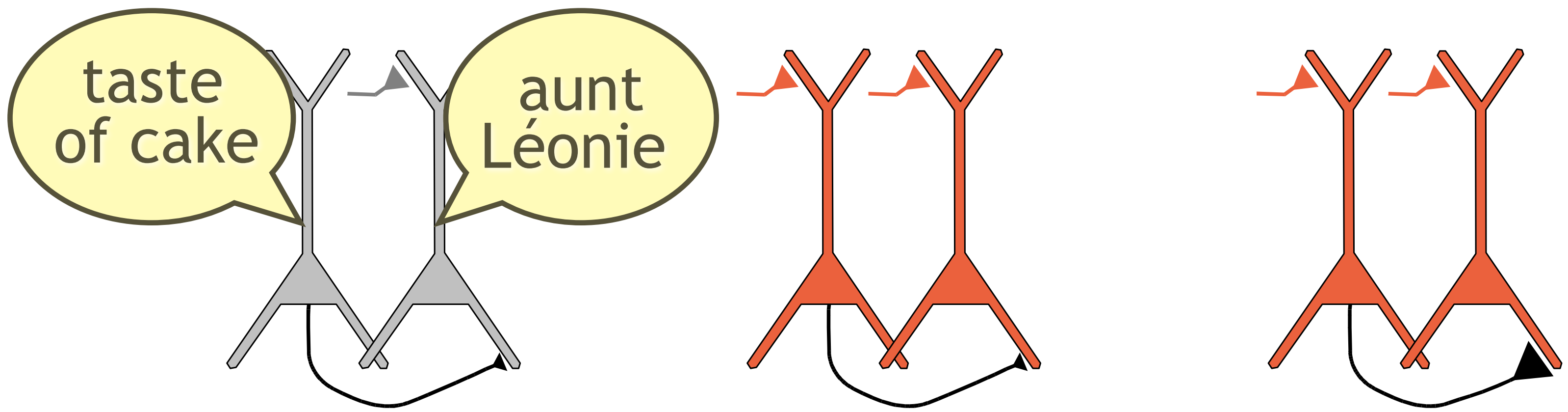


recall

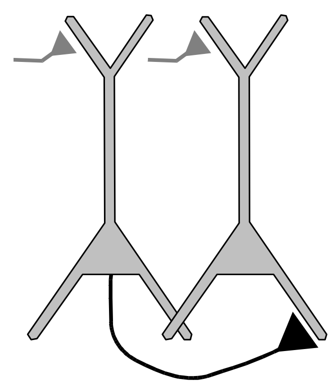
MEMORY PROCESSING IN NEURAL NETWORKS

the Hebbian paradigm

storage



recall

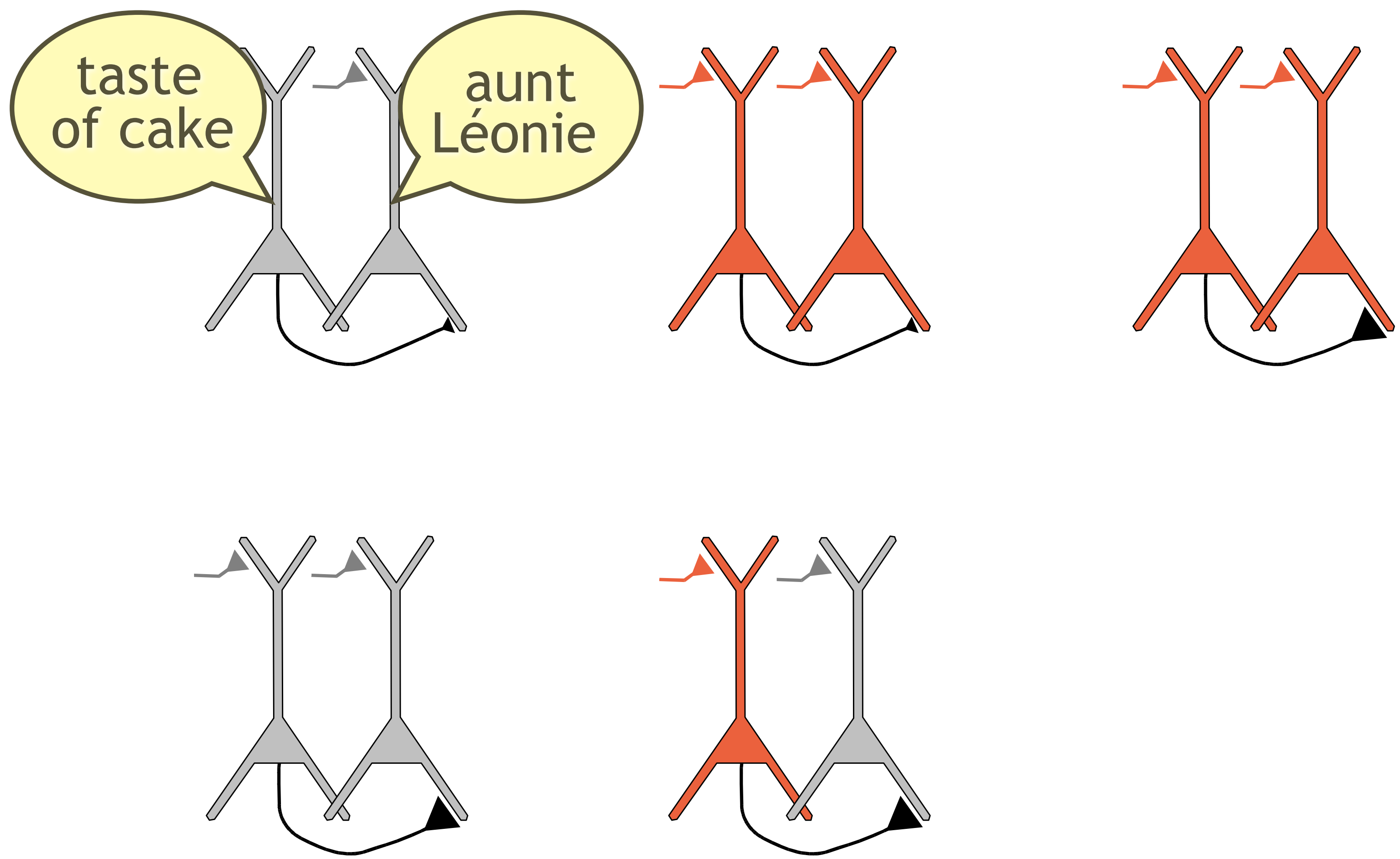


MEMORY PROCESSING IN NEURAL NETWORKS

the Hebbian paradigm

storage

recall

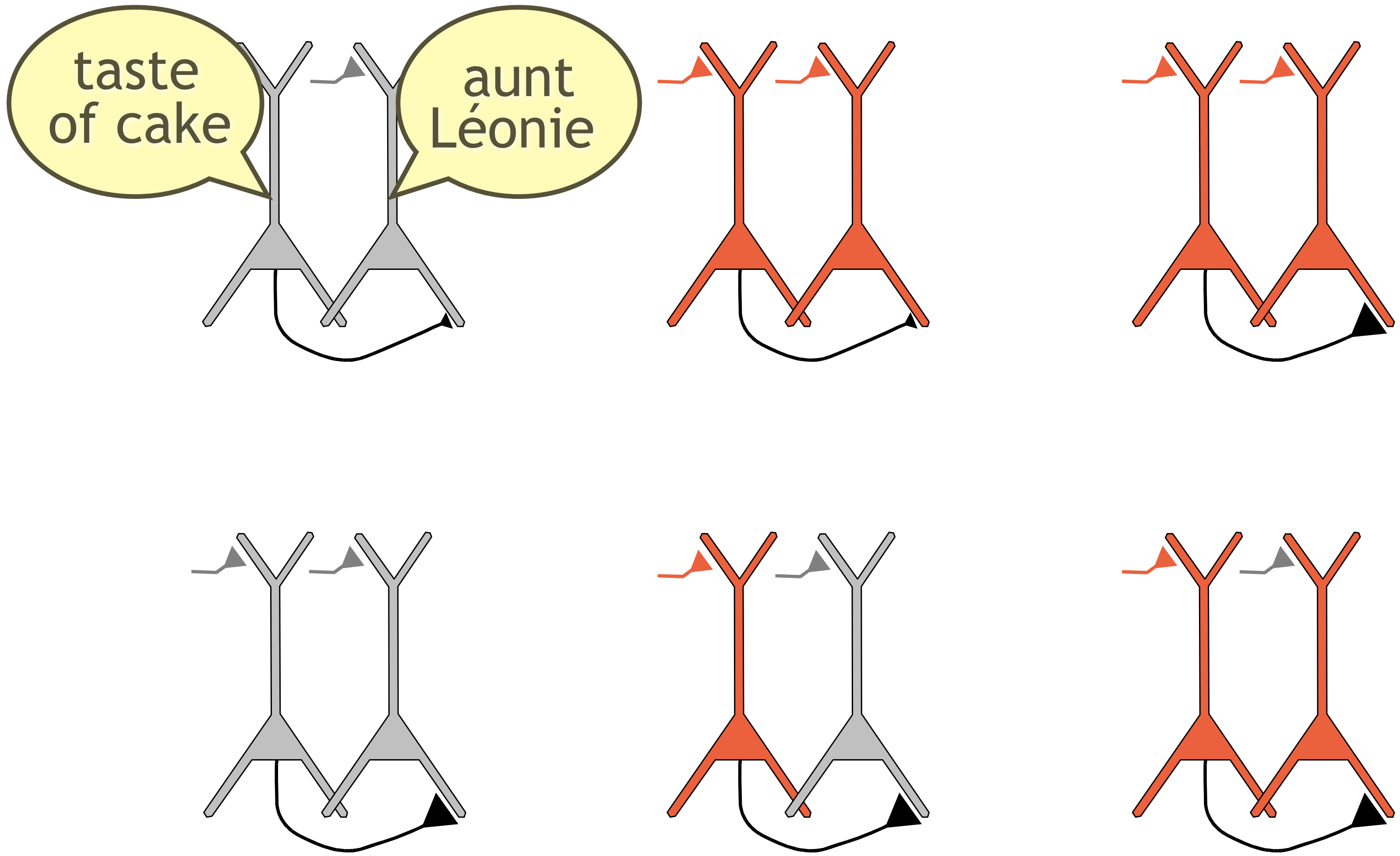


MEMORY PROCESSING IN NEURAL NETWORKS

the Hebbian paradigm

storage

recall

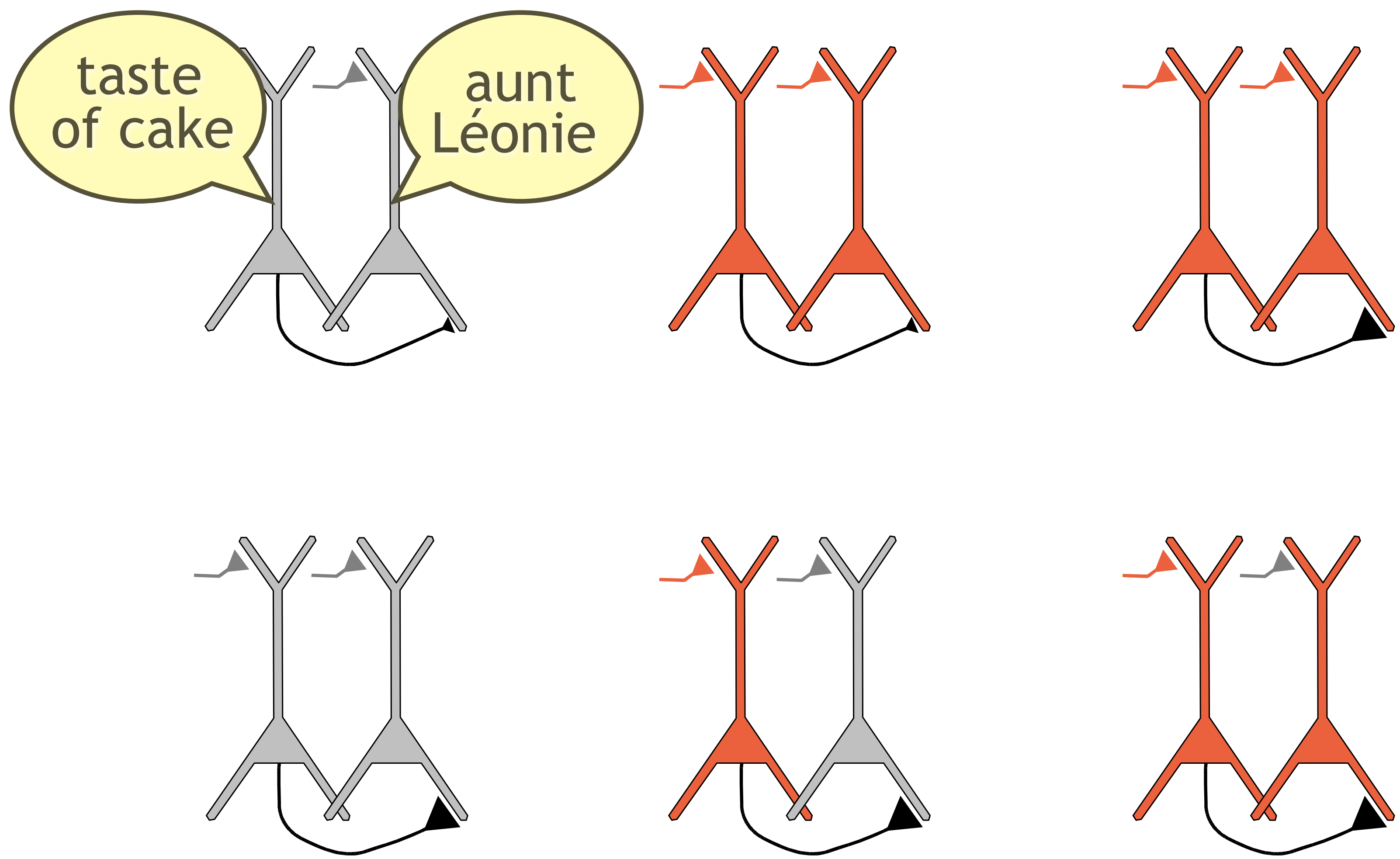


MEMORY PROCESSING IN NEURAL NETWORKS

the Hebbian paradigm

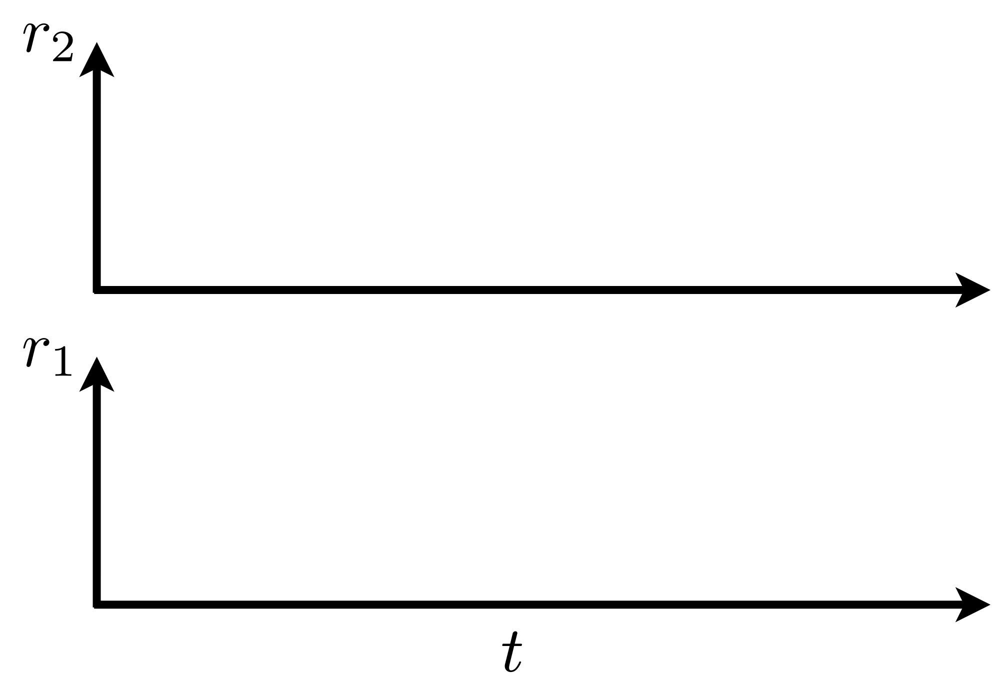
storage

recall

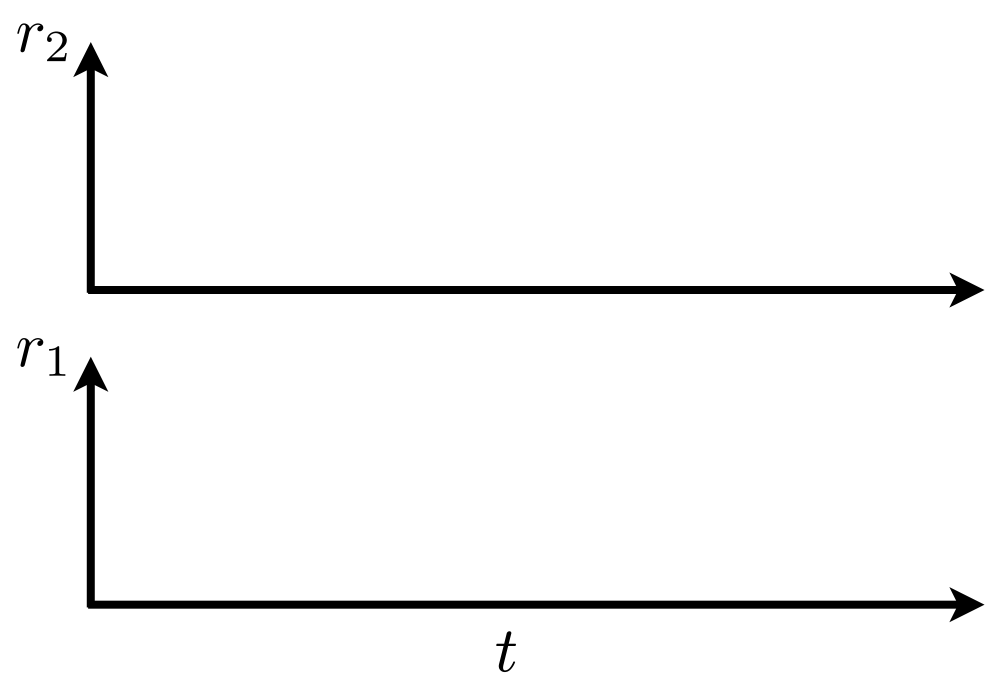


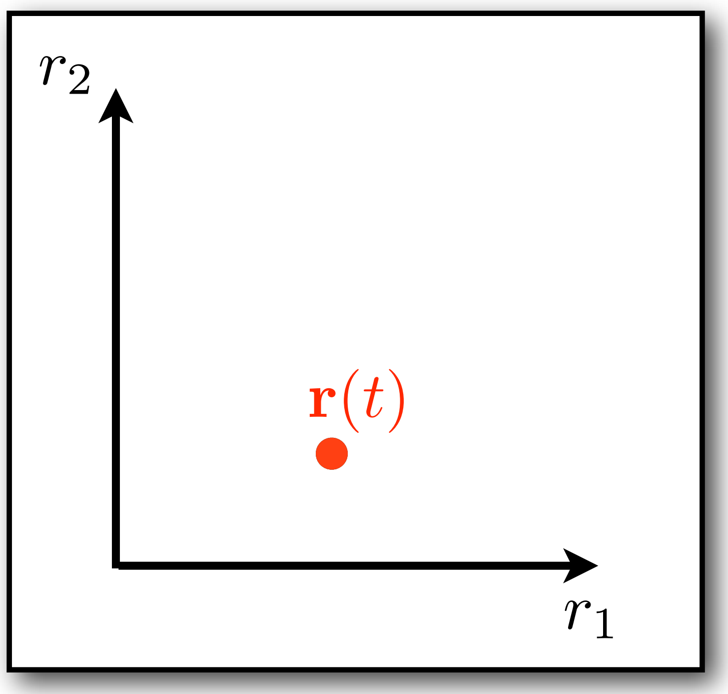
how does this work for **distributed representations**,
without assuming ~~aunt Léonie~~ grandmother neurons ?

THE PHASE PLANE

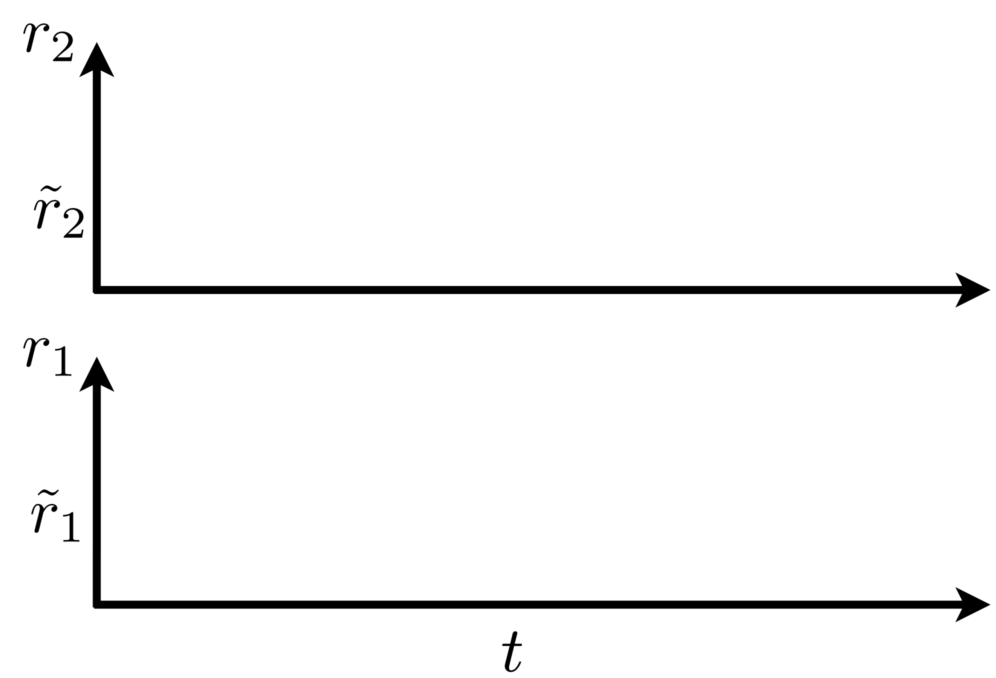


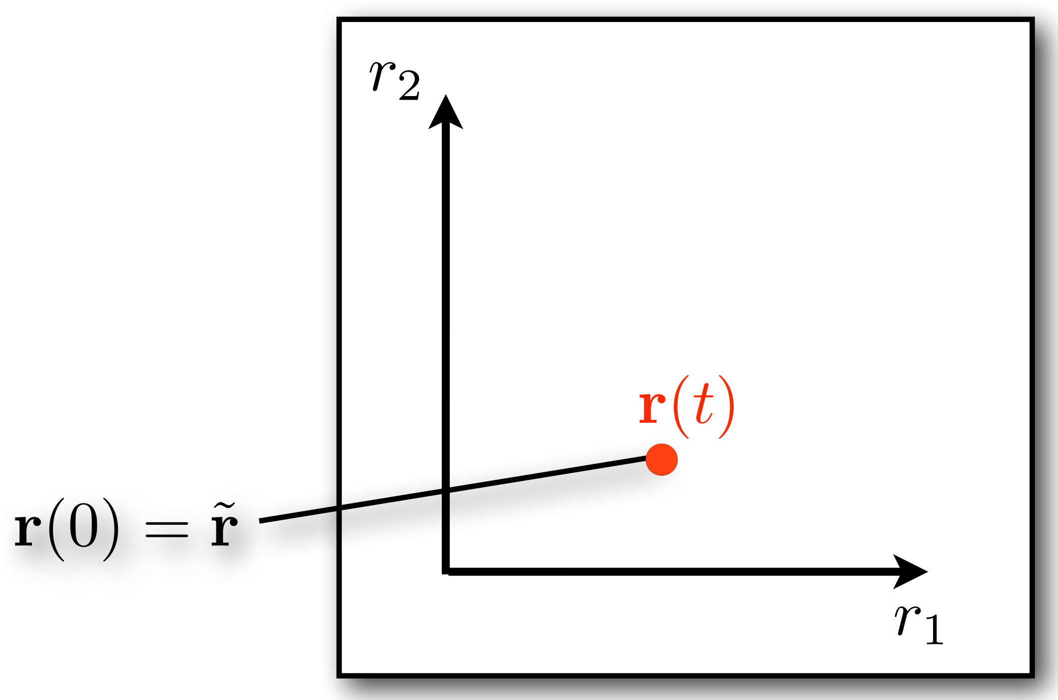
THE PHASE PLANE



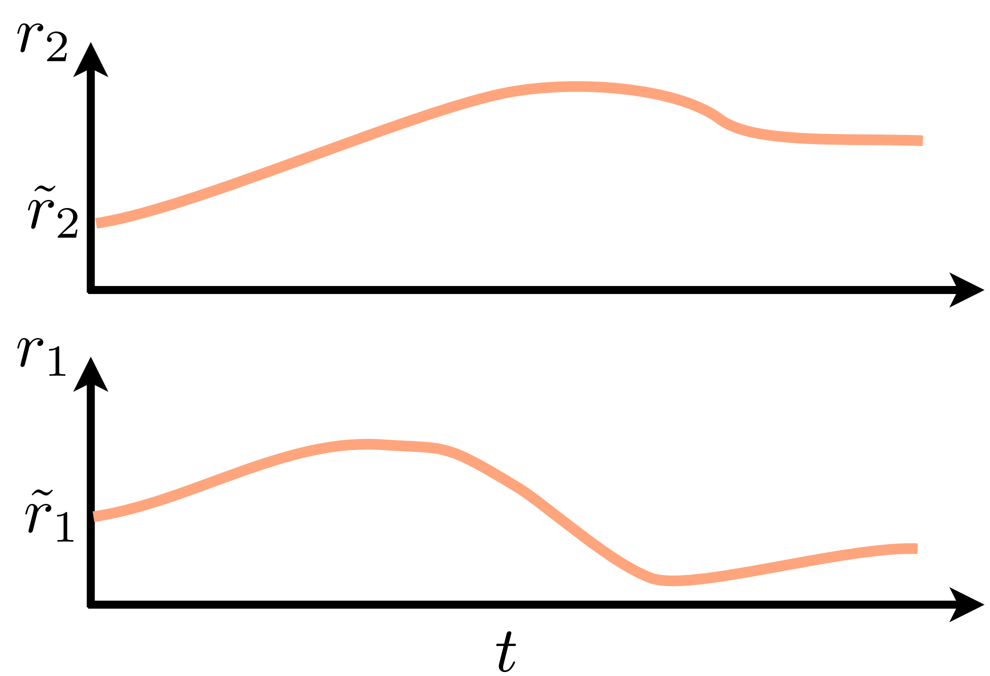


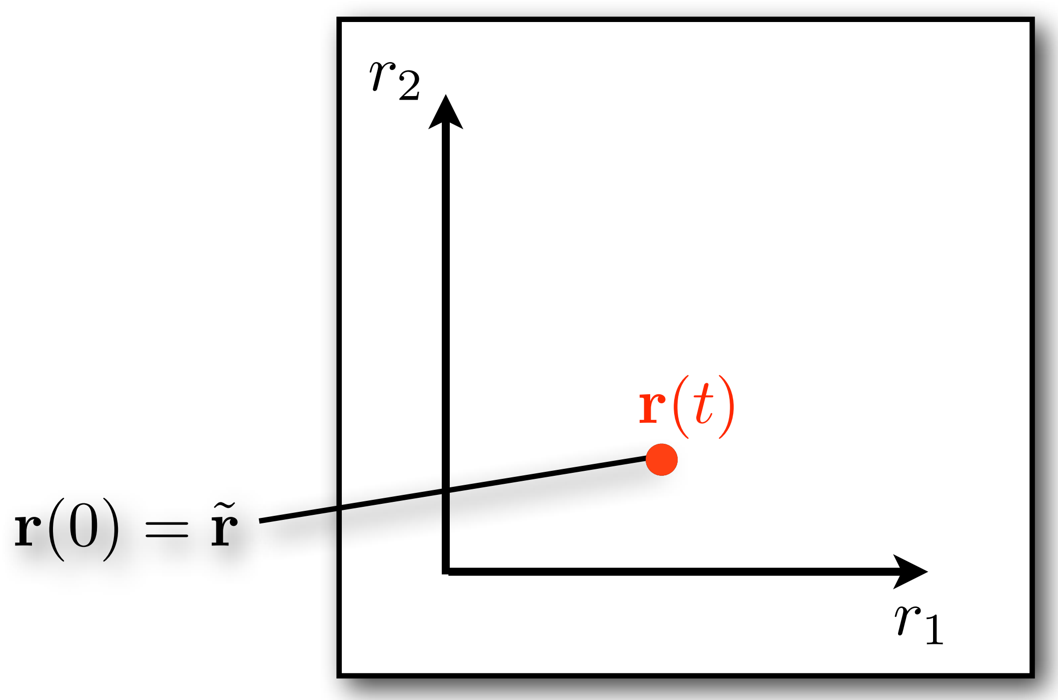
THE PHASE PLANE



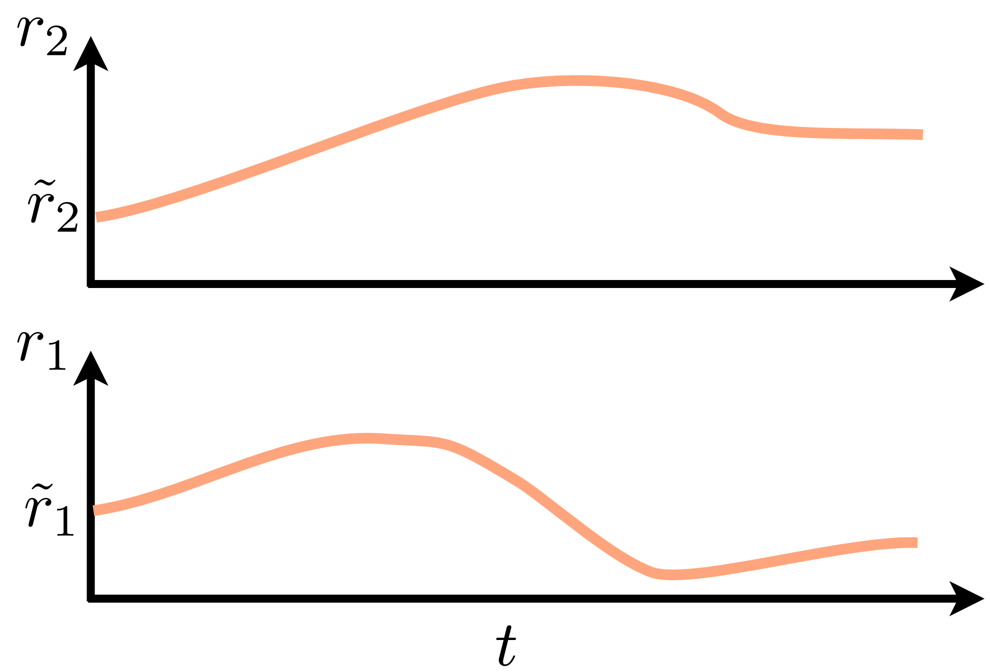


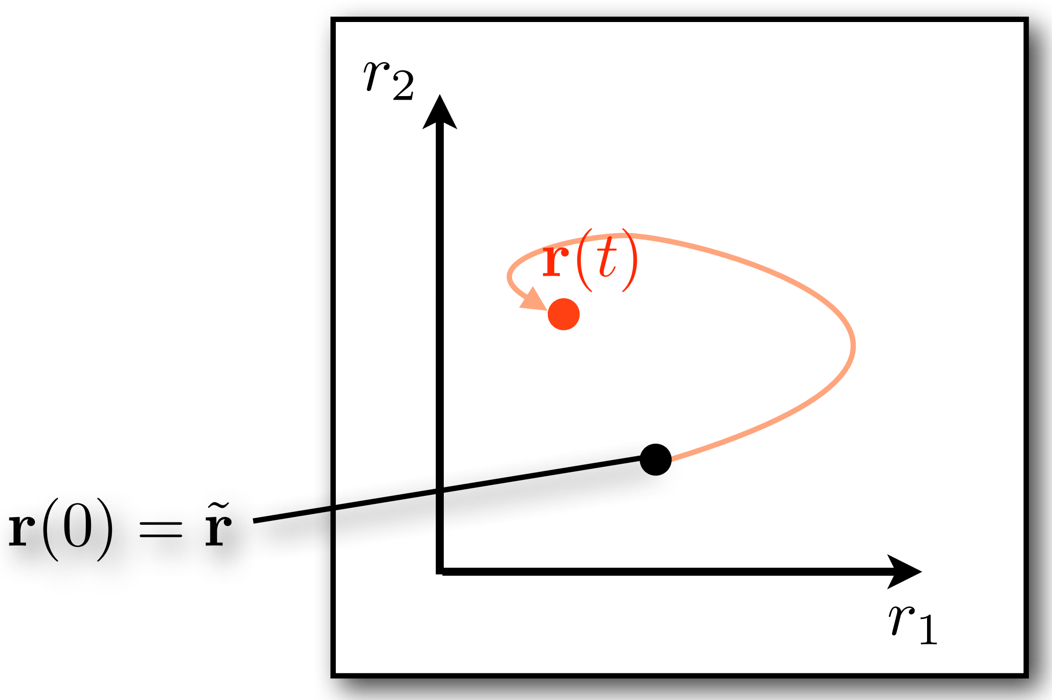
THE PHASE PLANE



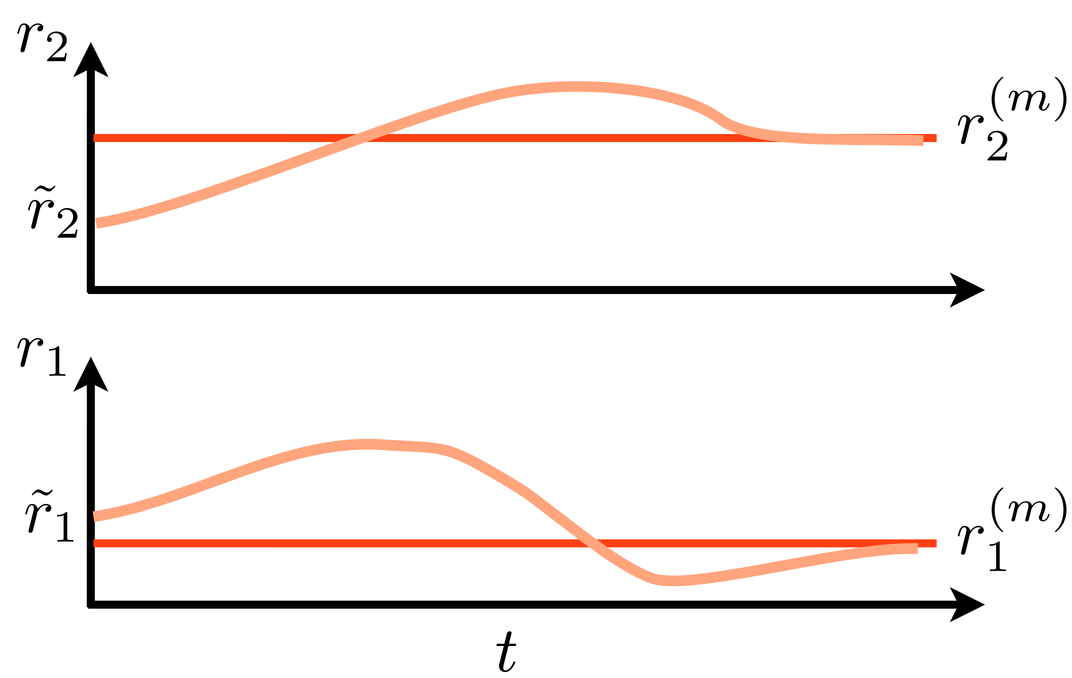


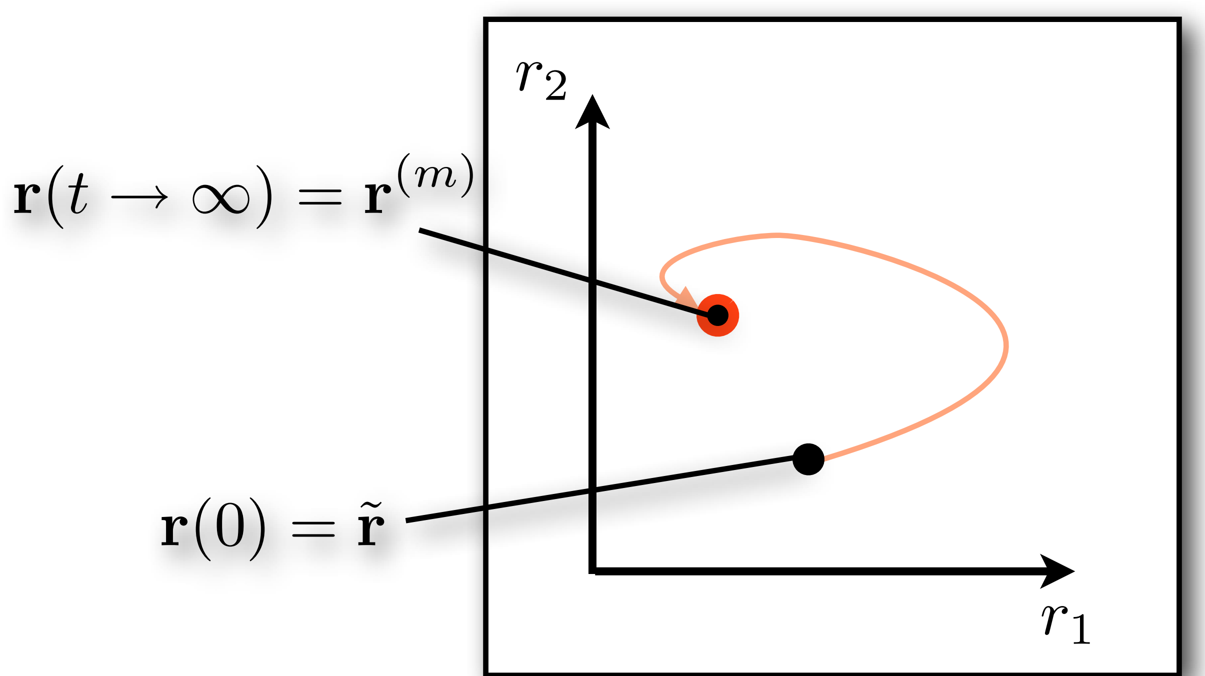
THE PHASE PLANE





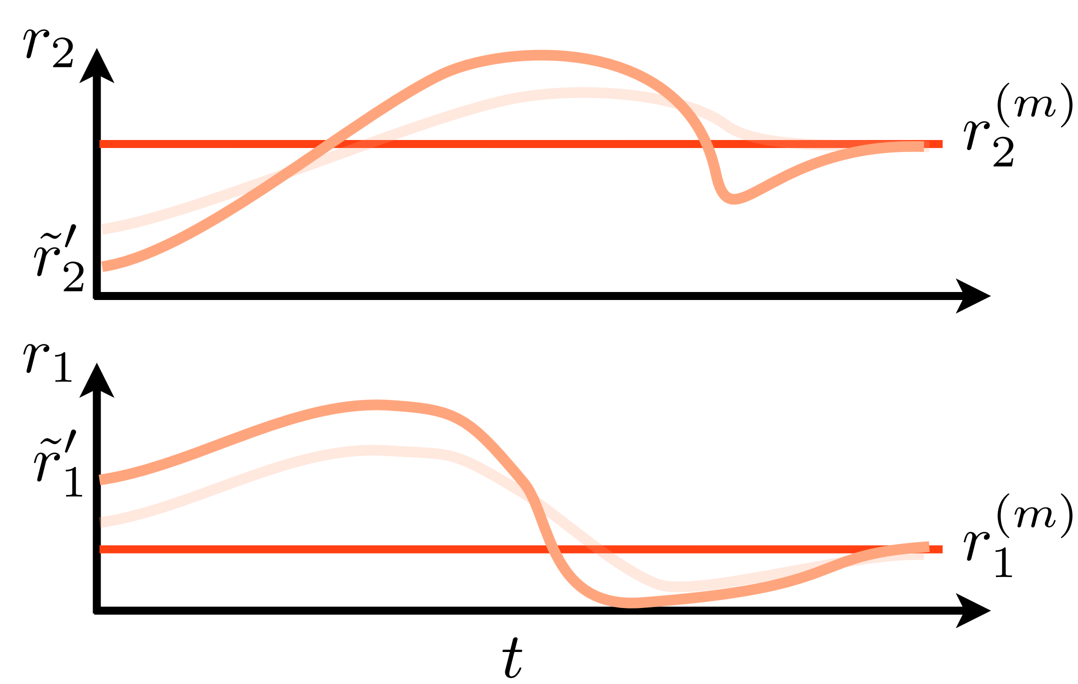
THE PHASE PLANE

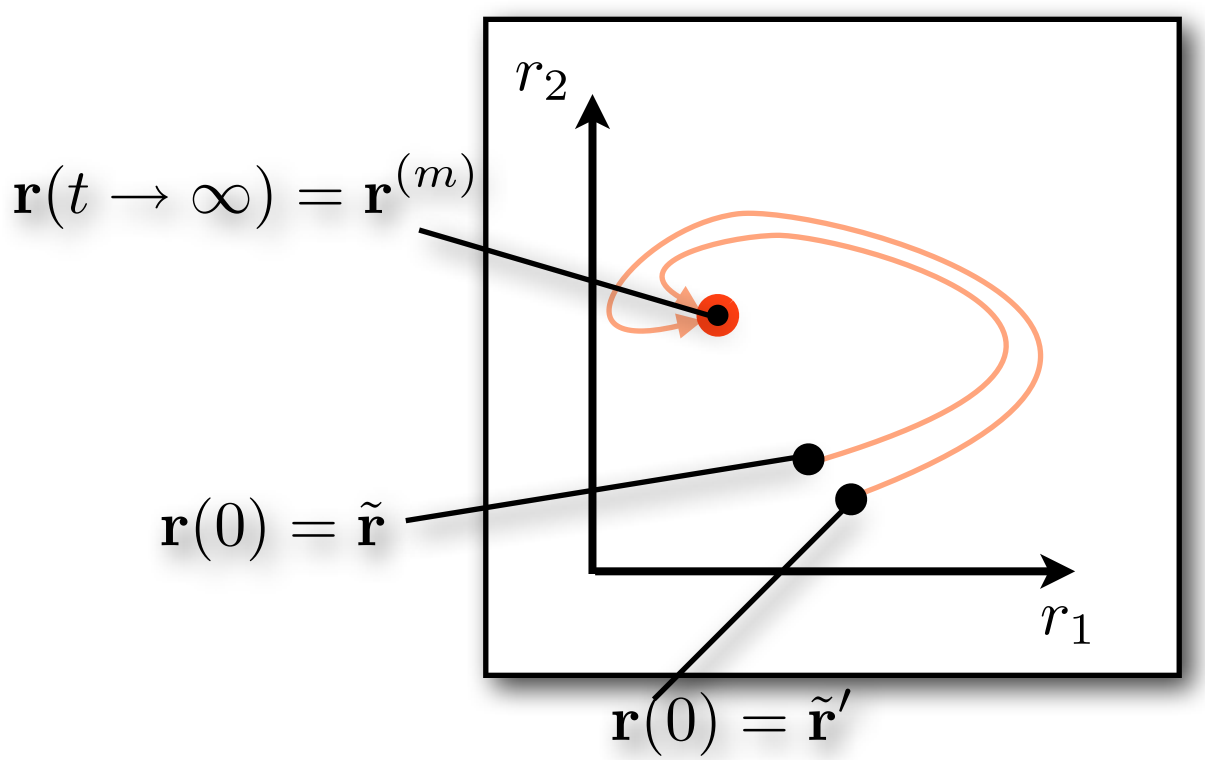




stored patterns should be (point) attractors

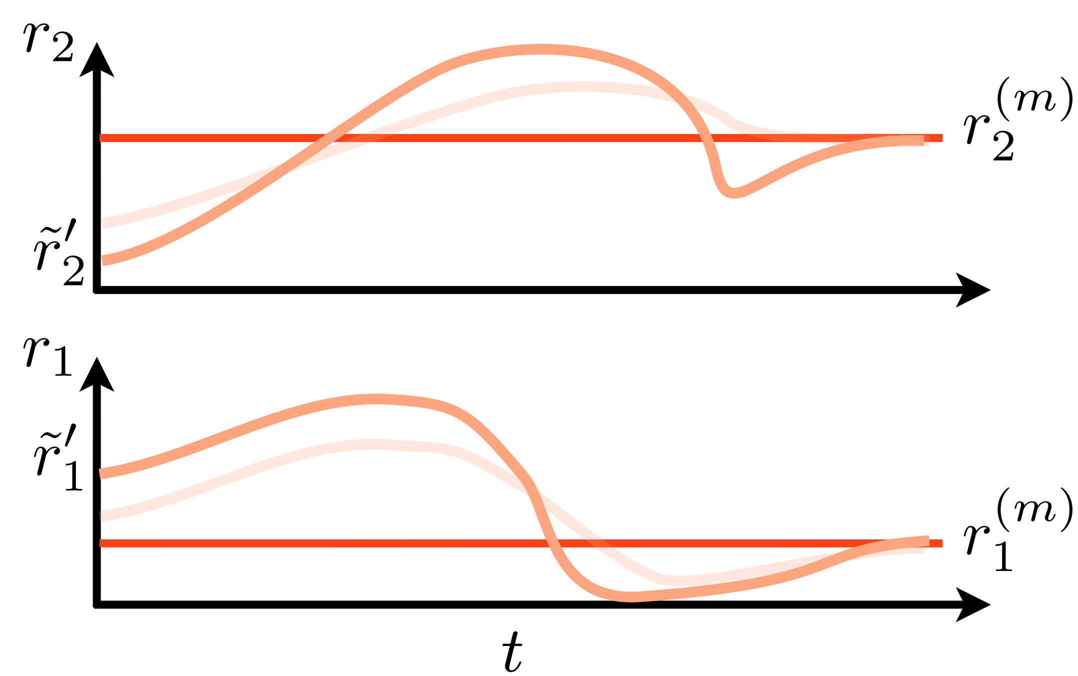
THE PHASE PLANE

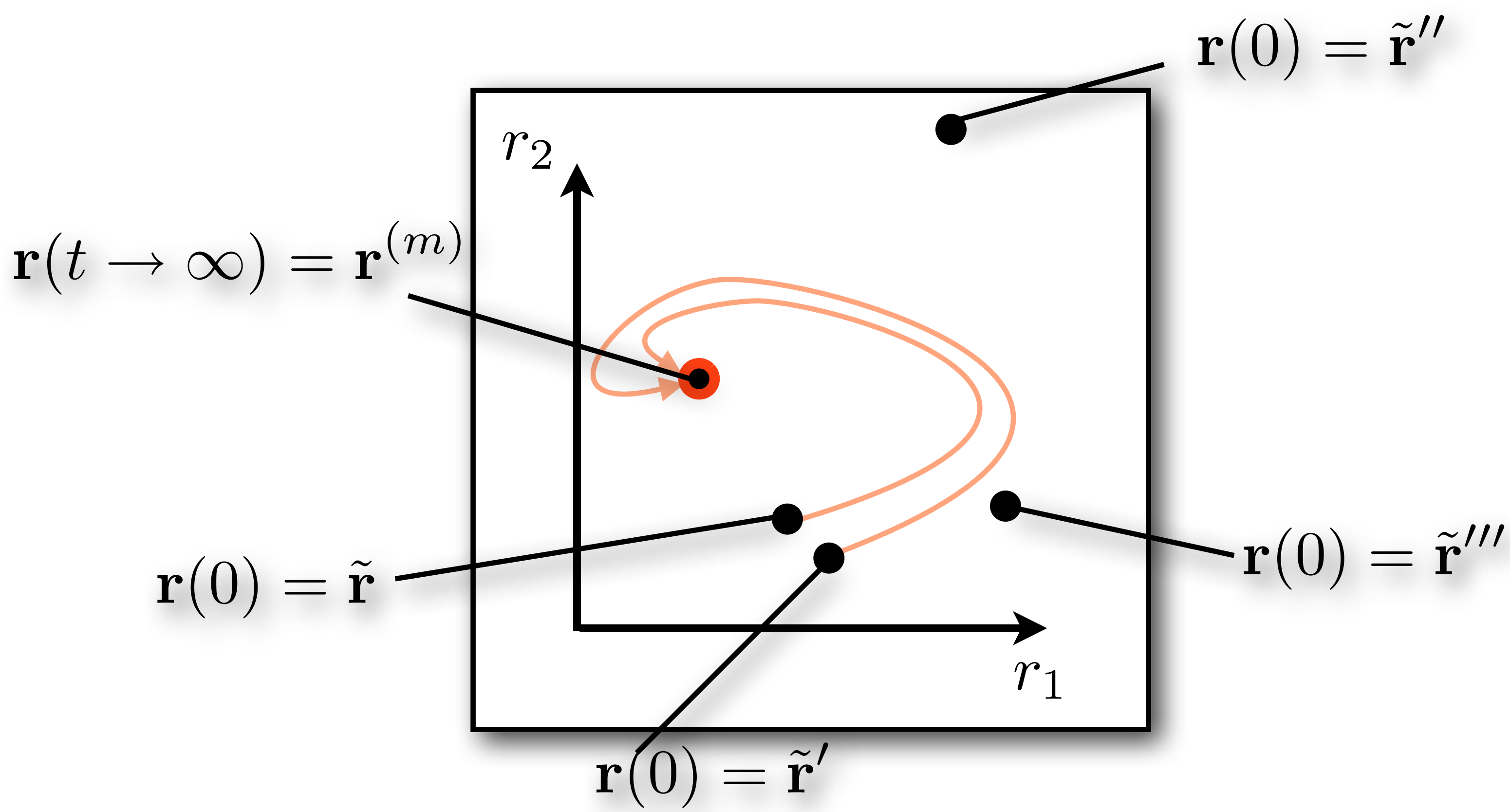




stored patterns should be (point) attractors

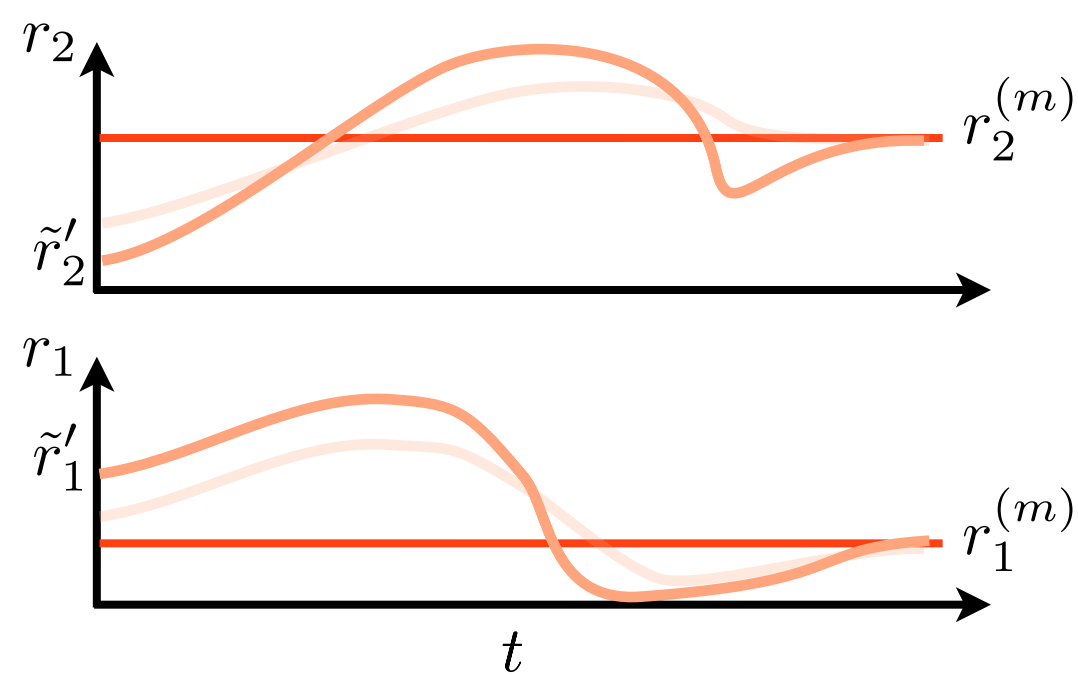
THE PHASE PLANE

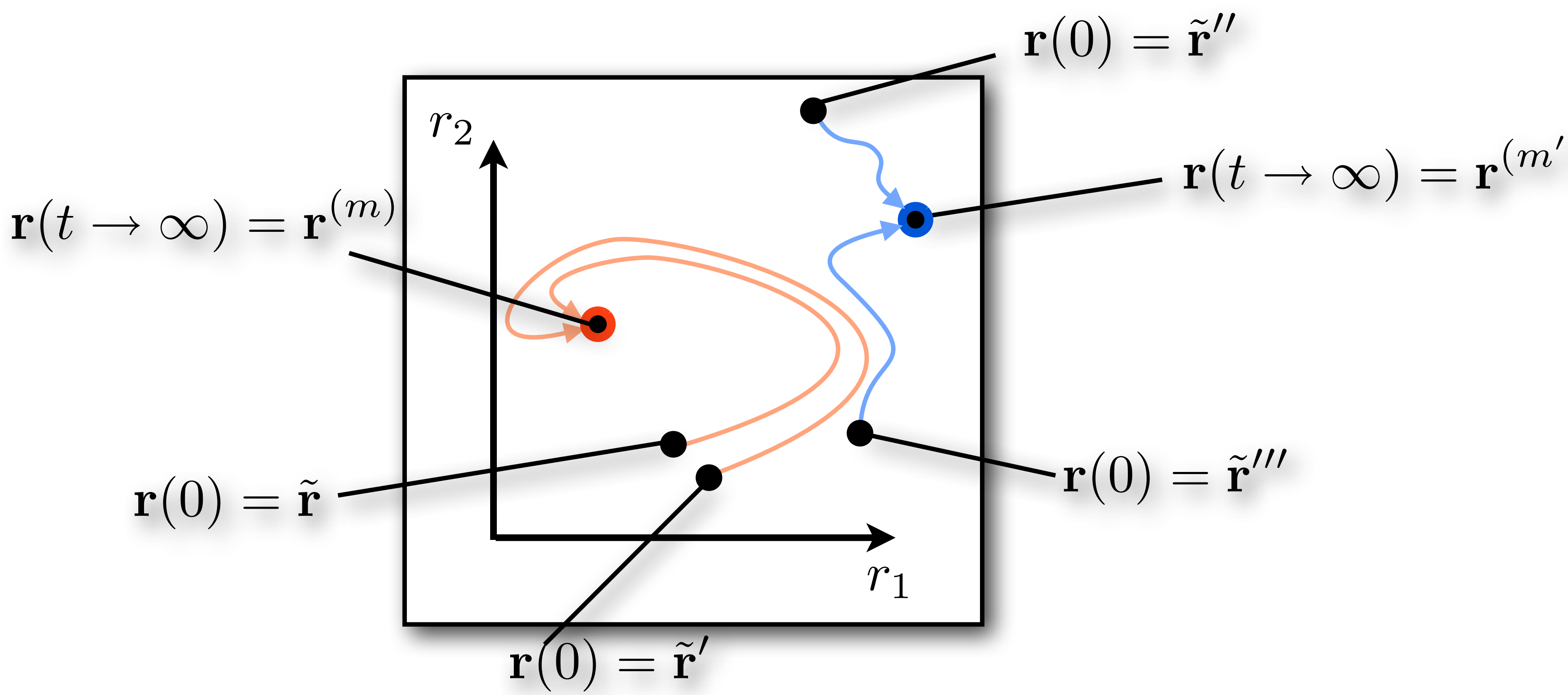




stored patterns should be (point) attractors

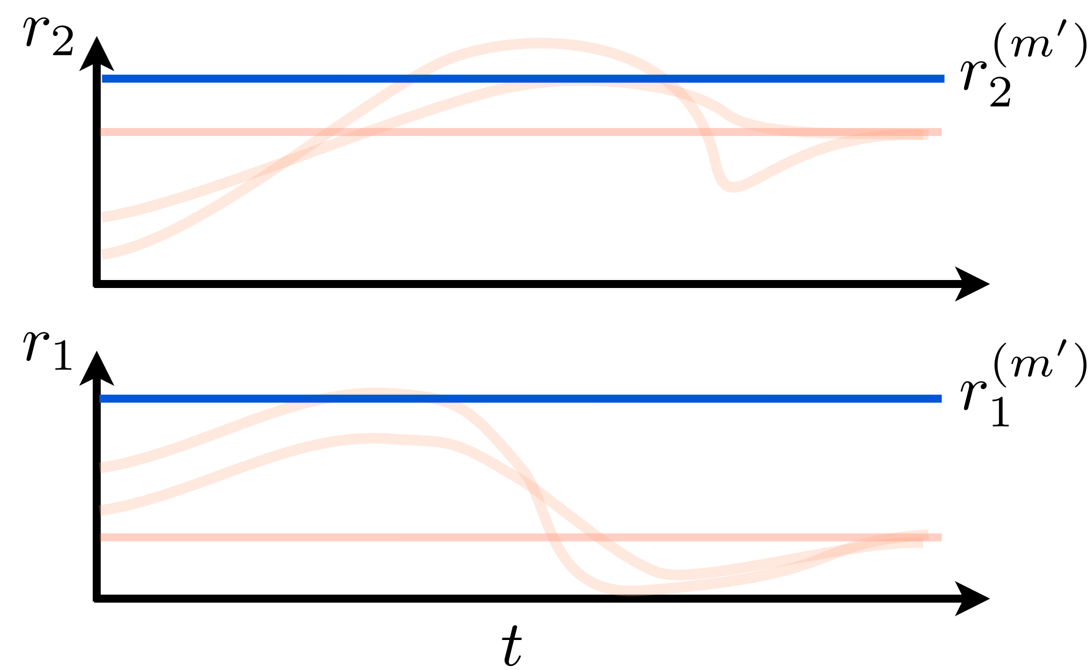
THE PHASE PLANE

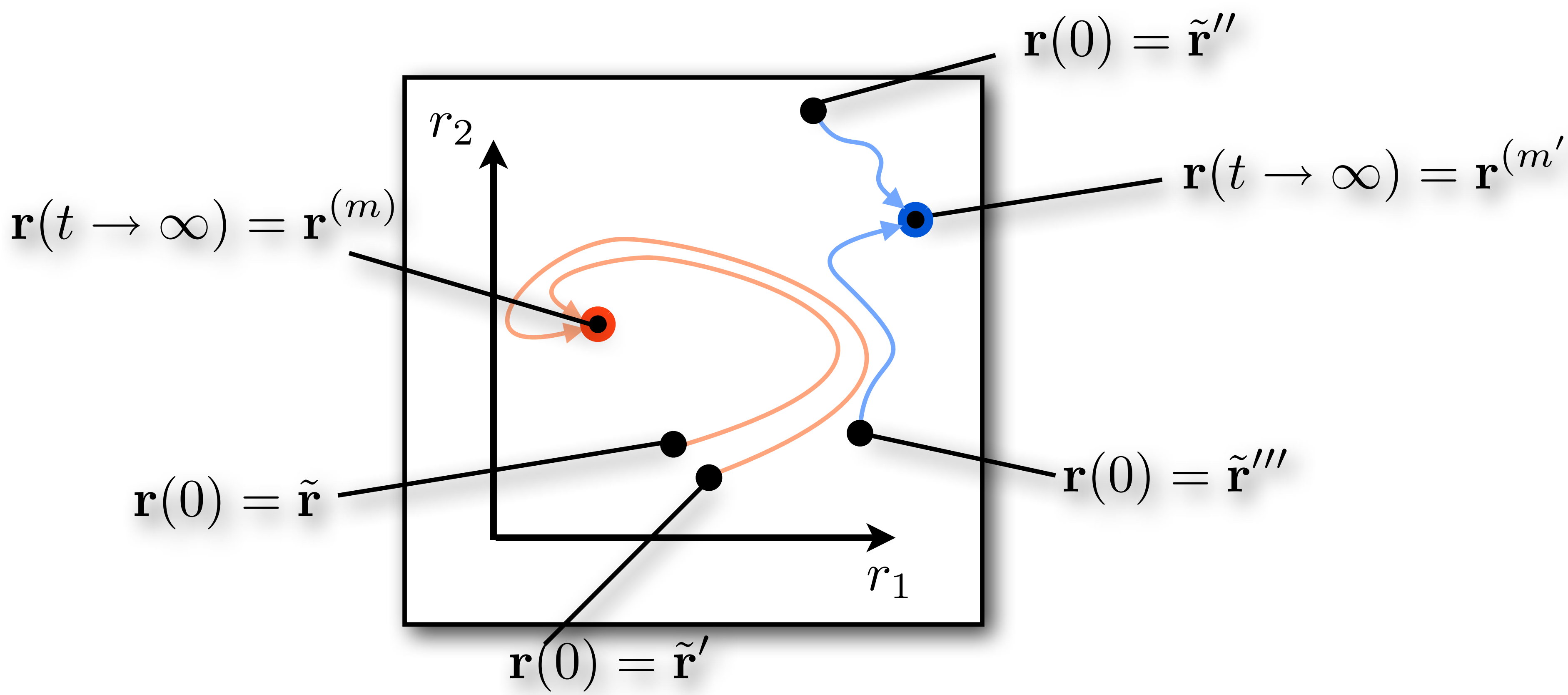




stored patterns should be (point) attractors

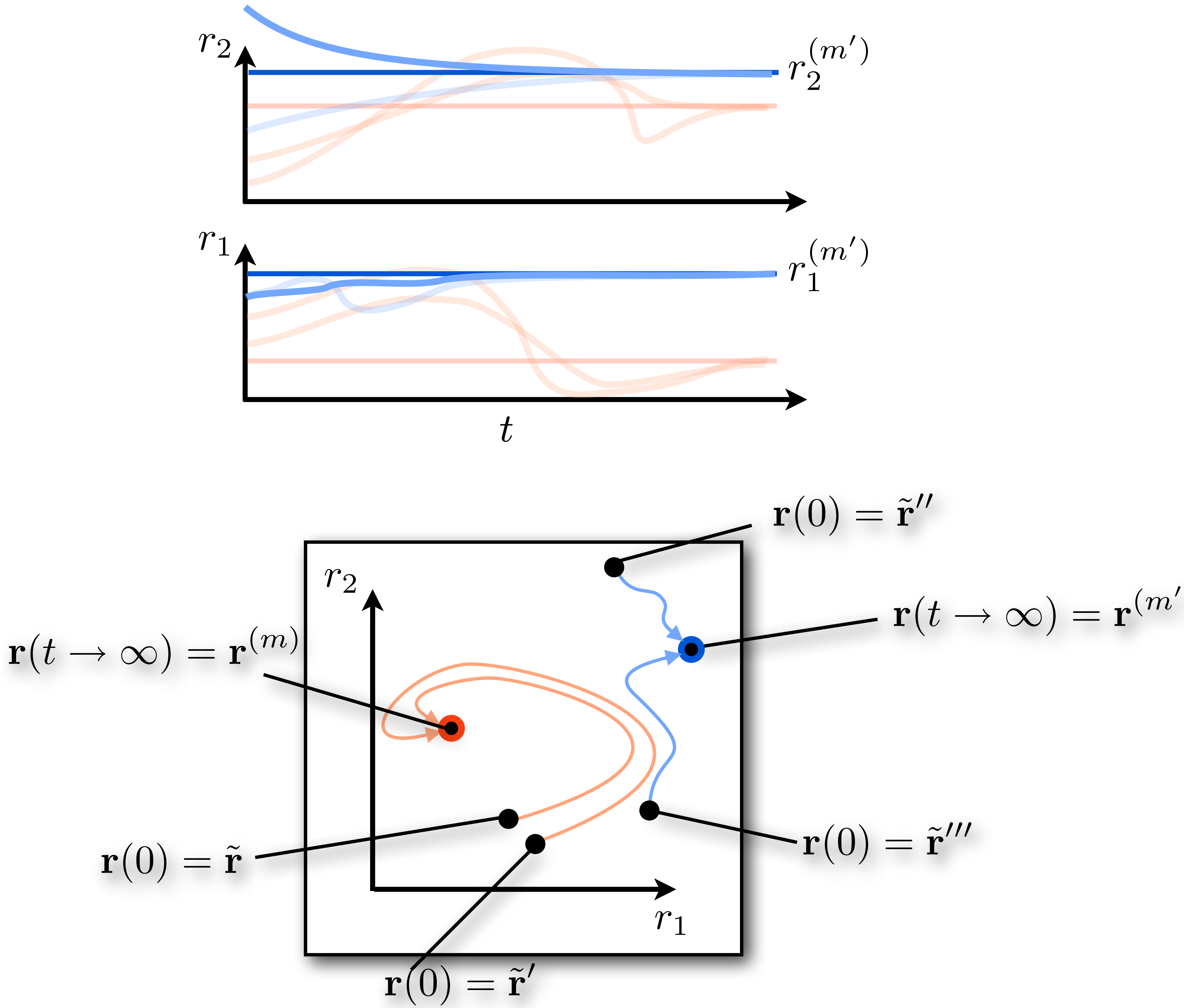
THE PHASE PLANE





stored patterns should be (point) attractors

THE PHASE PLANE



stored patterns should be (point) attractors

HOPFIELD NETWORK

Hopfield, 1982, 1984

HOPFIELD NETWORK

Hopfield, 1982, 1984

binary neurons

McCulloch & Pitts, 1943. A logical calculus of the ideas immanent in nervous activity.

HOPFIELD NETWORK

Hopfield, 1982, 1984

recall: analogue neurons with a firing rate-based description

HOPFIELD NETWORK

Hopfield, 1982, 1984

recall: analogue neurons with a firing rate-based description

$$\frac{dI_i}{dt} = -\frac{1}{\tau} I_i(t) + \sum_{j \neq i} W_{ij}\, r_j(t)$$

HOPFIELD NETWORK

Hopfield, 1982, 1984

recall: analogue neurons with a firing rate-based description

postsynaptic current in cell i

$$\frac{dI_i}{dt} = -\frac{1}{\tau} I_i(t) + \sum_{j \neq i} W_{ij}\, r_j(t)$$

HOPFIELD NETWORK

Hopfield, 1982, 1984

recall: analogue neurons with a firing rate-based description

postsynaptic current in cell i

$$\frac{dI_i}{dt} = -\frac{1}{\tau} I_i(t) + \sum_{j \neq i} W_{ij}\, r_j(t)$$

synaptic time constant

HOPFIELD NETWORK

Hopfield, 1982, 1984

recall: analogue neurons with a firing rate-based description

postsynaptic current in cell i

$$\frac{dI_i}{dt} = -\frac{1}{\tau} I_i(t) + \sum_{j \neq i} W_{ij}\, r_j(t)$$

synaptic time constant

synaptic efficacy between presynaptic cell j and postsynaptic cell i

HOPFIELD NETWORK

Hopfield, 1982, 1984

recall: analogue neurons with a firing rate-based description

postsynaptic current in cell i

firing rate of presynaptic cell j

$$\frac{dI_i}{dt} = -\frac{1}{\tau} I_i(t) + \sum_{j \neq i} W_{ij}\, r_j(t)$$

synaptic time constant

synaptic efficacy between presynaptic cell j and postsynaptic cell i

HOPFIELD NETWORK

Hopfield, 1982, 1984

recall: analogue neurons with a firing rate-based description

postsynaptic current in cell i

firing rate of presynaptic cell j

$$\frac{dI_i}{dt} = -\frac{1}{\tau} I_i(t) + \sum_{j \neq i} W_{ij}\, r_j(t)$$

synaptic time constant

synaptic efficacy between presynaptic cell j and postsynaptic cell i

$$r_i(t) = F(I_i(t))$$

HOPFIELD NETWORK

Hopfield, 1982, 1984

recall: analogue neurons with a firing rate-based description

postsynaptic current in cell i

firing rate of presynaptic cell j

$$\frac{dI_i}{dt} = -\frac{1}{\tau} I_i(t) + \sum_{j \neq i} W_{ij}\, r_j(t)$$

synaptic time constant

synaptic efficacy between presynaptic cell j and postsynaptic cell i

firing rate of cell i

$$r_i(t) = F(I_i(t))$$

HOPFIELD NETWORK

Hopfield, 1982, 1984

recall: analogue neurons with a firing rate-based description

postsynaptic current in cell i

firing rate of presynaptic cell j

$$\frac{dI_i}{dt} = -\frac{1}{\tau} I_i(t) + \sum_{j \neq i} W_{ij}\, r_j(t)$$

synaptic time constant

synaptic efficacy between presynaptic cell j and postsynaptic cell i

firing rate of cell i

$$r_i(t) = F(I_i(t))$$

f-I characteristic of cells

HOPFIELD NETWORK

Hopfield, 1982, 1984

recall: analogue neurons with a firing rate-based description

postsynaptic current in cell i

firing rate of presynaptic cell j

$$\frac{dI_i}{dt} = -\frac{1}{\tau} I_i(t) + \sum_{j \neq i} W_{ij}\, r_j(t)$$

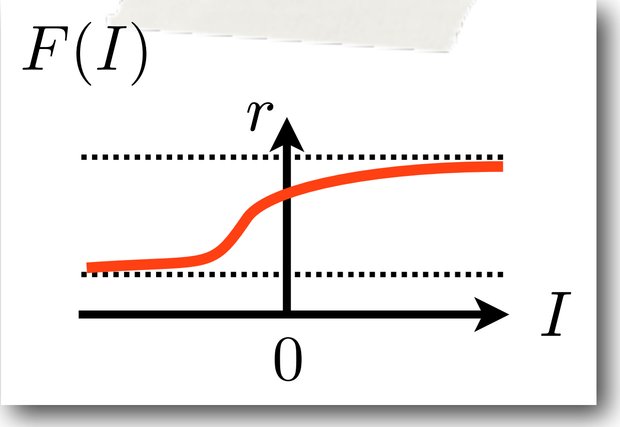
synaptic time constant

synaptic efficacy between presynaptic cell j and postsynaptic cell i

firing rate of cell i

$$r_i(t) = F(I_i(t))$$

f-I characteristic of cells



HOPFIELD NETWORK

Hopfield, 1982, 1984

recall: analogue neurons with a firing rate-based description

postsynaptic current in cell i

firing rate of presynaptic cell j

$$\frac{dI_i}{dt} = -\frac{1}{\tau} I_i(t) + \sum_{j \neq i} W_{ij}\, r_j(t)$$

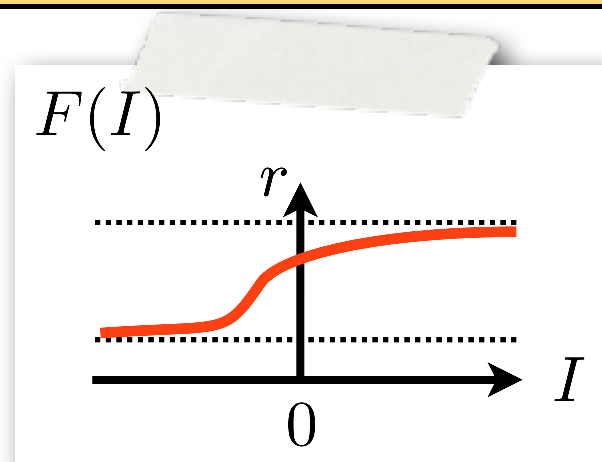
synaptic time constant

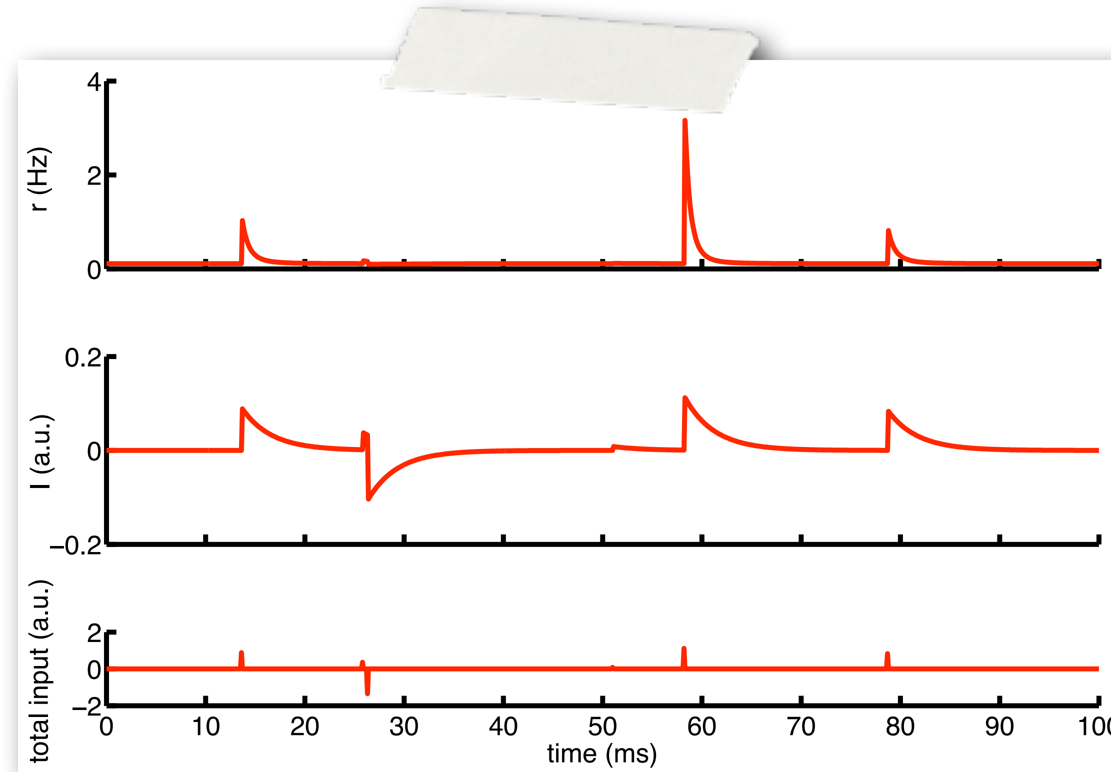
synaptic efficacy between presynaptic cell j and postsynaptic cell i

firing rate of cell i

$$r_i(t) = F(I_i(t))$$

f-I characteristic of cells





HOPFIELD NETWORK

Hopfield, 1982, 1984

recall: analogue neurons with a firing rate-based description

postsynaptic current in cell i

firing rate of presynaptic cell j

$$\frac{dI_i}{dt} = -\frac{1}{\tau} I_i(t) + \sum_{j \neq i} W_{ij}\, r_j(t)$$

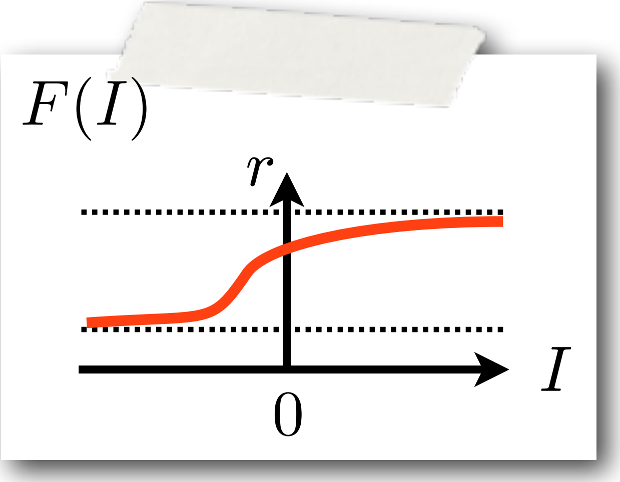
synaptic time constant

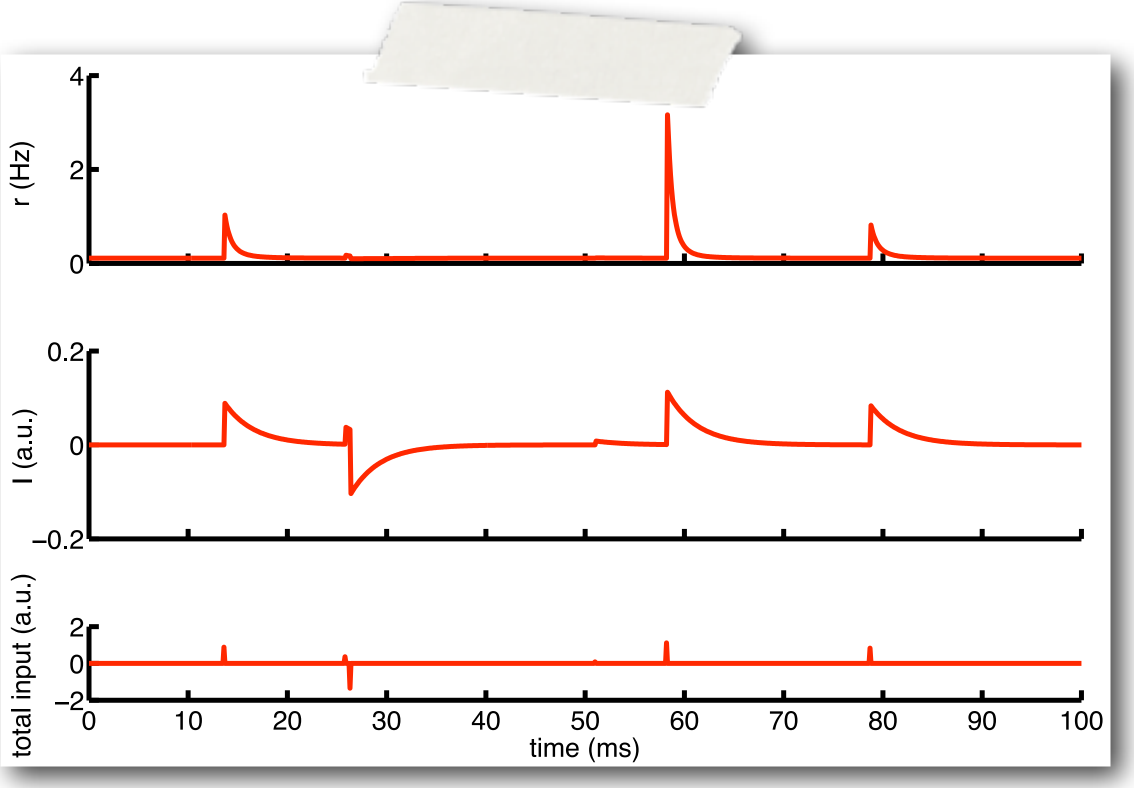
synaptic efficacy between presynaptic cell j and postsynaptic cell i

firing rate of cell i

$$r_i(t) = F(I_i(t))$$

f-I characteristic of cells





storage: "Hebbian" synaptic plasticity

HOPFIELD NETWORK

Hopfield, 1982, 1984

recall: analogue neurons with a firing rate-based description

postsynaptic current in cell i

firing rate of presynaptic cell j

$$\frac{dI_i}{dt} = -\frac{1}{\tau} I_i(t) + \sum_{j \neq i} W_{ij}\, r_j(t)$$

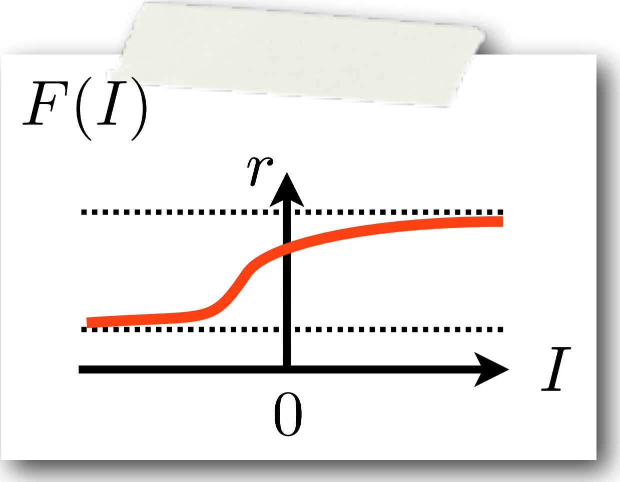
synaptic time constant

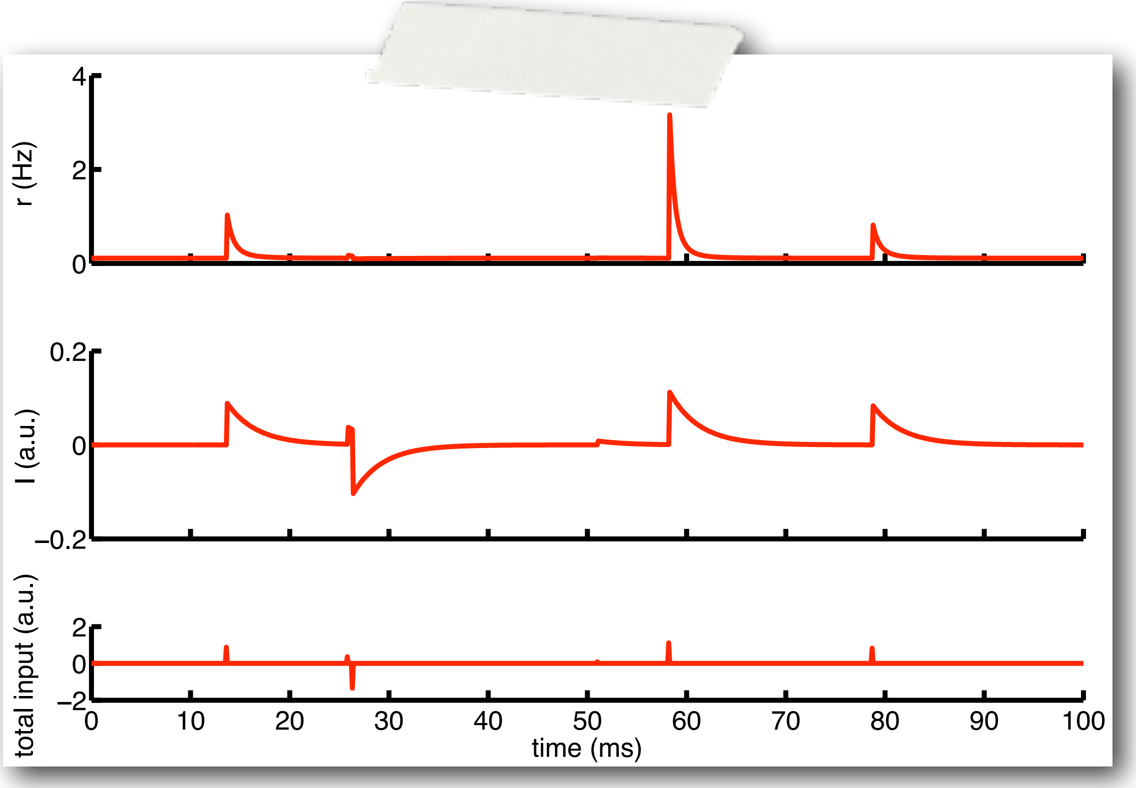
synaptic efficacy between presynaptic cell j and postsynaptic cell i

firing rate of cell i

$$r_i(t) = F(I_i(t))$$

f-I characteristic of cells





storage: "Hebbian" synaptic plasticity

binary, uncorrelated, balanced patterns

HOPFIELD NETWORK

Hopfield, 1982, 1984

recall: analogue neurons with a firing rate-based description

postsynaptic current in cell i

firing rate of presynaptic cell j

$$\frac{dI_i}{dt} = -\frac{1}{\tau} I_i(t) + \sum_{j \neq i} W_{ij}\, r_j(t)$$

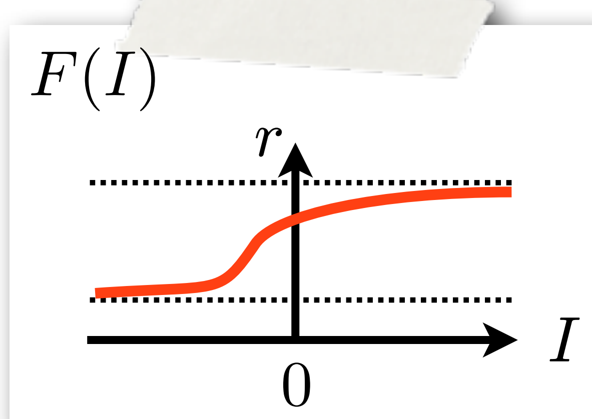
synaptic time constant

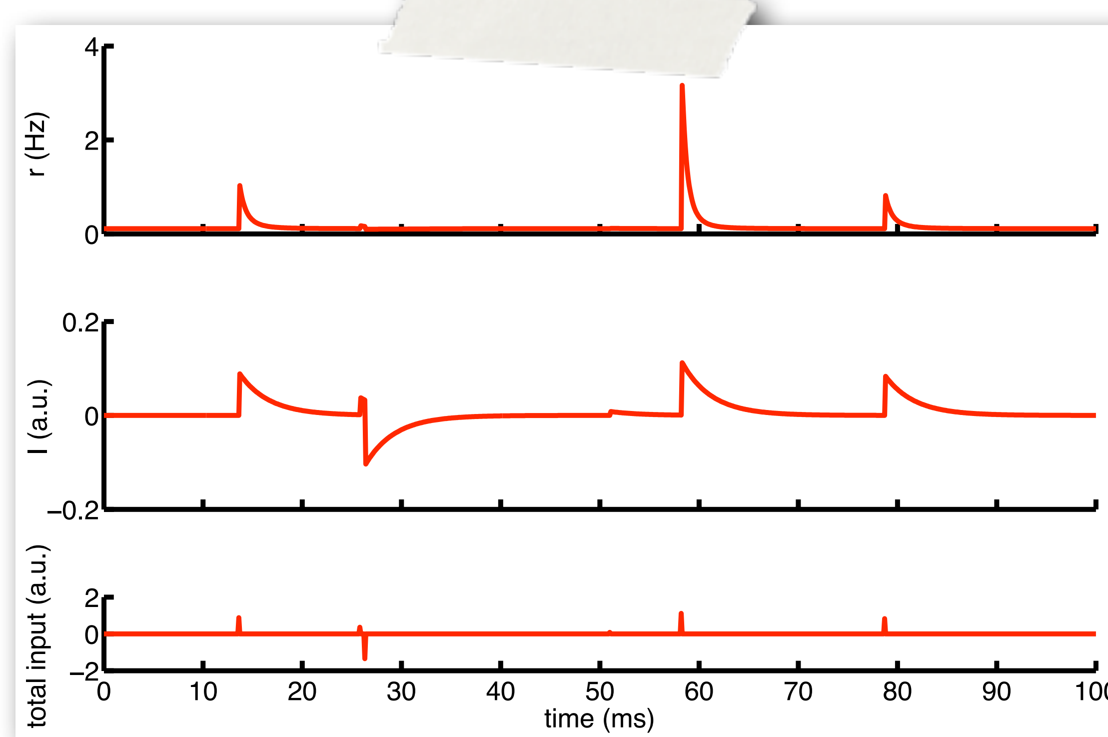
synaptic efficacy between presynaptic cell j and postsynaptic cell i

firing rate of cell i

$$r_i(t) = F(I_i(t))$$

f-I characteristic of cells





storage: "Hebbian" synaptic plasticity

binary, uncorrelated, balanced patterns

covariance rule

$$W_{ij} = \sum_{m=1}^{M} \left(r_i^{(m)} - \frac{1}{2}\right)\left(r_j^{(m)} - \frac{1}{2}\right)$$

$$W_{ii} = 0$$

HOPFIELD NETWORK

Hopfield, 1982, 1984

recall: analogue neurons with a firing rate-based description

postsynaptic current in cell i

firing rate of presynaptic cell j

$$\frac{dI_i}{dt} = -\frac{1}{\tau} I_i(t) + \sum_{j \neq i} W_{ij} r_j(t)$$

synapses are kept fixed: plasticity does not play a role

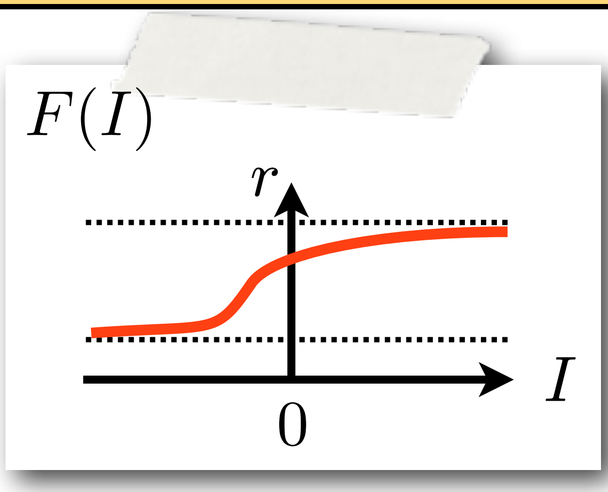
synaptic time constant

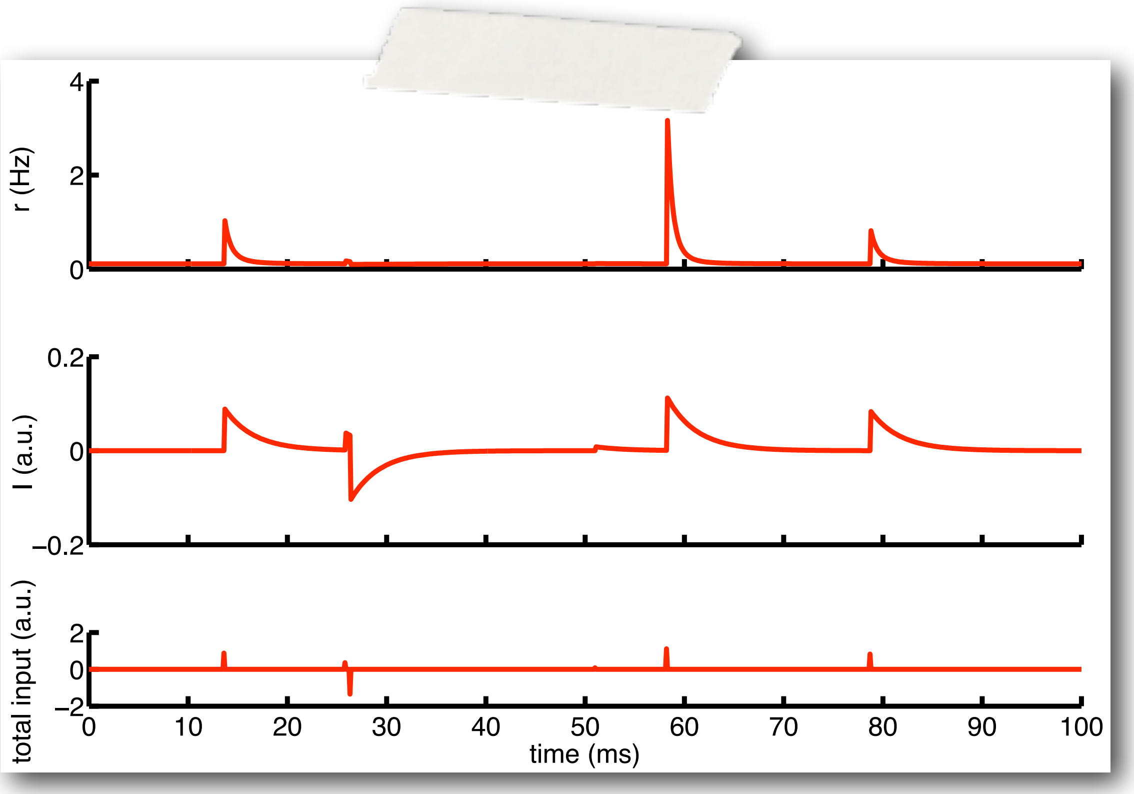
synaptic efficacy between presynaptic cell j and postsynaptic cell i

firing rate of cell i

$$r_i(t) = F(I_i(t))$$

f-I characteristic of cells





storage: "Hebbian" synaptic plasticity

binary, uncorrelated, balanced patterns

covariance rule

$$W_{ij} = \sum_{m=1}^{M} \left(r_i^{(m)} - \frac{1}{2} \right) \left(r_j^{(m)} - \frac{1}{2} \right)$$

$$W_{ii} = 0$$

HOPFIELD NETWORK

Hopfield, 1982, 1984

recall: analogue neurons with a firing rate-based description

postsynaptic current in cell i

firing rate of presynaptic cell j

$$\frac{dI_i}{dt} = -\frac{1}{\tau} I_i(t) + \sum_{j \neq i} W_{ij}\, r_j(t)$$

synapses are kept fixed:
plasticity does not play a role

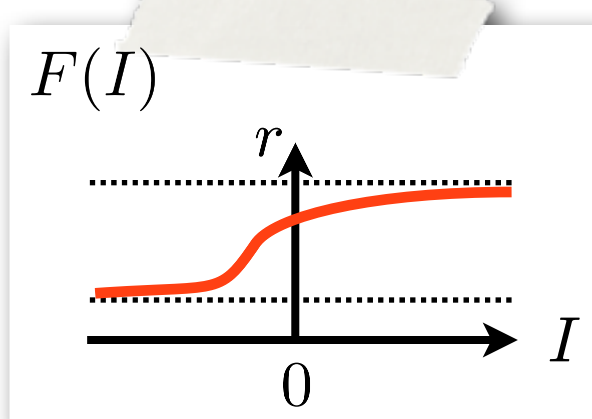
synaptic time constant

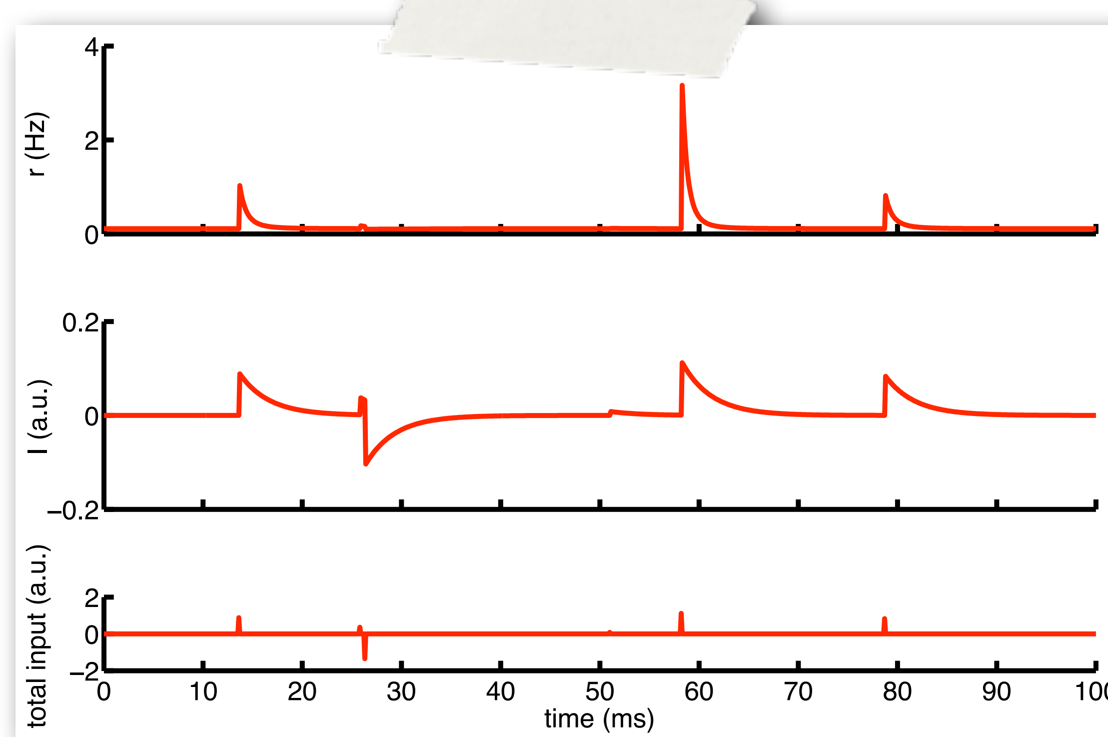
synaptic efficacy between presynaptic cell j and postsynaptic cell i

firing rate of cell i

$$r_i(t) = F(I_i(t))$$

f-I characteristic of cells





storage: "Hebbian" synaptic plasticity

binary, uncorrelated, balanced patterns

covariance rule

$$W_{ij} = \sum_{m=1}^{M} \left(r_i^{(m)} - \frac{1}{2} \right) \left(r_j^{(m)} - \frac{1}{2} \right)$$

$$W_{ii} = 0$$

activities are forced onto the network:
dynamics do not play a role

MATHEMATICAL ANALYSIS

→ equations for Hopfield network are like the equations describing ‘spin glasses’
→ same maths apply!

MATHEMATICAL ANALYSIS

→ equations for Hopfield network are like the equations describing ‘spin glasses’
→ same maths apply!

1. network dynamics have stable fixed points

MATHEMATICAL ANALYSIS

→ equations for Hopfield network are like the equations describing 'spin glasses'
→ same maths apply!

1. network dynamics have stable fixed points

   if we can show that there exists an
   'landscape' (energy or Lyapunov) function $E(\mathbf{r})$

MATHEMATICAL ANALYSIS

→ equations for Hopfield network are like the equations describing ‘spin glasses’
→ same maths apply!

1. network dynamics have stable fixed points
   if we can show that there exists an
   ‘landscape’ (energy or Lyapunov) function $E(\mathbf{r})$

   - the dynamics of $\mathbf{r}$ are such that
     the it never moves upwards on the landscape
     $E(\mathbf{r}(t+\Delta t)) \leq E(\mathbf{r}(t))$

MATHEMATICAL ANALYSIS

→ equations for Hopfield network are like the equations describing 'spin glasses'
→ same maths apply!

1. network dynamics have stable fixed points
   if we can show that there exists an
   'landscape' (energy or Lyapunov) function $E(\mathbf{r})$

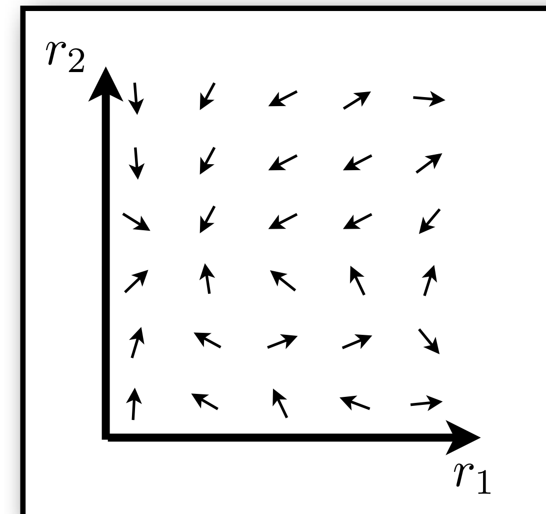
   - the dynamics of $\mathbf{r}$ are such that
     the it never moves upwards on the landscape
     $E(\mathbf{r}(t + \Delta t)) \leq E(\mathbf{r}(t))$
   - the landscape has finite depth (lower bounded)
     $E(\mathbf{r}) \geq 0$

MATHEMATICAL ANALYSIS

→ equations for Hopfield network are like the equations describing 'spin glasses'
→ same maths apply!

1. network dynamics have stable fixed points
   if we can show that there exists an
   'landscape' (energy or Lyapunov) function $E(\mathbf{r})$

   - the dynamics of $\mathbf{r}$ are such that
     the it never moves upwards on the landscape
     $E(\mathbf{r}(t+\Delta t)) \leq E(\mathbf{r}(t))$
   - the landscape has finite depth (lower bounded)
     $E(\mathbf{r}) \geq 0$



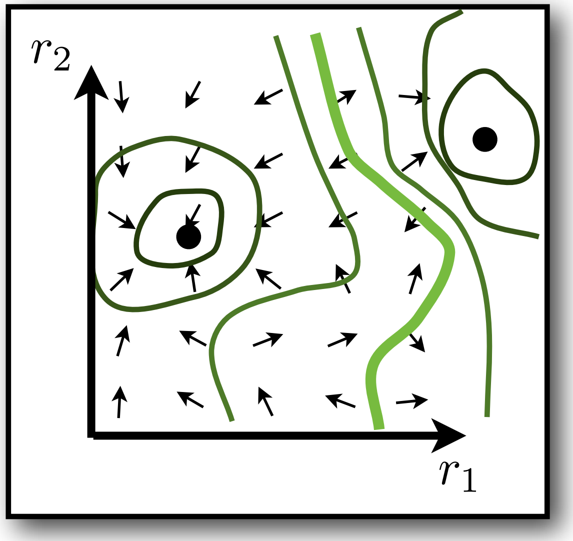
MATHEMATICAL ANALYSIS

→ equations for Hopfield network are like the equations describing 'spin glasses'
→ same maths apply!

1. network dynamics have stable fixed points

   if we can show that there exists an
   'landscape' (energy or Lyapunov) function $E(\mathbf{r})$

   - the dynamics of $\mathbf{r}$ are such that
     the it never moves upwards on the landscape
     $E(\mathbf{r}(t+\Delta t)) \leq E(\mathbf{r}(t))$
   - the landscape has finite depth (lower bounded)
     $E(\mathbf{r}) \geq 0$



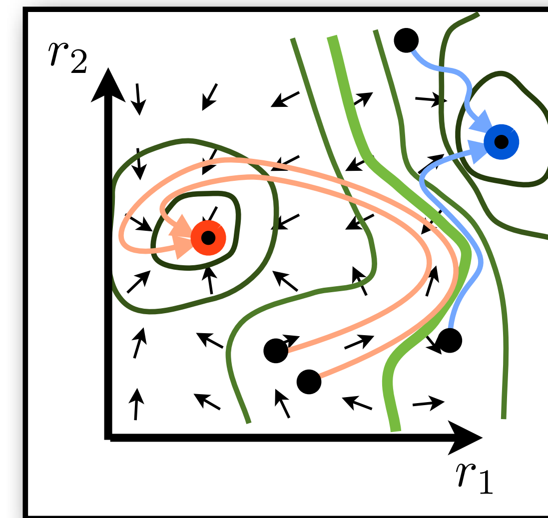
MATHEMATICAL ANALYSIS

→ equations for Hopfield network are like the equations describing 'spin glasses'
→ same maths apply!

1. network dynamics have stable fixed points

   if we can show that there exists an
   'landscape' (energy or Lyapunov) function $E(\mathbf{r})$

   - the dynamics of $\mathbf{r}$ are such that
     the it never moves upwards on the landscape
     $E(\mathbf{r}(t + \Delta t)) \leq E(\mathbf{r}(t))$
   - the landscape has finite depth (lower bounded)
     $E(\mathbf{r}) \geq 0$



MATHEMATICAL ANALYSIS

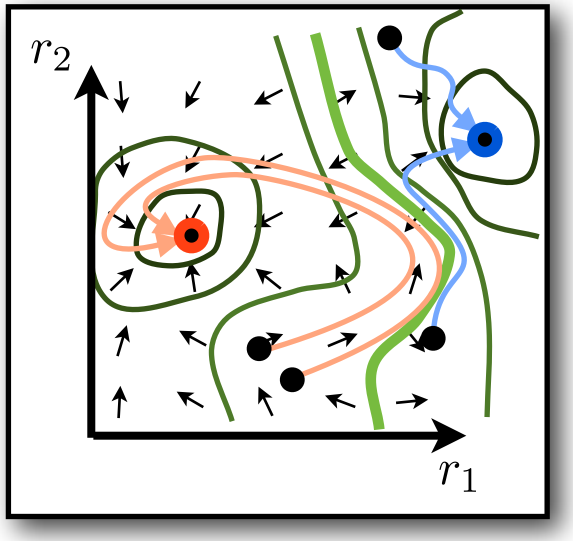
→ equations for Hopfield network are like the equations describing 'spin glasses'
→ same maths apply!

1. network dynamics have stable fixed points

   if we can show that there exists an
   'landscape' (energy or Lyapunov) function $E(\mathbf{r})$

   - the dynamics of $\mathbf{r}$ are such that the it never moves upwards on the landscape
     $E(\mathbf{r}(t + \Delta t)) \leq E(\mathbf{r}(t))$
   - the landscape has finite depth (lower bounded)
     $E(\mathbf{r}) \geq 0$

in reality:
fixed points only in corners



MATHEMATICAL ANALYSIS

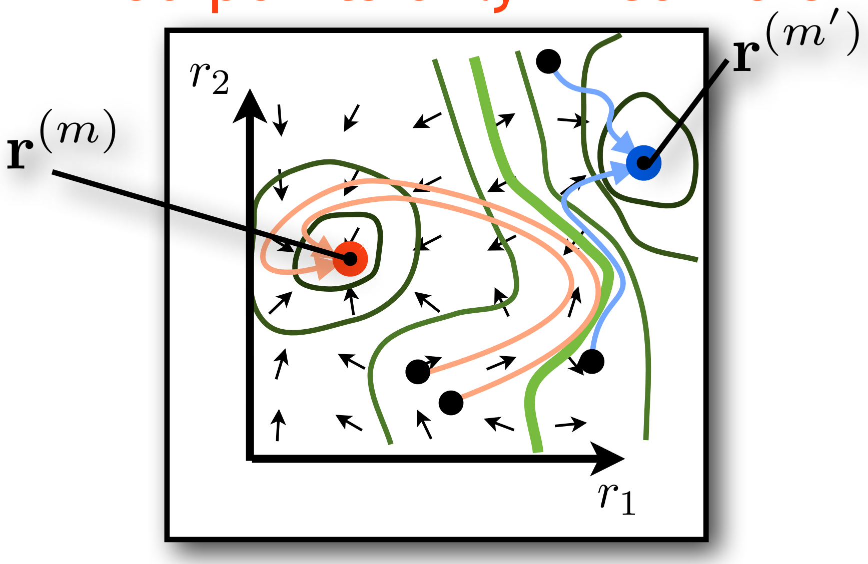
→ equations for Hopfield network are like the equations describing ‘spin glasses’
→ same maths apply!

1. network dynamics have stable fixed points

   if we can show that there exists an
   ‘landscape’ (energy or Lyapunov) function $E(\mathbf{r})$

   - the dynamics of $\mathbf{r}$ are such that
     the it never moves upwards on the landscape
     $E(\mathbf{r}(t+\Delta t)) \leq E(\mathbf{r}(t))$
   - the landscape has finite depth (lower bounded)
     $E(\mathbf{r}) \geq 0$

in reality:
fixed points only in corners

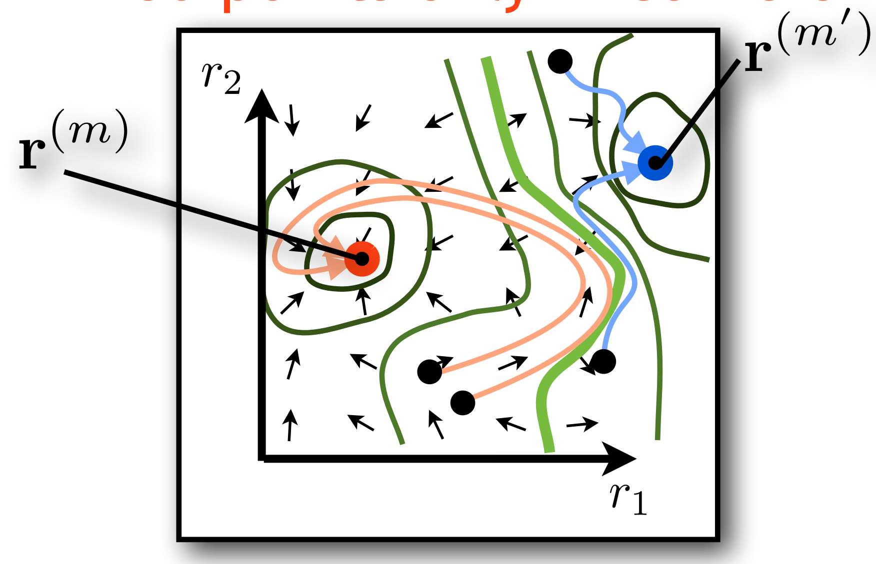


2. stored patterns are fixed points

MATHEMATICAL ANALYSIS

→ equations for Hopfield network are like the equations describing ‘spin glasses’
→ same maths apply!

in reality:
fixed points only in corners



1. network dynamics have stable fixed points

   if we can show that there exists an
   ‘landscape’ (energy or Lyapunov) function $E(\mathbf{r})$

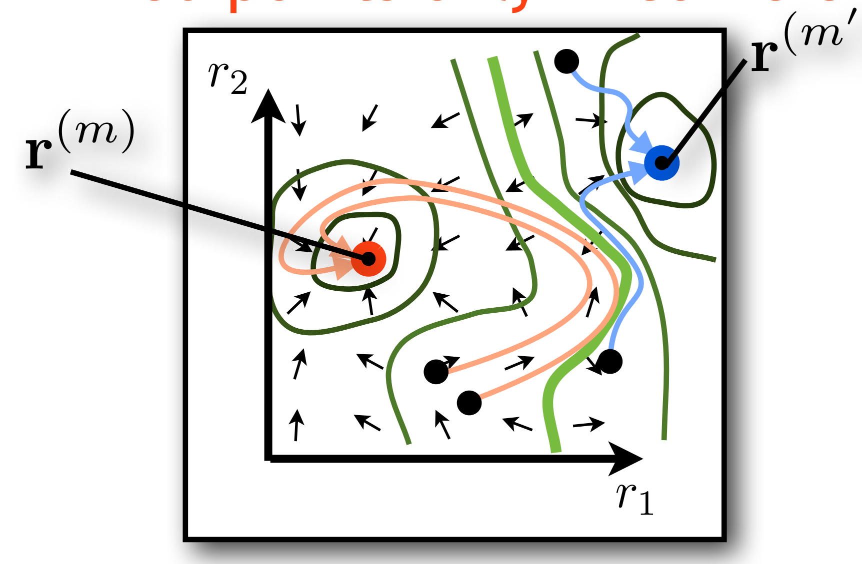
   - the dynamics of $\mathbf{r}$ are such that
     the it never moves upwards on the landscape
     $E(\mathbf{r}(t + \Delta t)) \leq E(\mathbf{r}(t))$
   - the landscape has finite depth (lower bounded)
     $E(\mathbf{r}) \geq 0$

stable?
2. stored patterns are fixed points

MATHEMATICAL ANALYSIS

→ equations for Hopfield network are like the equations describing 'spin glasses'
→ same maths apply!

in reality:
fixed points only in corners



1. network dynamics have stable fixed points

   if we can show that there exists an
   'landscape' (energy or Lyapunov) function $E(\mathbf{r})$

   - the dynamics of $\mathbf{r}$ are such that
     the it never moves upwards on the landscape
     $E(\mathbf{r}(t + \Delta t)) \leq E(\mathbf{r}(t))$
   - the landscape has finite depth (lower bounded)
     $E(\mathbf{r}) \geq 0$

stable?

2. stored patterns are fixed points

   sources of recall errors:

   - stored patterns are unstable fixed points

MATHEMATICAL ANALYSIS

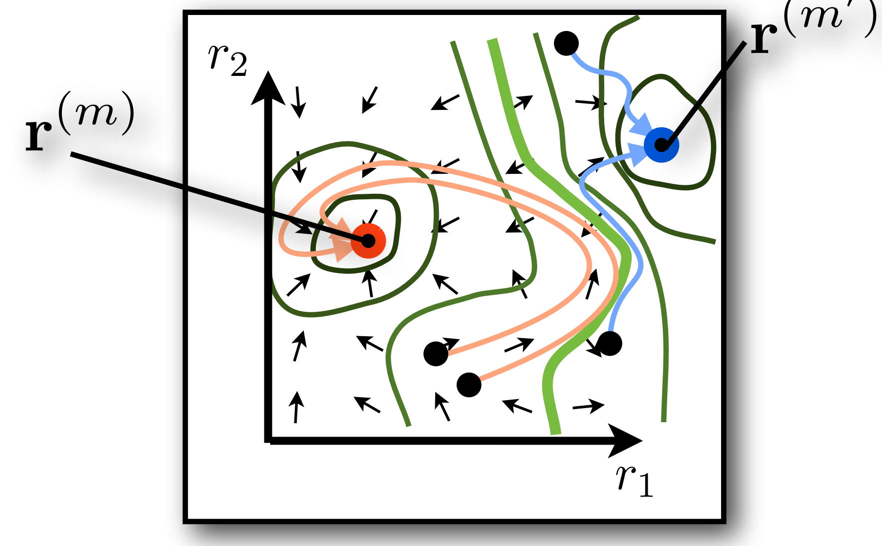
→ equations for Hopfield network are like the equations describing 'spin glasses'
→ same maths apply!

1. network dynamics have stable fixed points

   if we can show that there exists an
   'landscape' (energy or Lyapunov) function $E(\mathbf{r})$

   - the dynamics of $\mathbf{r}$ are such that
     the it never moves upwards on the landscape
     $E(\mathbf{r}(t + \Delta t)) \leq E(\mathbf{r}(t))$
   - the landscape has finite depth (lower bounded)
     $E(\mathbf{r}) \geq 0$

in reality:
fixed points only in corners



stable?

2. stored patterns are fixed points

sources of recall errors:

- stored patterns are unstable fixed points
- stored patterns are not fixed points

MATHEMATICAL ANALYSIS

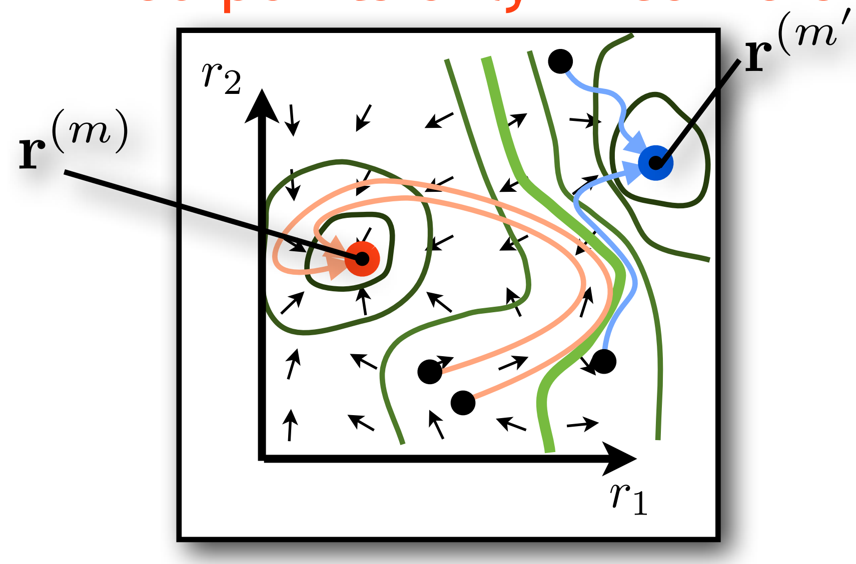
→ equations for Hopfield network are like the equations describing ‘spin glasses’
→ same maths apply!

1. network dynamics have stable fixed points

   if we can show that there exists an
   ‘landscape’ (energy or Lyapunov) function $E(\mathbf{r})$

   - the dynamics of $\mathbf{r}$ are such that
     the it never moves upwards on the landscape
     $E(\mathbf{r}(t+\Delta t)) \leq E(\mathbf{r}(t))$
   - the landscape has finite depth (lower bounded)
     $E(\mathbf{r}) \geq 0$

in reality:
fixed points only in corners



stable?

2. stored patterns are fixed points on average

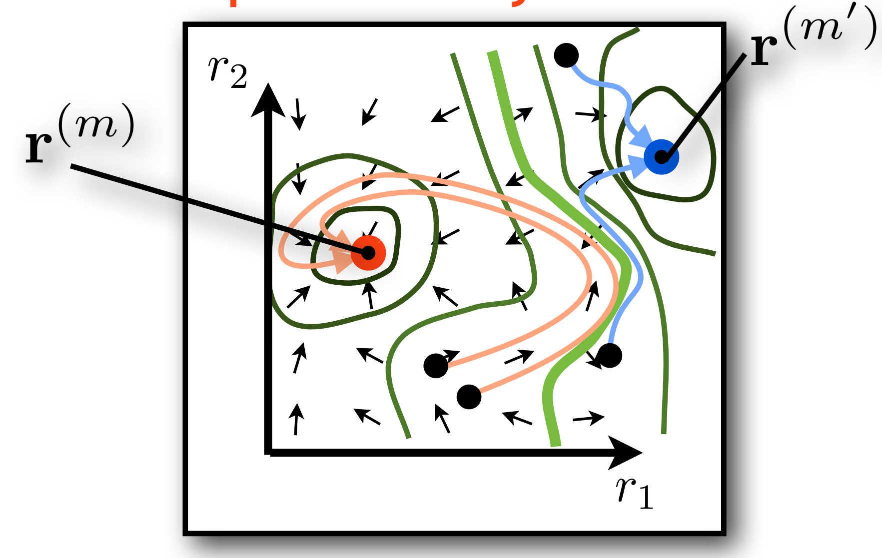
sources of recall errors:

- stored patterns are unstable fixed points
- stored patterns are not fixed points

MATHEMATICAL ANALYSIS

→ equations for Hopfield network are like the equations describing 'spin glasses'
→ same maths apply!

in reality:
fixed points only in corners



1. network dynamics have stable fixed points

   if we can show that there exists an **'landscape' (energy or Lyapunov) function** $E(\mathbf{r})$

   - the dynamics of $\mathbf{r}$ are such that the it never moves upwards on the landscape
     $E(\mathbf{r}(t + \Delta t)) \leq E(\mathbf{r}(t))$
   - the landscape has finite depth (lower bounded)
     $E(\mathbf{r}) \geq 0$

stable?

2. stored patterns are fixed points on average

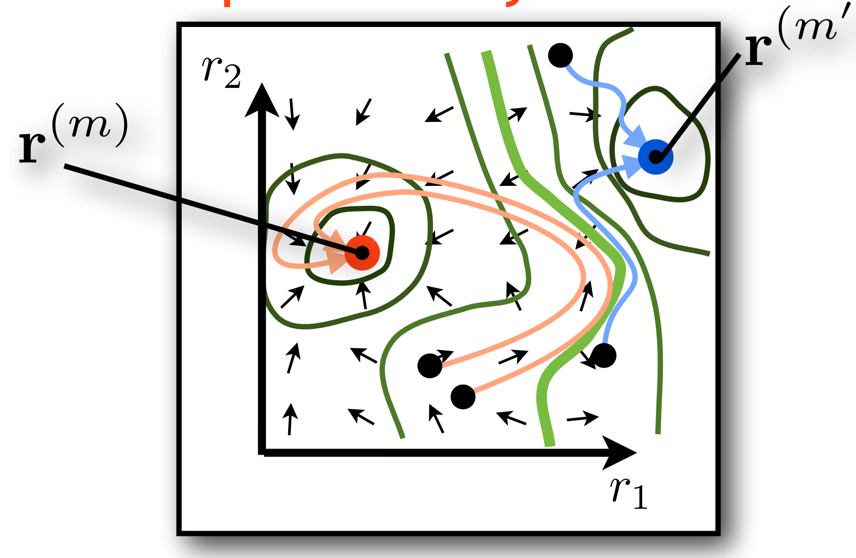
   sources of recall errors:

   - stored patterns are unstable fixed points
   - stored patterns are not fixed points
   - spurious attractors

MATHEMATICAL ANALYSIS

→ equations for Hopfield network are like the equations describing 'spin glasses'
→ same maths apply!

in reality:
fixed points only in corners



1. network dynamics have stable fixed points

   if we can show that there exists an
   'landscape' (energy or Lyapunov) function $E(\mathbf{r})$

   - the dynamics of $\mathbf{r}$ are such that
     the it never moves upwards on the landscape
     $E(\mathbf{r}(t + \Delta t)) \leq E(\mathbf{r}(t))$
   - the landscape has finite depth (lower bounded)
     $E(\mathbf{r}) \geq 0$

2. stored patterns are fixed points on average
   stable?
   (other stable fixed points?)

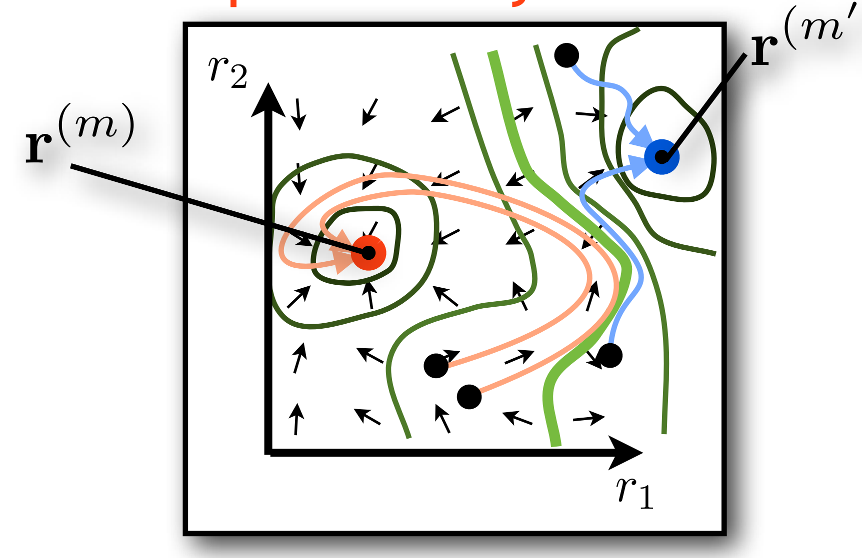
   sources of recall errors:

   - stored patterns are unstable fixed points
   - stored patterns are not fixed points
   - spurious attractors

MATHEMATICAL ANALYSIS

→ equations for Hopfield network are like the equations describing ‘spin glasses’
→ same maths apply!

in reality:
fixed points only in corners



1. network dynamics have stable fixed points

   if we can show that there exists an
   ‘landscape’ (energy or Lyapunov) function $E(\mathbf{r})$

   - the dynamics of $\mathbf{r}$ are such that the it never moves upwards on the landscape
     $E(\mathbf{r}(t + \Delta t)) \leq E(\mathbf{r}(t))$
   - the landscape has finite depth (lower bounded)
     $E(\mathbf{r}) \geq 0$

2. stored patterns are fixed points stable? on average
   (other stable fixed points?)

possible to analyse mathematically: capacity, phase transitions, etc.

sources of recall errors:

- stored patterns are unstable fixed points
- stored patterns are not fixed points
- spurious attractors

MATHEMATICAL ANALYSIS

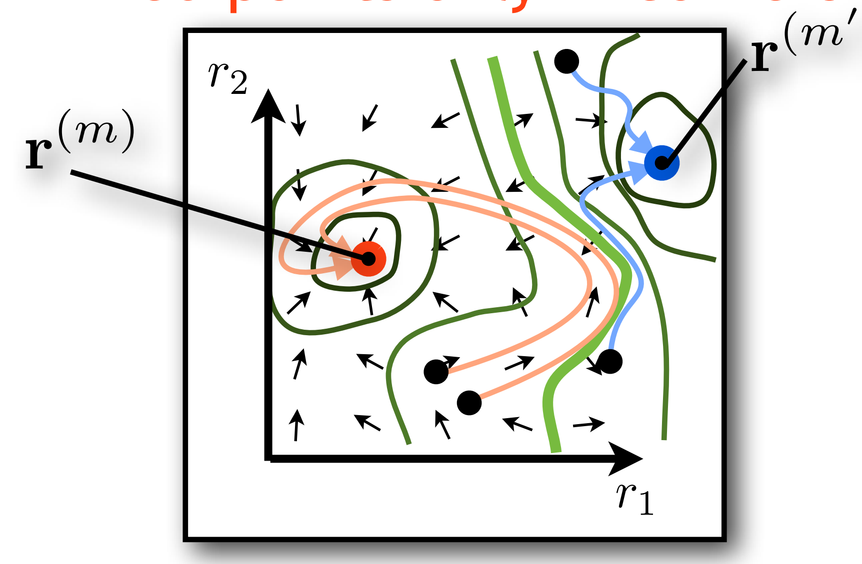
→ equations for Hopfield network are like the equations describing 'spin glasses'
→ same maths apply!

1. network dynamics have stable fixed points

   if we can show that there exists an
   'landscape' (energy or Lyapunov) function $E(\mathbf{r})$

   - the dynamics of $\mathbf{r}$ are such that
     the it never moves upwards on the landscape
     $E(\mathbf{r}(t + \Delta t)) \leq E(\mathbf{r}(t))$
   - the landscape has finite depth (lower bounded)
     $E(\mathbf{r}) \geq 0$

in reality:
fixed points only in corners



2. stored patterns are fixed points on average (stable?)
   (other stable fixed points?)

   sources of recall errors:

   - stored patterns are unstable fixed points
   - stored patterns are not fixed points
   - spurious attractors

possible to analyse mathematically:
capacity, phase transitions, etc.

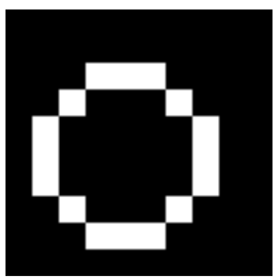
e.g. capacity is determined by

$$\frac{\text{number of stored patterns}}{\text{number of synapses/cell}}$$

HOPFIELD NETWORK IN OPERATION

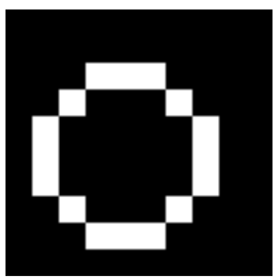
HOPFIELD NETWORK IN OPERATION

stored patterns



HOPFIELD NETWORK IN OPERATION

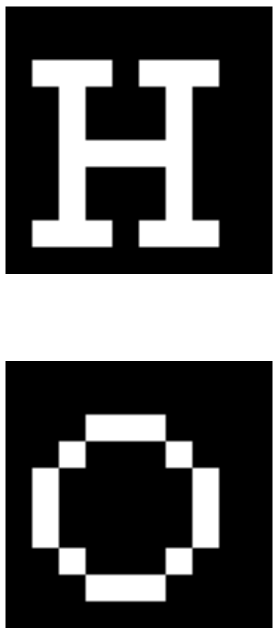
stored patterns



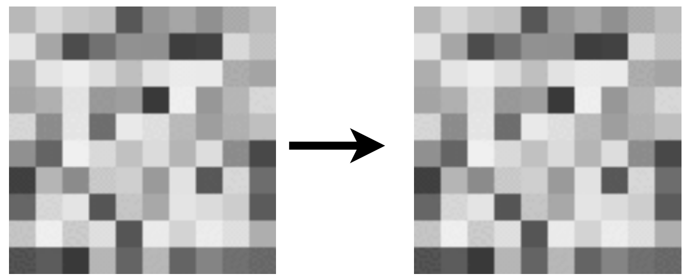
noisy input

HOPFIELD NETWORK IN OPERATION

stored patterns

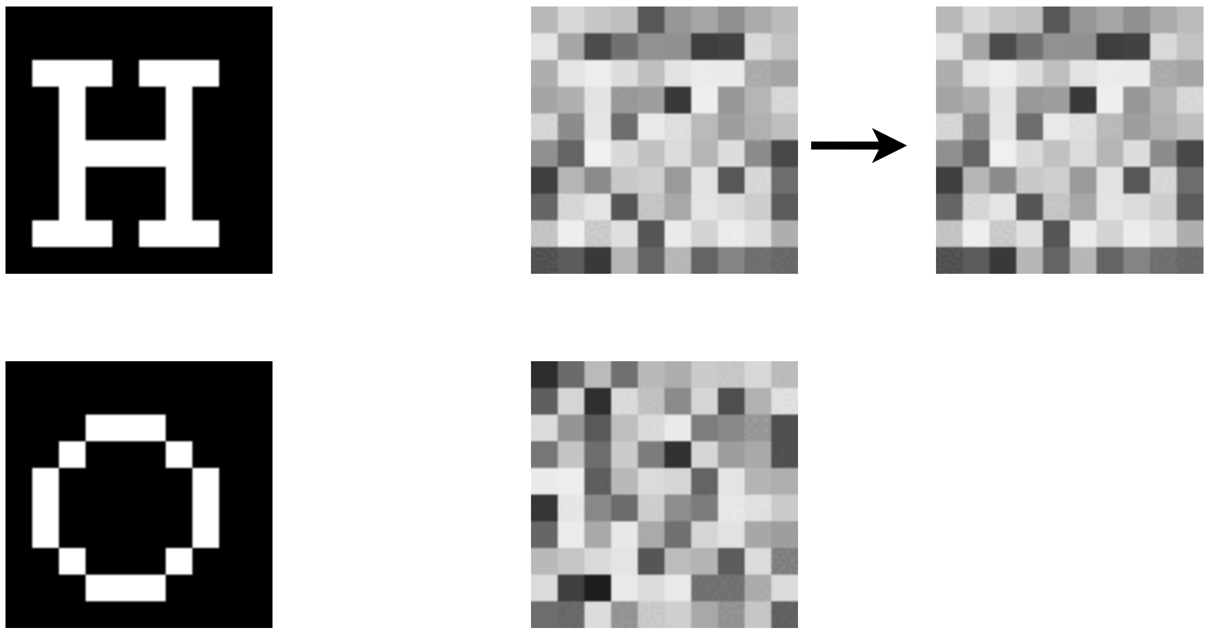


noisy input



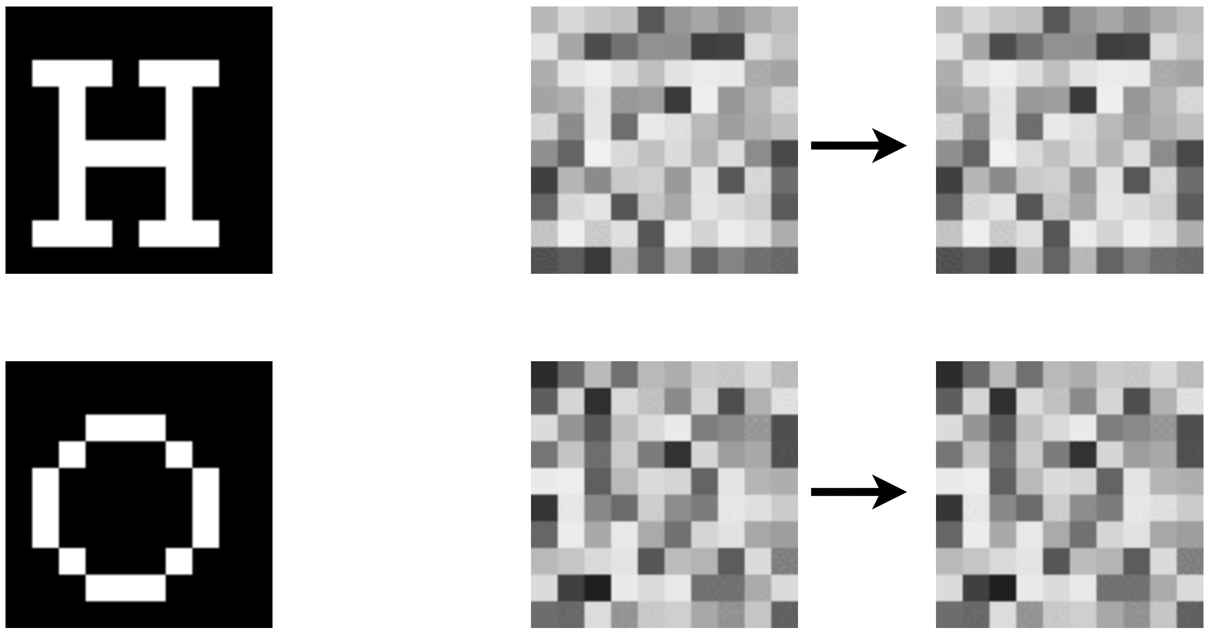
HOPFIELD NETWORK IN OPERATION

stored patterns   noisy input



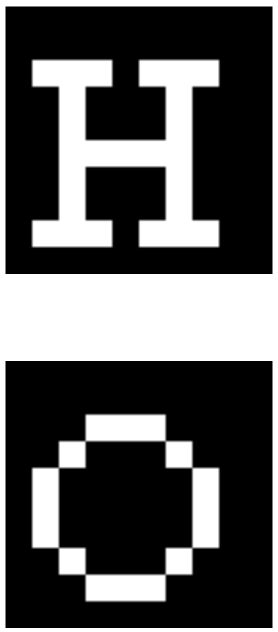
HOPFIELD NETWORK IN OPERATION

stored patterns   noisy input

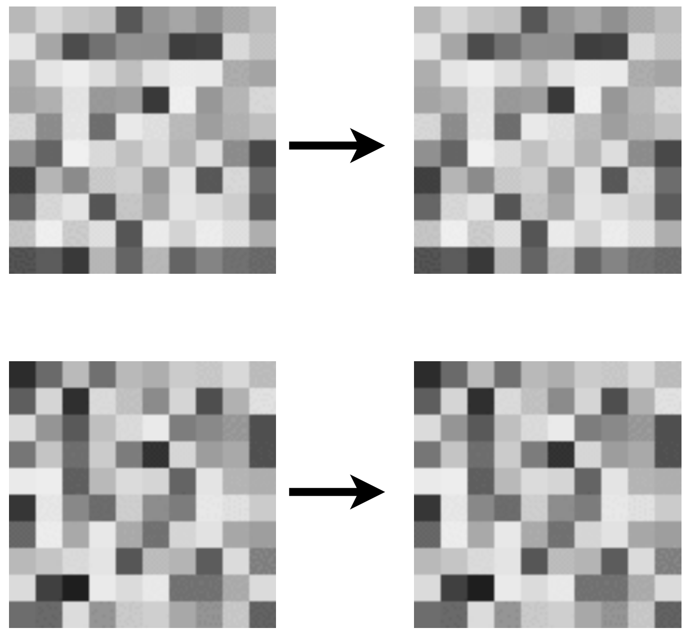


HOPFIELD NETWORK IN OPERATION

stored patterns

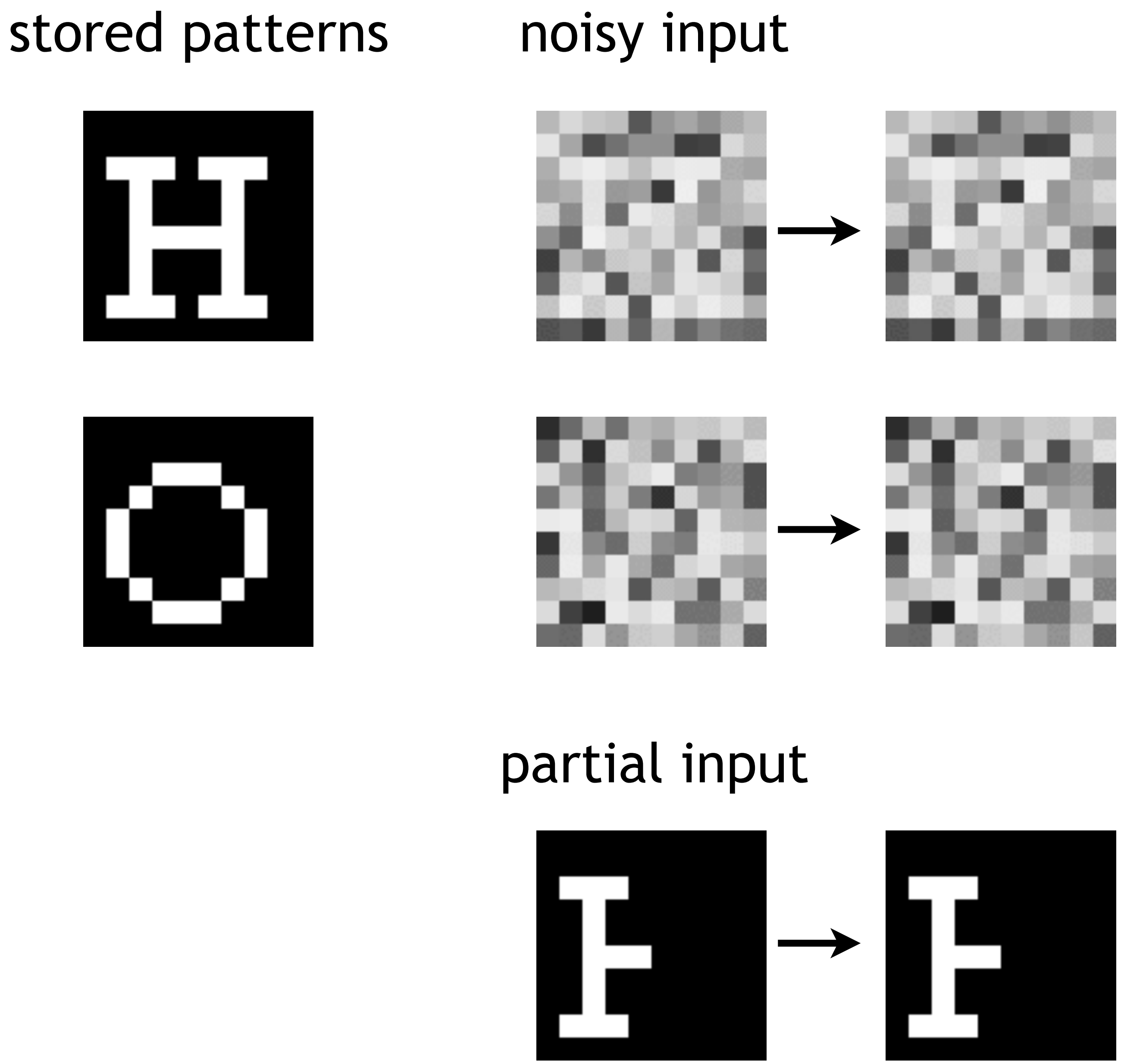


noisy input



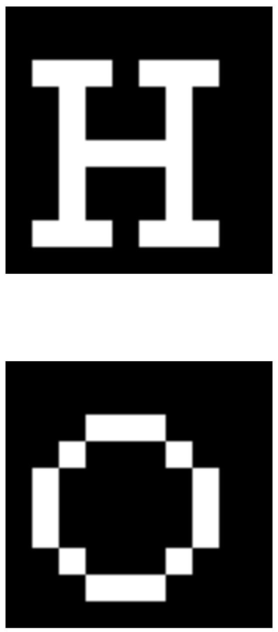
partial input

HOPFIELD NETWORK IN OPERATION

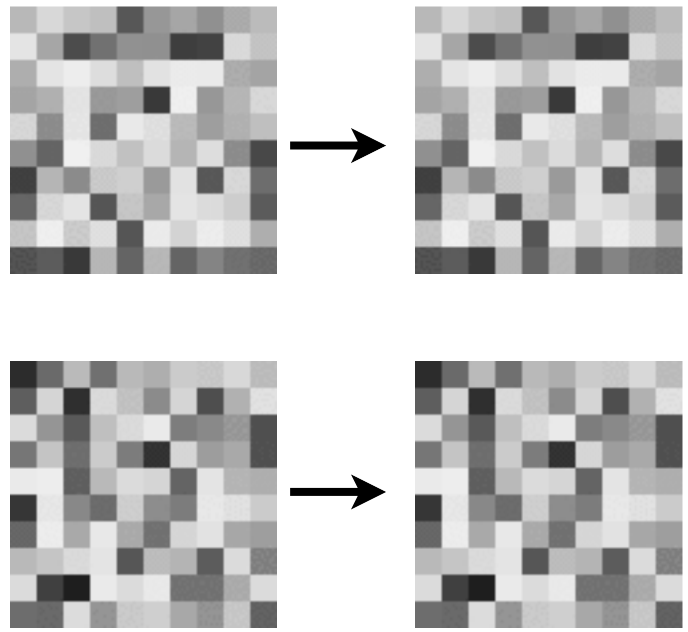


HOPFIELD NETWORK IN OPERATION

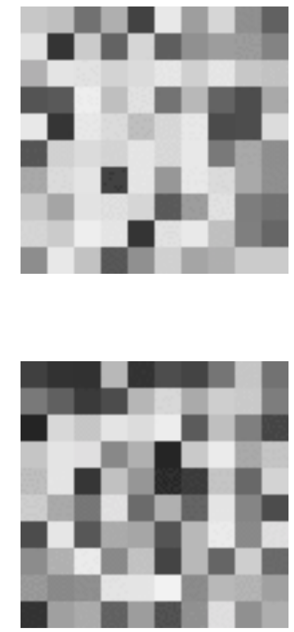
stored patterns



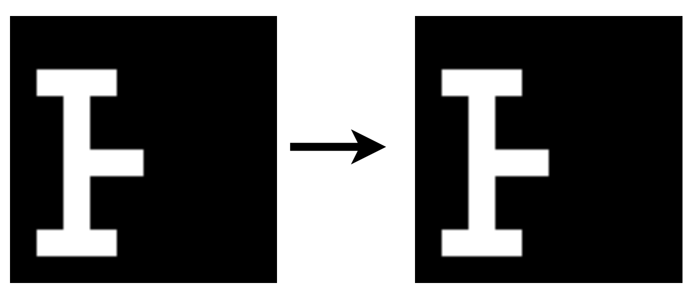
noisy input



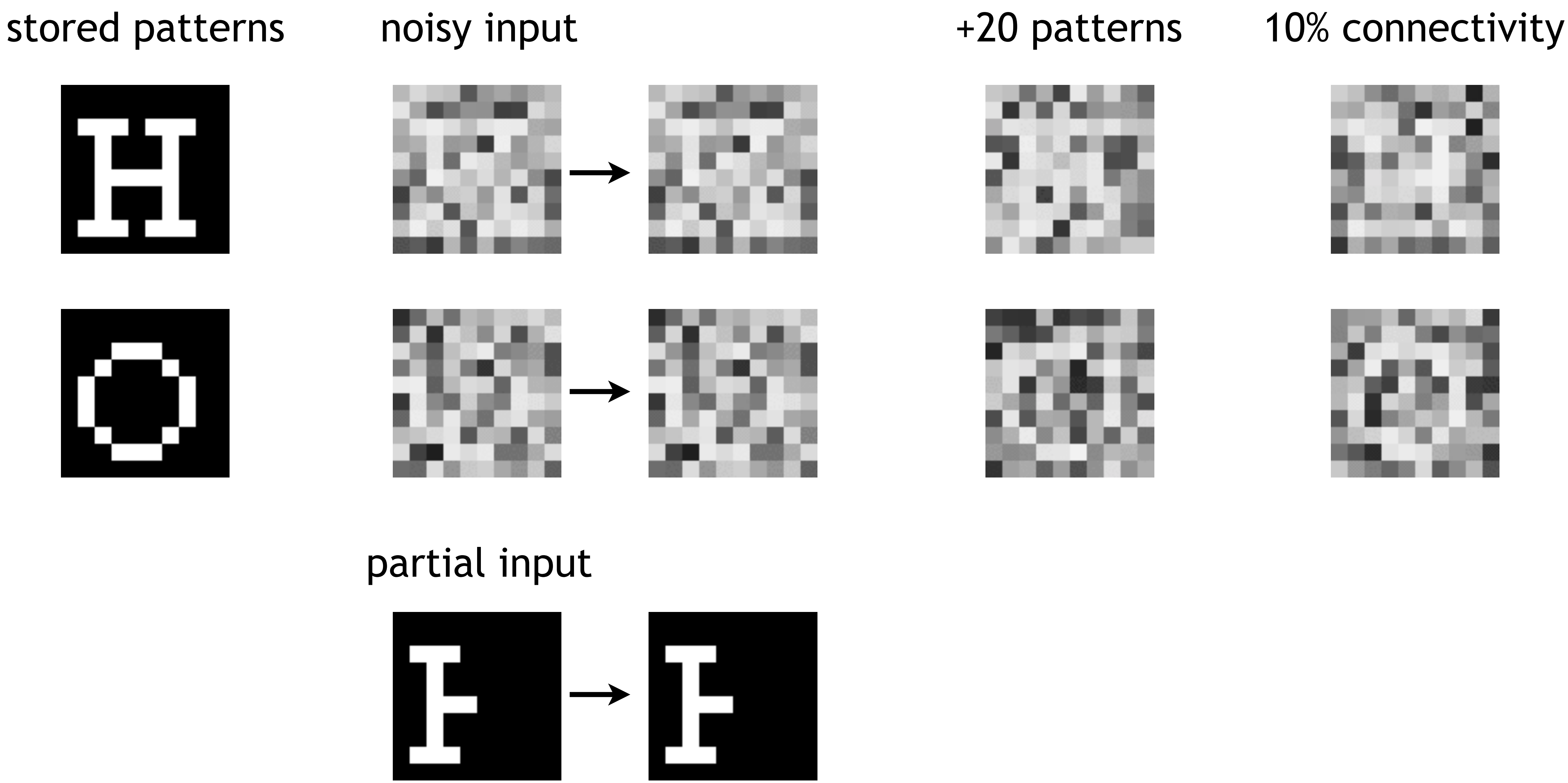
+20 patterns



partial input



HOPFIELD NETWORK IN OPERATION



MEMORY PROCESSING IN NEURAL NETWORKS

MEMORY PROCESSING IN NEURAL NETWORKS

the Hopfieldian paradigm

MEMORY PROCESSING IN NEURAL NETWORKS

the Hopfieldian paradigm

- memories are represented as *distributed* patterns of activity

MEMORY PROCESSING IN NEURAL NETWORKS

the Hopfieldian paradigm

- memories are represented as *distributed* patterns of activity
- assume particular *learning rule*

MEMORY PROCESSING IN NEURAL NETWORKS

the Hopfieldian paradigm

- memories are represented as *distributed* patterns of activity
- assume particular *learning rule*
- assume particular *network dynamics*

MEMORY PROCESSING IN NEURAL NETWORKS

the Hopfieldian paradigm

- memories are represented as *distributed* patterns of activity
- assume particular *learning rule*
- assume particular *network dynamics*
- show that the two together work appropriately → *energy function*

MEMORY PROCESSING IN NEURAL NETWORKS

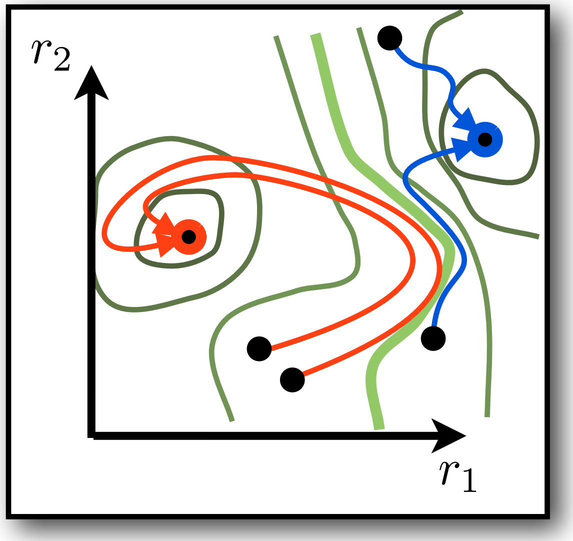
the Hopfieldian paradigm

- memories are represented as *distributed* patterns of activity
- assume particular *learning rule*
- assume particular *network dynamics*
- show that the two together work appropriately → *energy function*
  - stored patterns correspond to (point) *attractors*

MEMORY PROCESSING IN NEURAL NETWORKS

the Hopfieldian paradigm

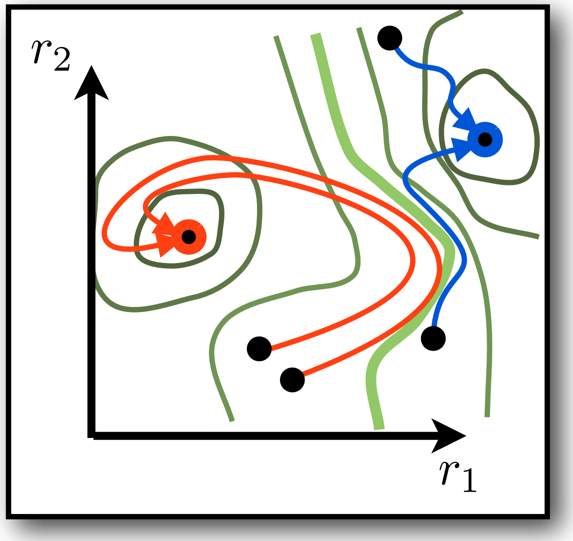
- memories are represented as *distributed* patterns of activity
- assume particular *learning rule*
- assume particular *network dynamics*
- show that the two together work appropriately → *energy function*
  - stored patterns correspond to (point) *attractors*



MEMORY PROCESSING IN NEURAL NETWORKS

the Hopfieldian paradigm

- memories are represented as *distributed* patterns of activity
- assume particular *learning rule*
- assume particular *network dynamics*
- show that the two together work appropriately → *energy function*
  - stored patterns correspond to (point) *attractors*

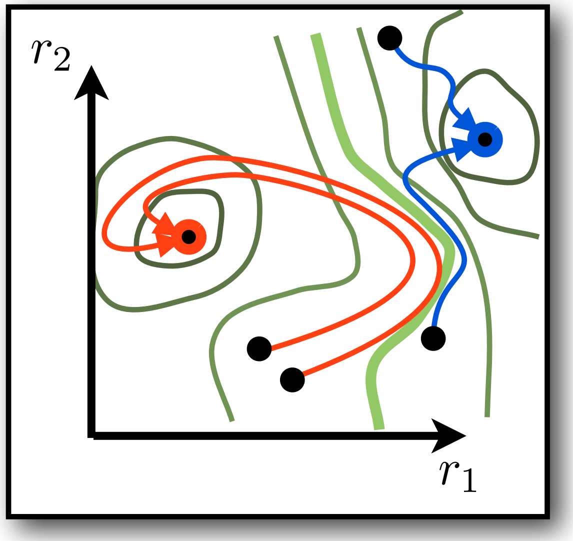


challenges:

MEMORY PROCESSING IN NEURAL NETWORKS

the Hopfieldian paradigm

- memories are represented as *distributed* patterns of activity
- assume particular *learning rule*
- assume particular *network dynamics*
- show that the two together work appropriately → *energy function*
    - stored patterns correspond to (point) *attractors*



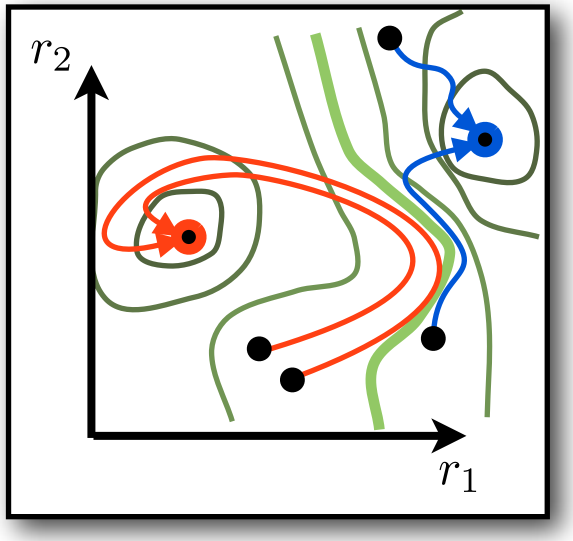
challenges:

- general theory (different learning rules, network dynamics) : hard to find the corresponding energy function

MEMORY PROCESSING IN NEURAL NETWORKS

the Hopfieldian paradigm

- memories are represented as *distributed* patterns of activity
- assume particular *learning rule*
- assume particular *network dynamics*
- show that the two together work appropriately → *energy function*
  - stored patterns correspond to (point) *attractors*



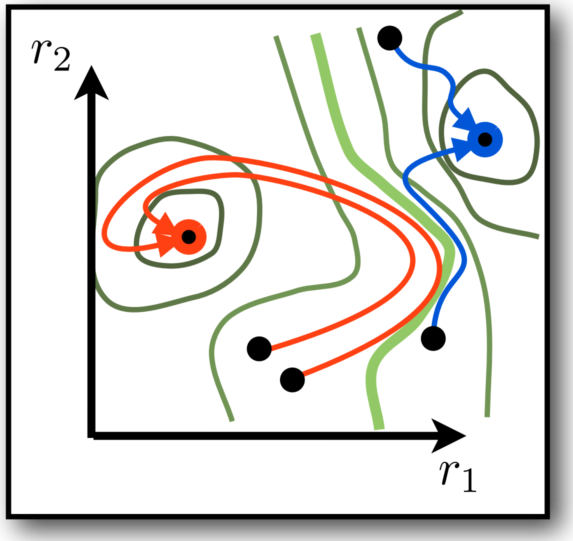
challenges:

- ❖ general theory (different learning rules, network dynamics) : hard to find the corresponding energy function
- ❖ analogue-valued memories

MEMORY PROCESSING IN NEURAL NETWORKS

the Hopfieldian paradigm

- memories are represented as *distributed* patterns of activity
- assume particular *learning rule*
- assume particular *network dynamics*
- show that the two together work appropriately → *energy function*
  - stored patterns correspond to (point) *attractors*

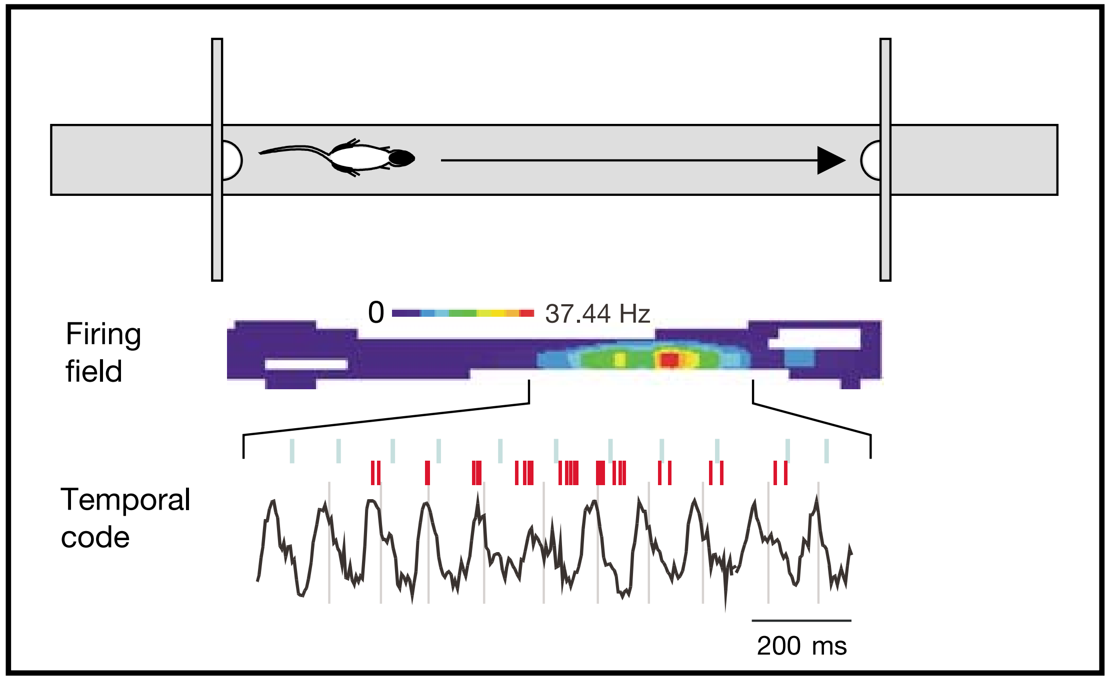


challenges:

- general theory (different learning rules, network dynamics) : hard to find the corresponding energy function
- analogue-valued memories
- spike timings, oscillations

THE IMPORTANCE OF SPIKE TIMINGS IN THE HIPPOCAMPUS

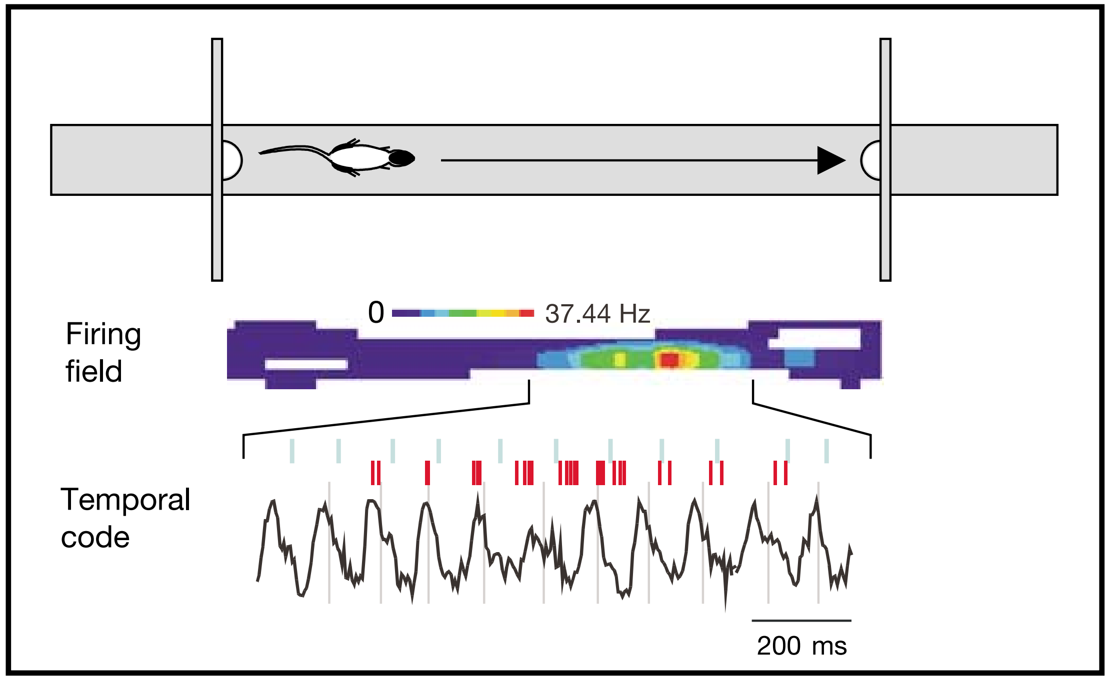
THE IMPORTANCE OF SPIKE TIMINGS IN THE HIPPOCAMPUS



Huxter et al, 1993

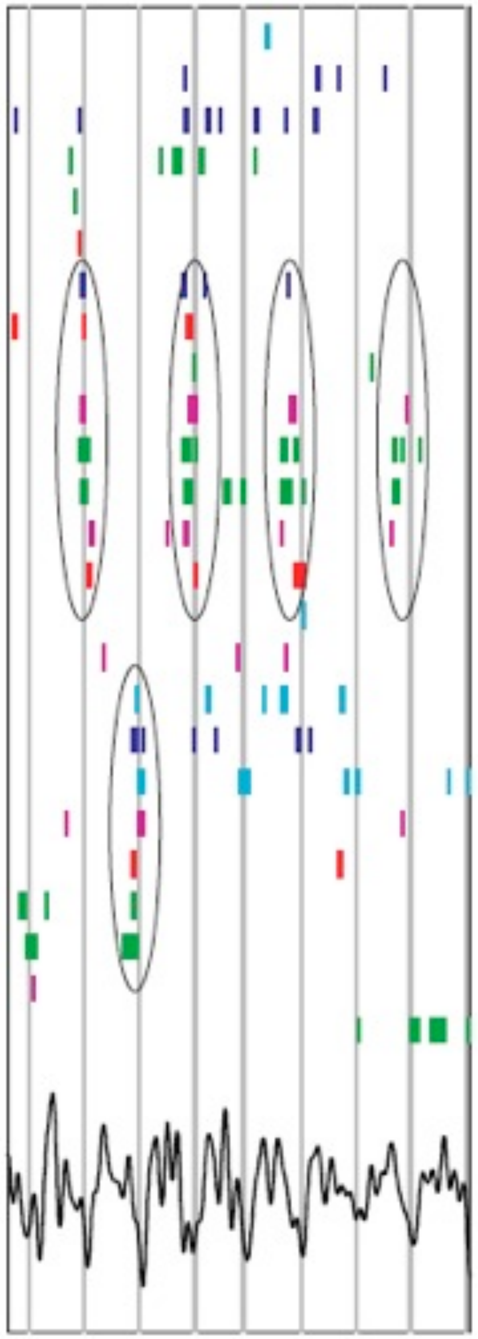
spike times convey
spatial information

THE IMPORTANCE OF SPIKE TIMINGS IN THE HIPPOCAMPUS



Huxter et al, 1993

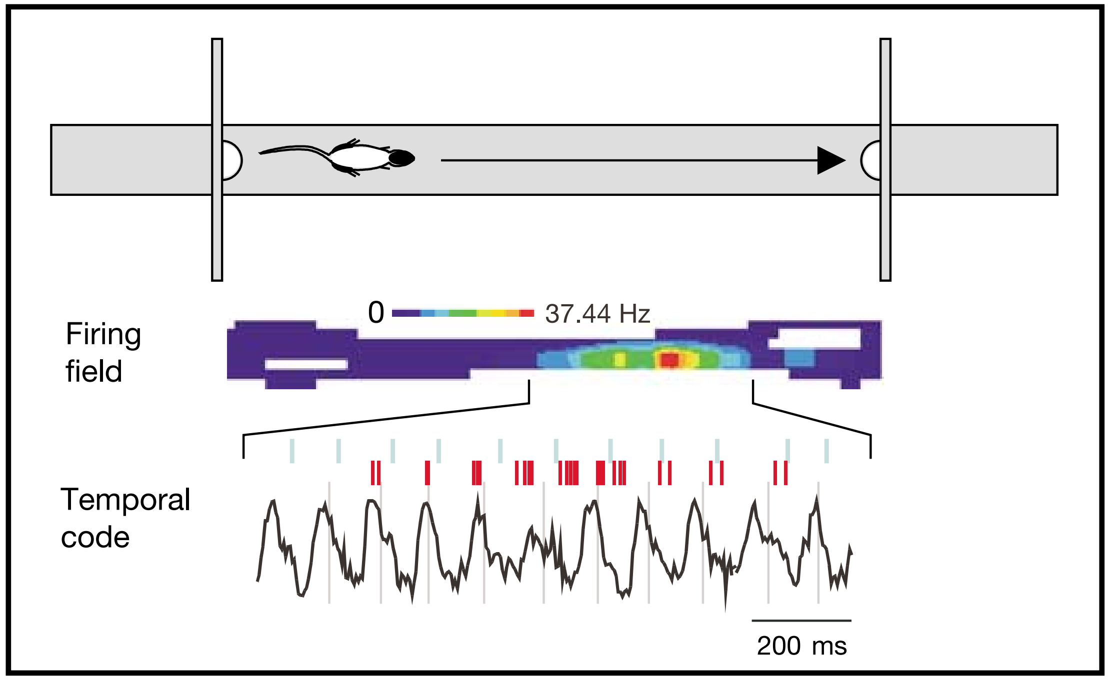
spike times convey spatial information



Harris & al, 2003

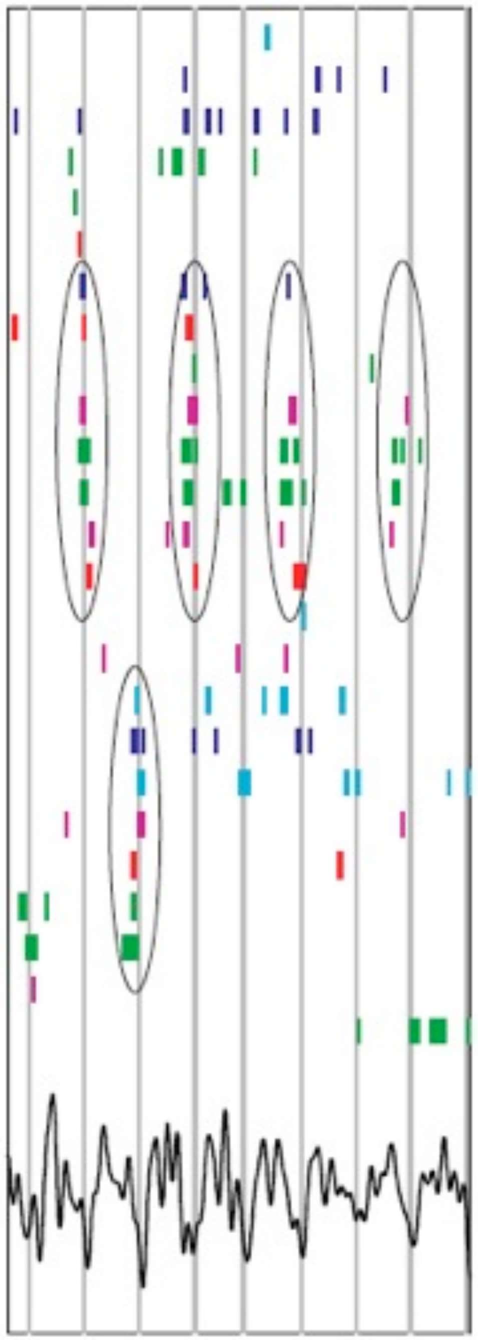
spatio-temporal firing patterns consistently reappear during awake behavior

THE IMPORTANCE OF SPIKE TIMINGS IN THE HIPPOCAMPUS



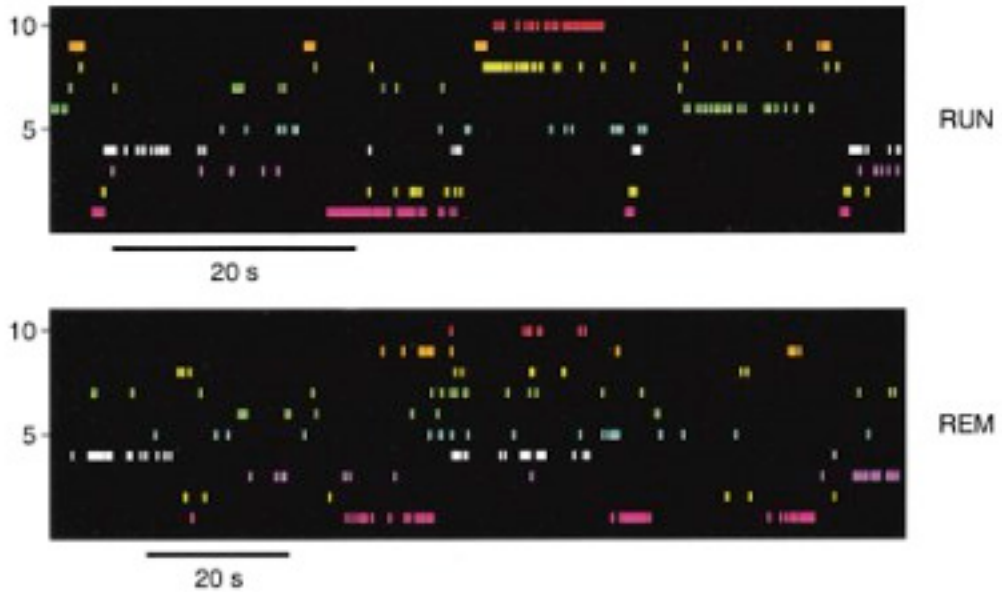
Huxter et al, 1993

spike times convey spatial information



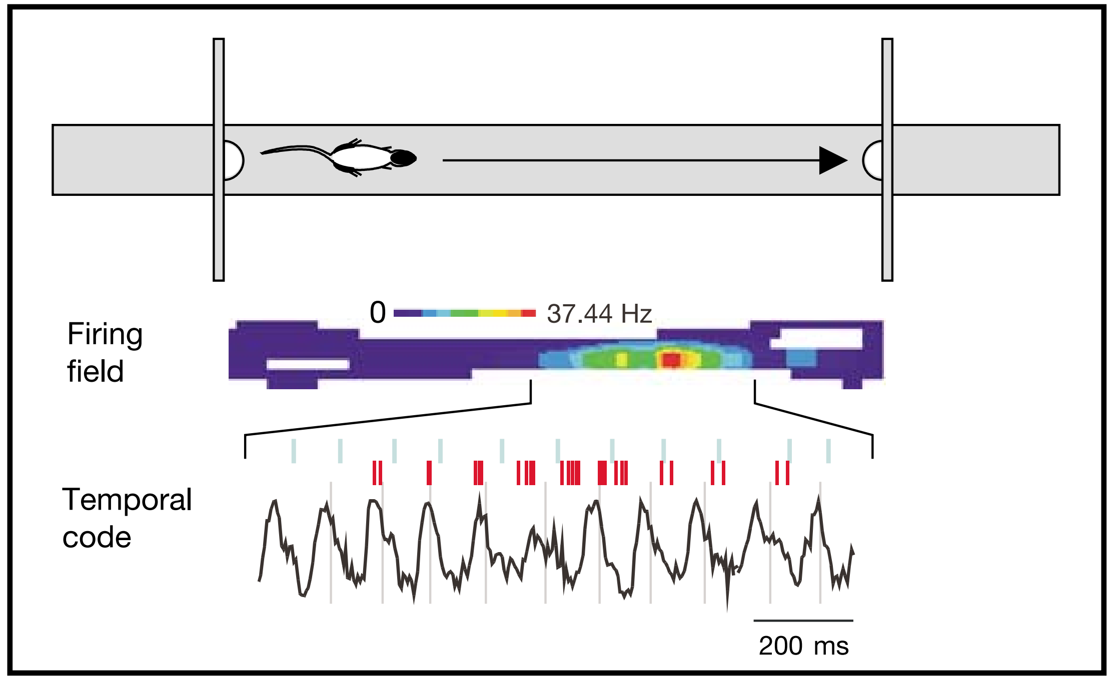
Harris & al, 2003

spatio-temporal firing patterns consistently reappear during awake behavior ... and sleep



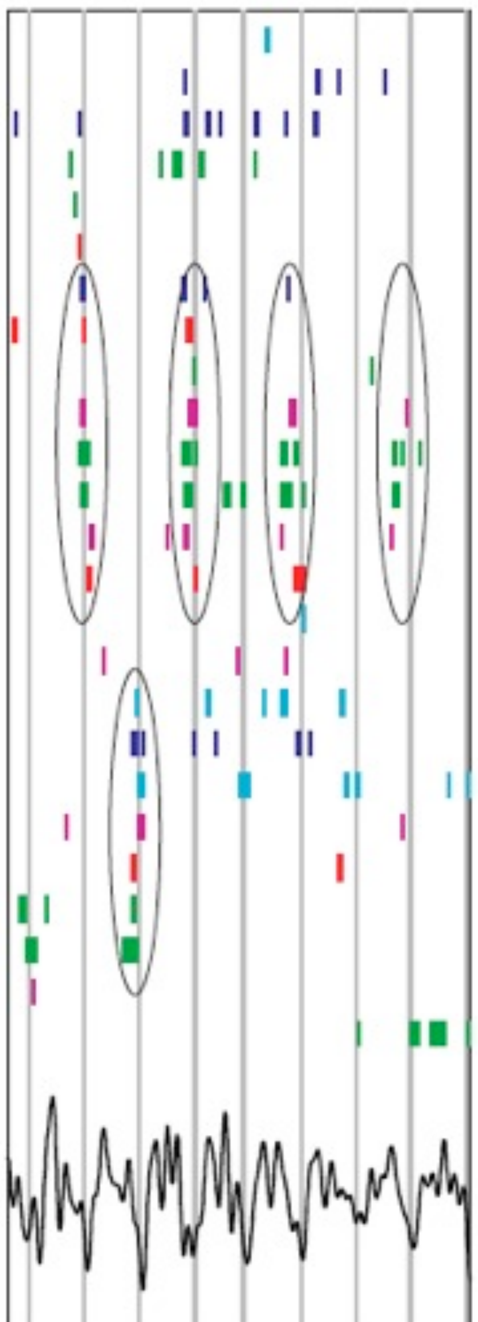
Louie and Wilson, 2001

THE IMPORTANCE OF SPIKE TIMINGS IN THE HIPPOCAMPUS



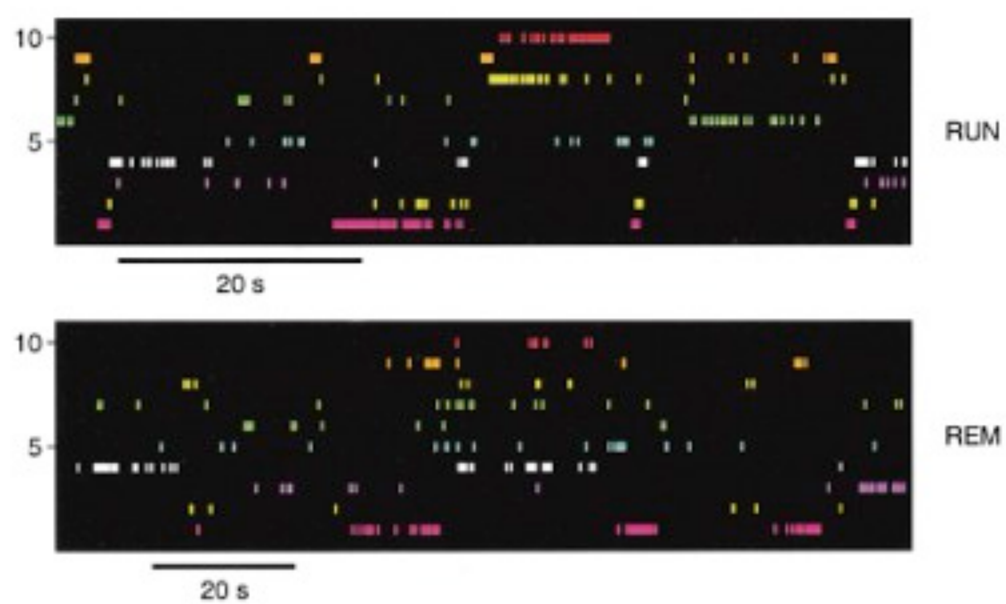
spike times convey spatial information

Huxter et al, 1993



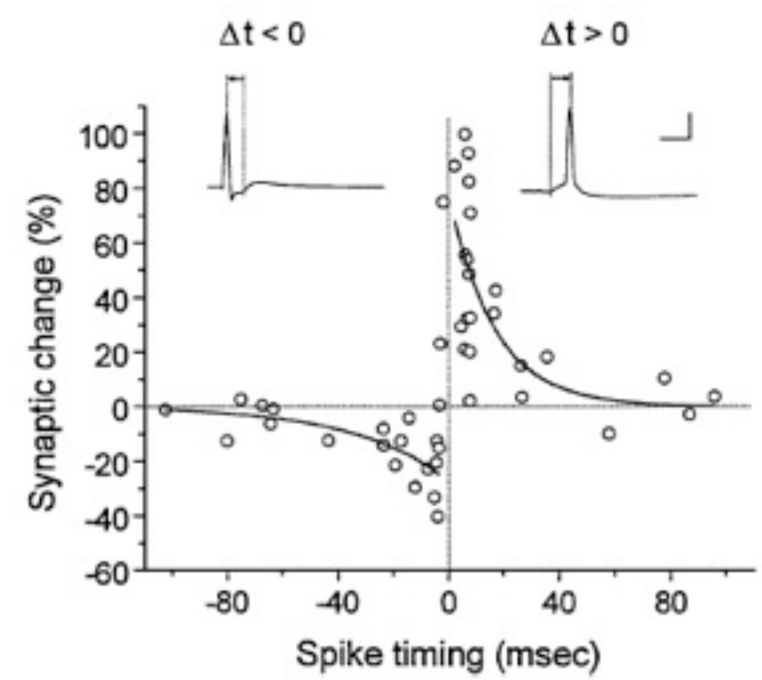
Harris & al, 2003

spatio-temporal firing patterns consistently reappear during awake behavior ... and sleep



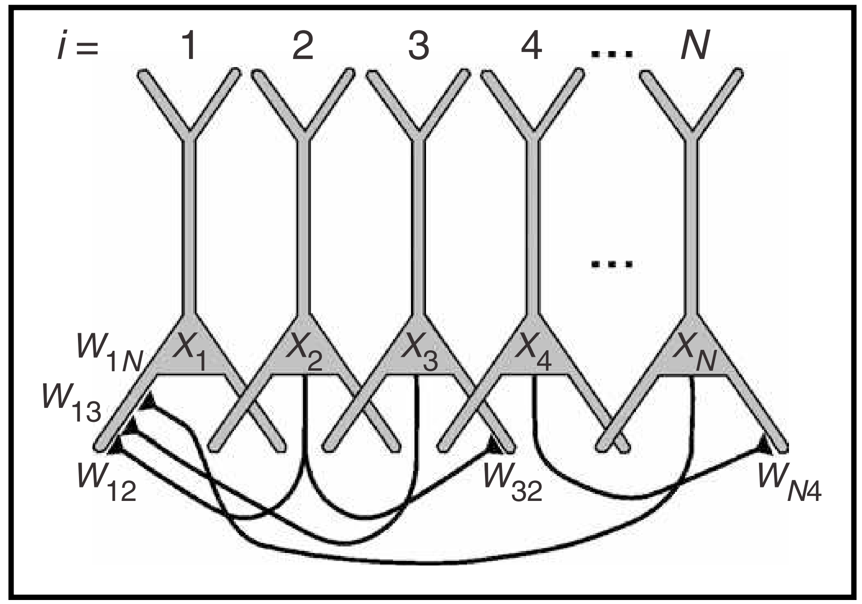
Louie and Wilson, 2001

synaptic plasticity depends on relative spike timings



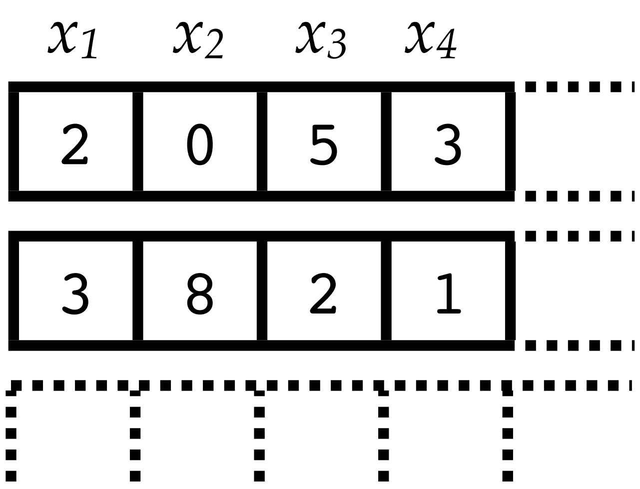
Bi & Poo, 2001

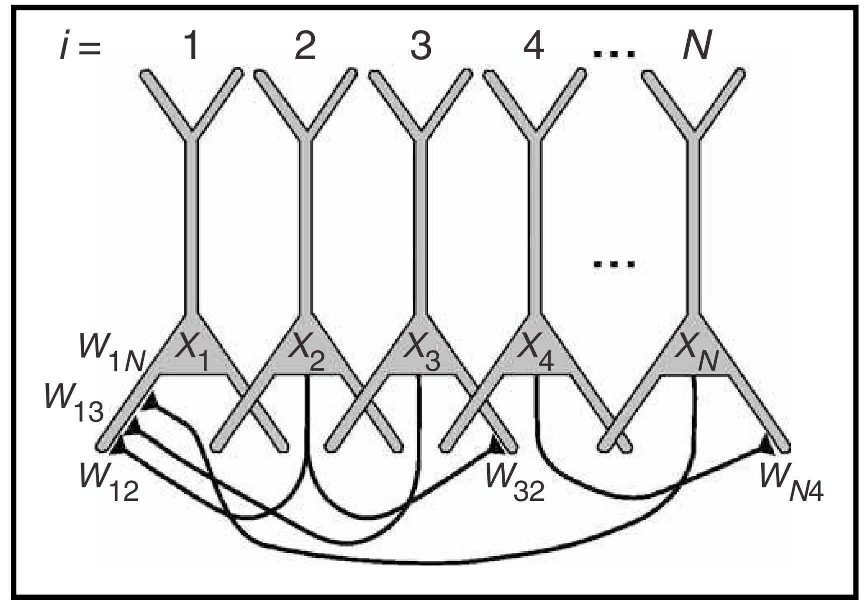
MEMORY RETRIEVAL AS PROBABILISTIC INFERENCE



MEMORY RETRIEVAL AS PROBABILISTIC INFERENCE

stored activities

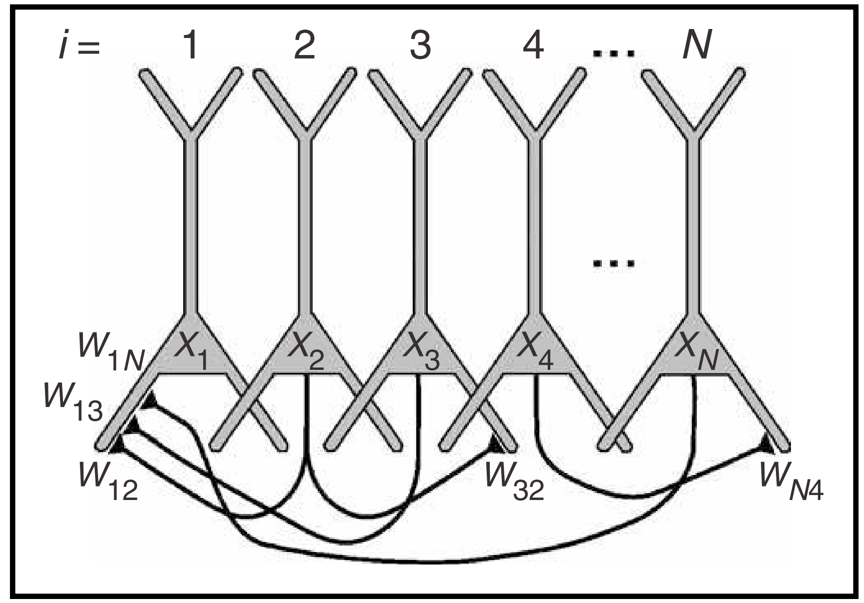




MEMORY RETRIEVAL AS PROBABILISTIC INFERENCE

stored activities

x_1	x_2	x_3	x_4
2	0	5	3
3	8	2	1

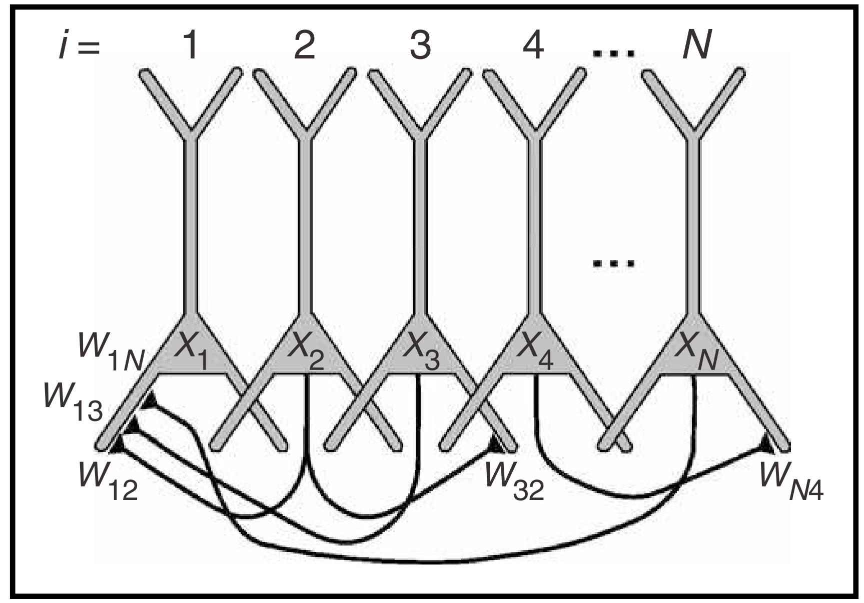


storage in synaptic efficacies

MEMORY RETRIEVAL AS PROBABILISTIC INFERENCE

stored activities

x_1	x_2	x_3	x_4
2	0	5	3
3	8	2	1



storage in synaptic efficacies

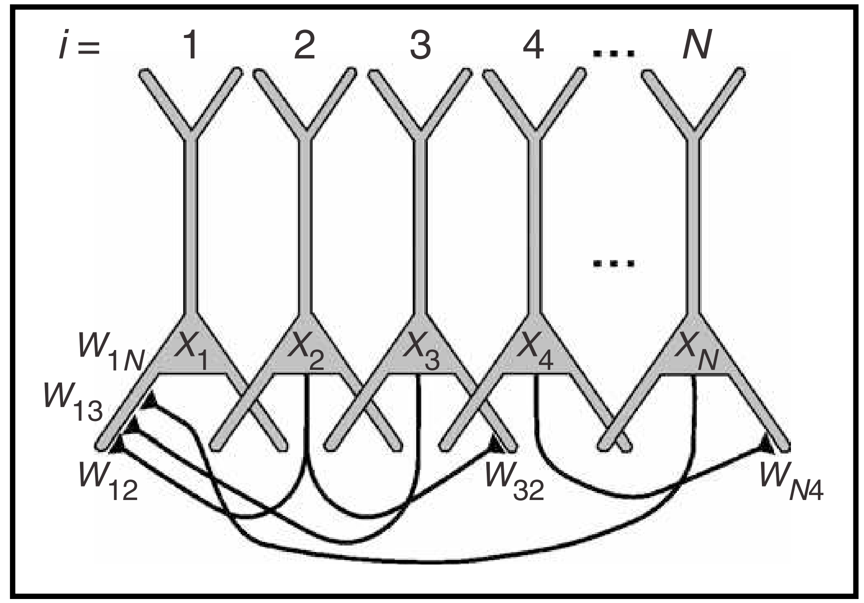
input for retrieval

$\tilde{x}_1$	$\tilde{x}_2$	$\tilde{x}_3$	$\tilde{x}_4$
2	8	2	3

MEMORY RETRIEVAL AS PROBABILISTIC INFERENCE

stored activities

x_1	x_2	x_3	x_4	
2	0	5	3	...
3	8	2	1	...
...	...	...	...	...



storage in synaptic efficacies

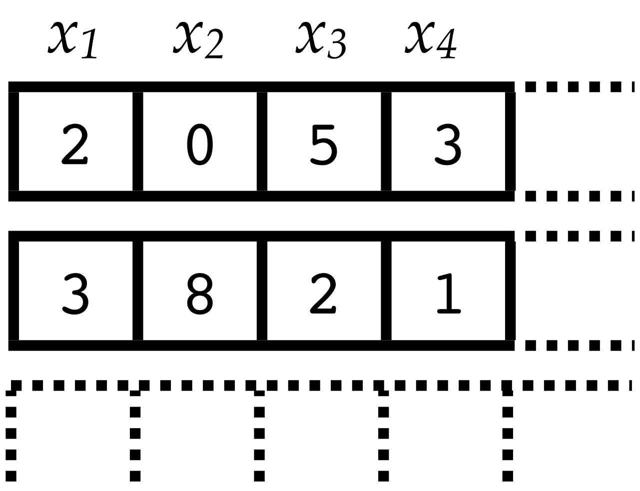
input for retrieval

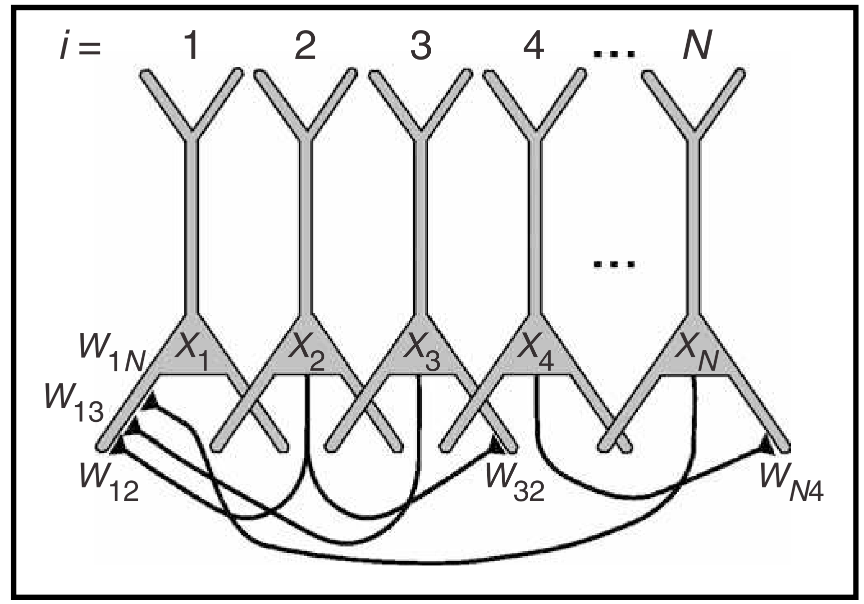
$\tilde{x}_1$	$\tilde{x}_2$	$\tilde{x}_3$	$\tilde{x}_4$	
2	8	2	3	...

❖ input is ambiguous

MEMORY RETRIEVAL AS PROBABILISTIC INFERENCE

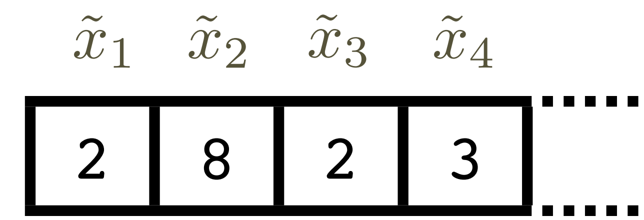
stored activities





storage in synaptic efficacies

input for retrieval



- input is ambiguous
- storage in synapses loses information

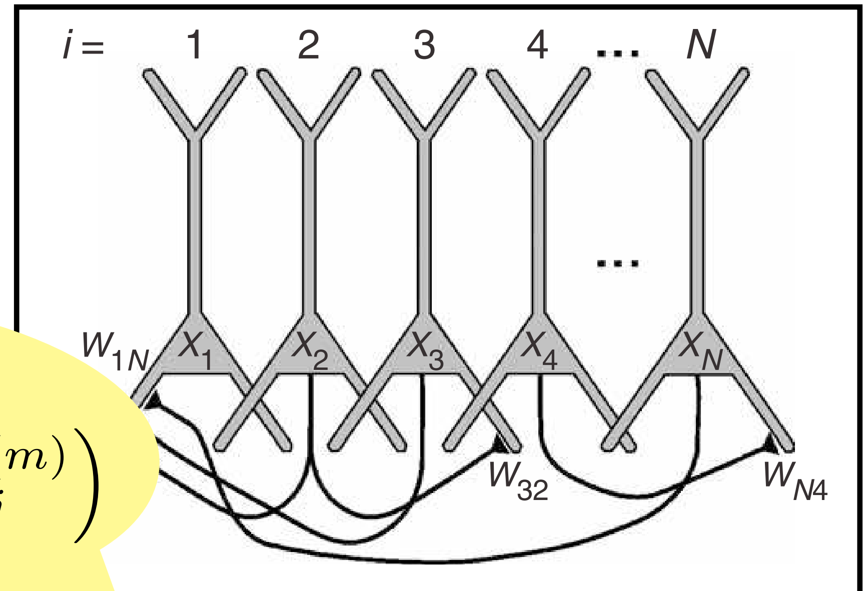
MEMORY RETRIEVAL AS PROBABILISTIC INFERENCE

stored activities

x_1	x_2	x_3	x_4
2	0	5	3
3	8	2	

general form:

$$w_{ij} = \sum_{m=1}^{M} \Omega\left(x_i^{(m)}, x_j^{(m)}\right)$$



storage in synaptic efficacies

w_{13} = 2×5 + 3×2 = 16

input for retrieval

$\tilde{x}_1$	$\tilde{x}_2$	$\tilde{x}_3$	$\tilde{x}_4$
2	8	2	3

- input is ambiguous
- storage in synapses loses information

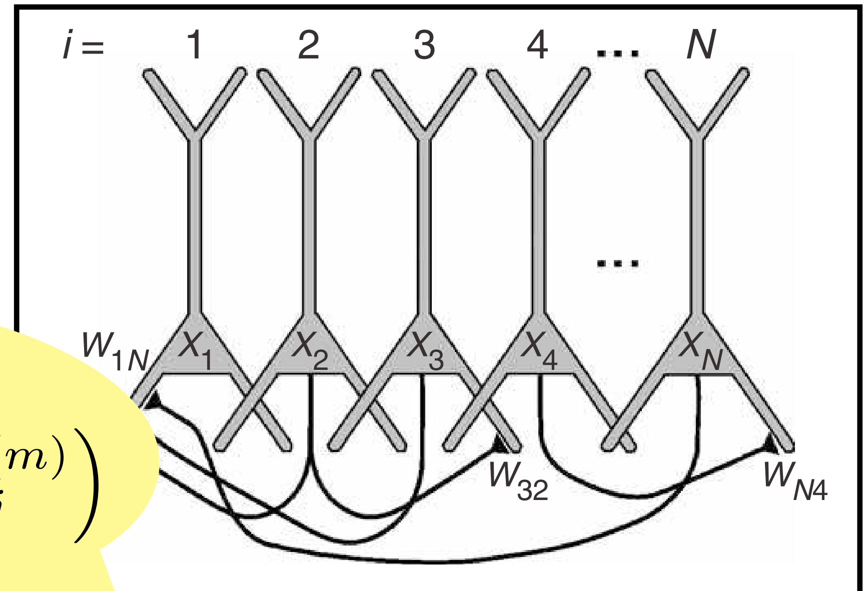
MEMORY RETRIEVAL AS PROBABILISTIC INFERENCE

stored activities

x_1	x_2	x_3	x_4
2	0	5	3
3	8	2	

general form:

$$w_{ij} = \sum_{m=1}^{M} \Omega\left(x_i^{(m)}, x_j^{(m)}\right)$$



storage in synaptic efficacies

$w_{13} = 2{\times}5 + 3{\times}2 = 16$

input for retrieval

$\tilde{x}_1$	$\tilde{x}_2$	$\tilde{x}_3$	$\tilde{x}_4$
2	8	2	3

- input is ambiguous
- storage in synapses loses information
- prior knowledge about kind of patterns stored

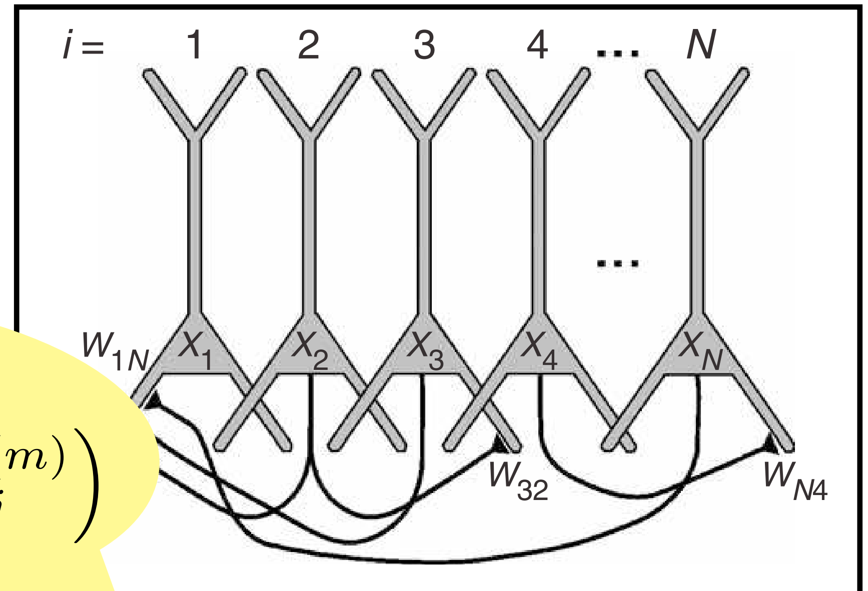
MEMORY RETRIEVAL AS PROBABILISTIC INFERENCE

stored activities

x_1	x_2	x_3	x_4
2	0	5	3
3	8	2	

general form:

$$w_{ij} = \sum_{m=1}^{M} \Omega\left(x_i^{(m)}, x_j^{(m)}\right)$$



storage in synaptic efficacies

w_{13} = 2×5 + 3×2 = 16

input for retrieval

$\tilde{x}_1$	$\tilde{x}_2$	$\tilde{x}_3$	$\tilde{x}_4$
2	8	2	3

- input is ambiguous
- storage in synapses loses information
- prior knowledge about kind of patterns stored

full answer to a memory query is a probability distribution

P(population activity pattern | recall cue, synaptic efficacies)

MacKay 1991, Sommer & Dayan 1998

UNDERSTANDING THE POSTERIOR

P(population activity pattern | recall cue, synaptic efficacies)

UNDERSTANDING THE POSTERIOR

$$P(\mathbf{x}|\tilde{\mathbf{x}}, \mathbf{W})$$

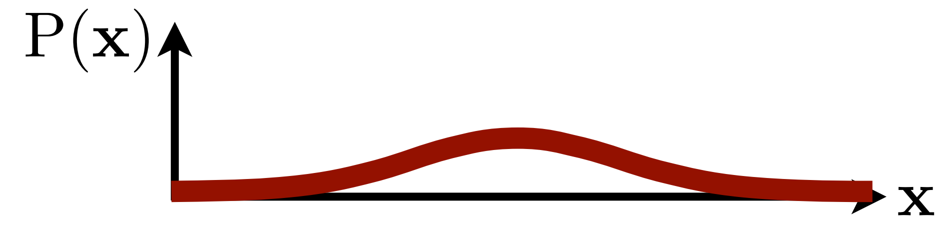
UNDERSTANDING THE POSTERIOR

$$\mathrm{P}(\mathbf{x}|\tilde{\mathbf{x}}, \mathbf{W}) \propto \mathrm{P}(\mathbf{x})\ \mathrm{P}(\tilde{\mathbf{x}}|\mathbf{x})\ \mathrm{P}(\mathbf{W}|\mathbf{x})$$

UNDERSTANDING THE POSTERIOR

$$\mathrm{P}(\mathbf{x}|\tilde{\mathbf{x}}, \mathbf{W}) \propto \mathrm{P}(\mathbf{x}) \; \mathrm{P}(\tilde{\mathbf{x}}|\mathbf{x}) \; \mathrm{P}(\mathbf{W}|\mathbf{x})$$

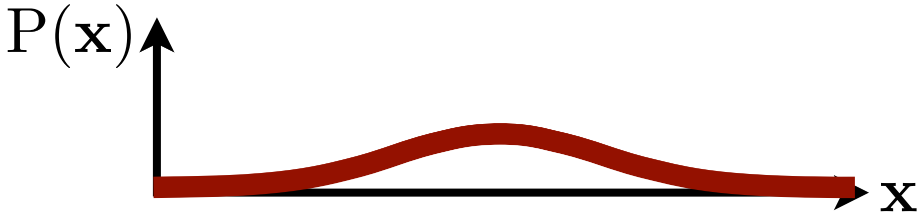
$\mathrm{P}(\mathbf{x})$ the probability that pattern $\mathbf{x}$ is chosen to be stored



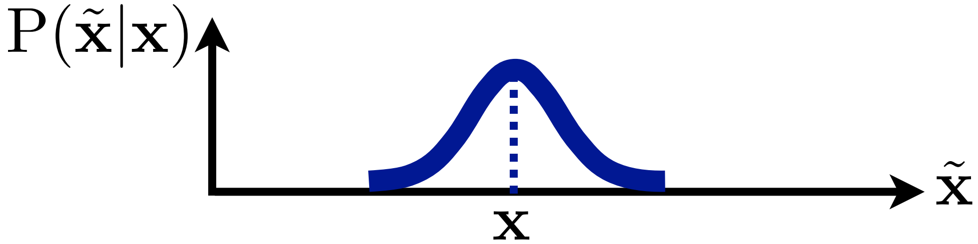
UNDERSTANDING THE POSTERIOR

$$\mathrm{P}(\mathbf{x}|\tilde{\mathbf{x}}, \mathbf{W}) \propto \mathrm{P}(\mathbf{x})\ \mathrm{P}(\tilde{\mathbf{x}}|\mathbf{x})\ \mathrm{P}(\mathbf{W}|\mathbf{x})$$

$\mathrm{P}(\mathbf{x})$ the probability that pattern $\mathbf{x}$ is chosen to be stored



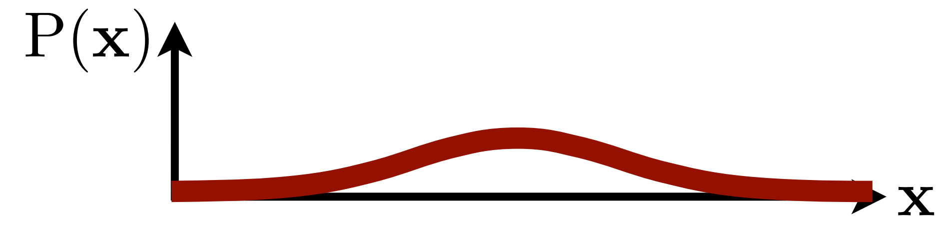
$\mathrm{P}(\tilde{\mathbf{x}}|\mathbf{x})$ the probability that input $\tilde{\mathbf{x}}$ is received when the true pattern to be recalled is $\mathbf{x}$



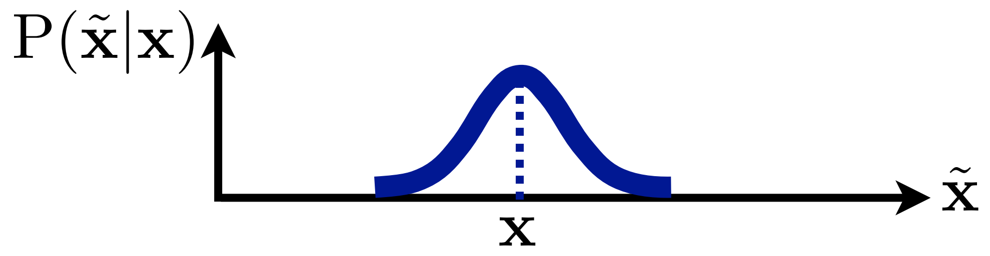
UNDERSTANDING THE POSTERIOR

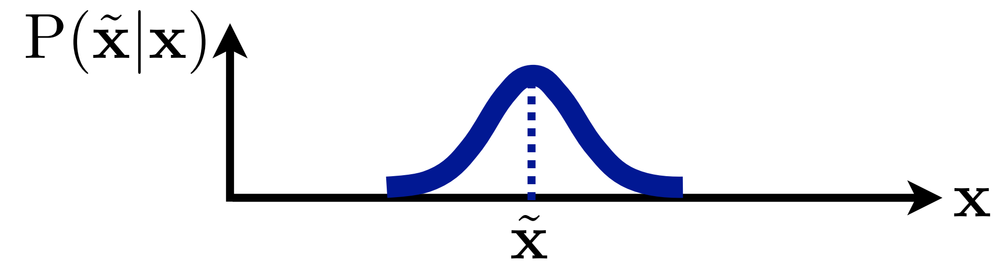
$$P(\mathbf{x}|\tilde{\mathbf{x}}, \mathbf{W}) \propto P(\mathbf{x})\ P(\tilde{\mathbf{x}}|\mathbf{x})\ P(\mathbf{W}|\mathbf{x})$$

$P(\mathbf{x})$ the probability that pattern $\mathbf{x}$ is chosen to be stored



$P(\tilde{\mathbf{x}}|\mathbf{x})$ the probability that input $\tilde{\mathbf{x}}$ is received when the true pattern to be recalled is $\mathbf{x}$

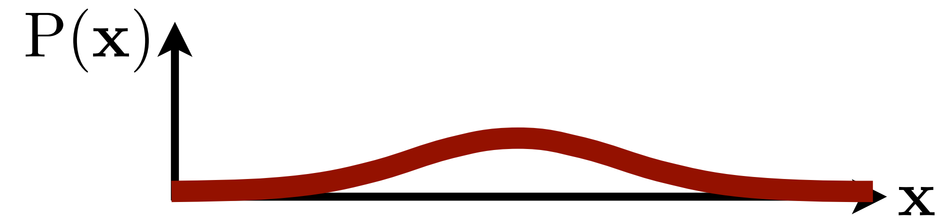




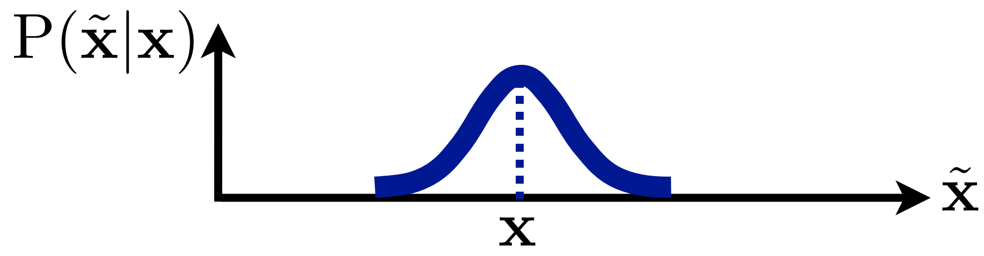
UNDERSTANDING THE POSTERIOR

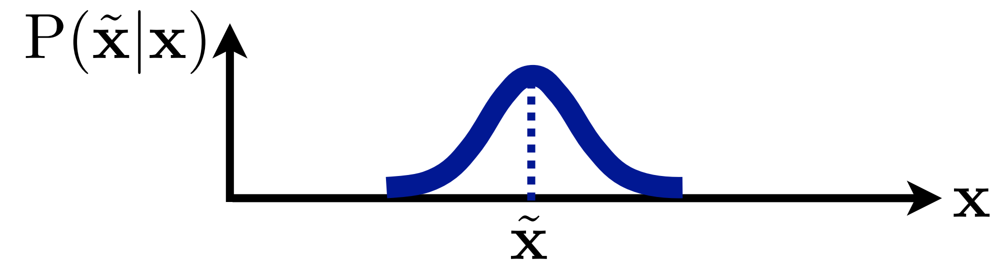
$$\mathrm{P}(\mathbf{x}|\tilde{\mathbf{x}}, \mathbf{W}) \propto \mathrm{P}(\mathbf{x}) \; \mathrm{P}(\tilde{\mathbf{x}}|\mathbf{x}) \; \mathrm{P}(\mathbf{W}|\mathbf{x})$$

$\mathrm{P}(\mathbf{x})$ the probability that pattern $\mathbf{x}$ is chosen to be stored

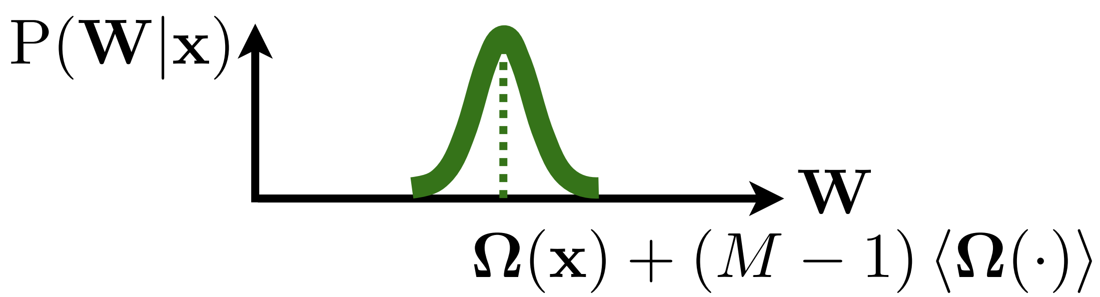


$\mathrm{P}(\tilde{\mathbf{x}}|\mathbf{x})$ the probability that input $\tilde{\mathbf{x}}$ is received when the true pattern to be recalled is $\mathbf{x}$





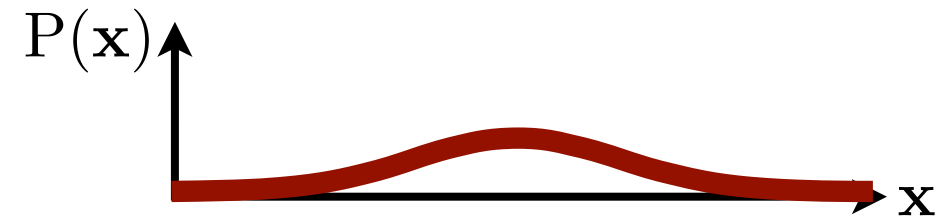
$\mathrm{P}(\mathbf{W}|\mathbf{x})$ the probability of obtaining weight matrix $\mathbf{W}$ by storing pattern $\mathbf{x}$ and M-1 random patterns



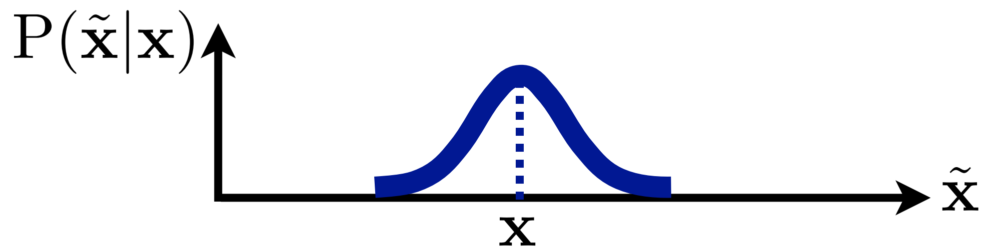
UNDERSTANDING THE POSTERIOR

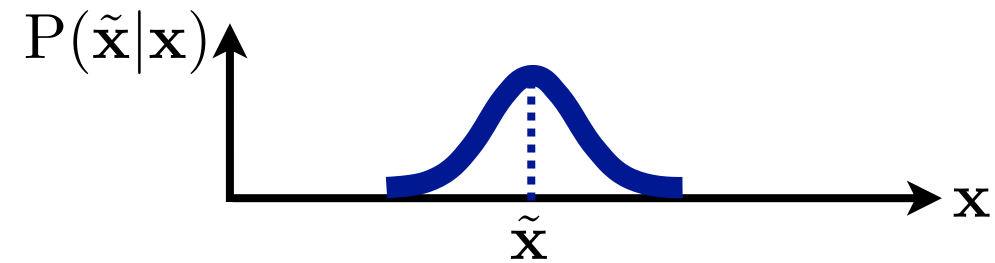
$$P(\mathbf{x}|\tilde{\mathbf{x}}, \mathbf{W}) \propto P(\mathbf{x})\ P(\tilde{\mathbf{x}}|\mathbf{x})\ P(\mathbf{W}|\mathbf{x})$$

$P(\mathbf{x})$ the probability that pattern $\mathbf{x}$ is chosen to be stored

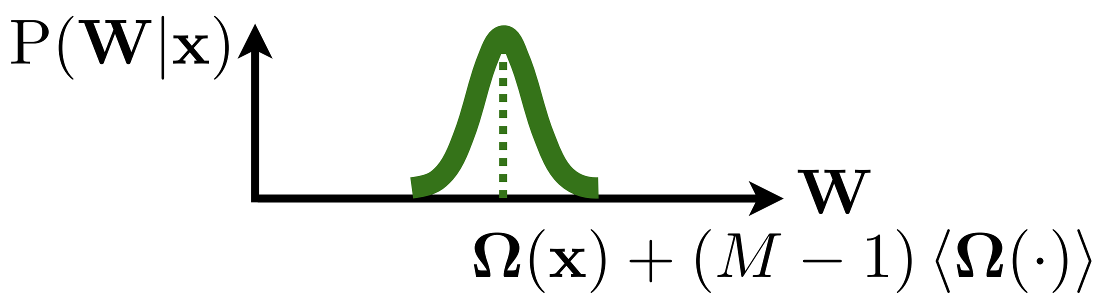


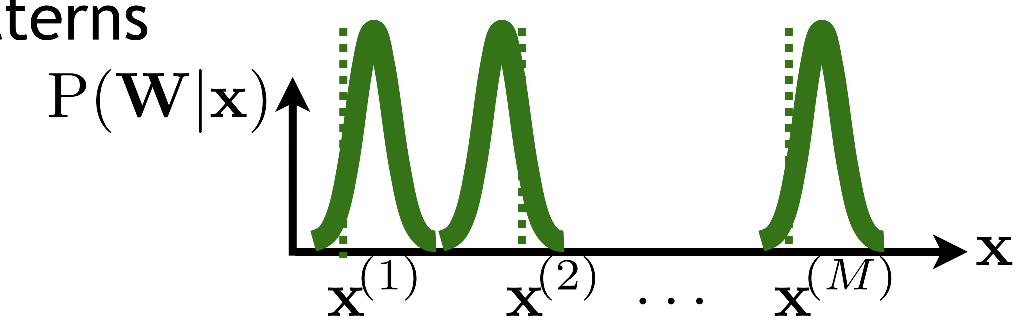
$P(\tilde{\mathbf{x}}|\mathbf{x})$ the probability that input $\tilde{\mathbf{x}}$ is received when the true pattern to be recalled is $\mathbf{x}$





$P(\mathbf{W}|\mathbf{x})$ the probability of obtaining weight matrix $\mathbf{W}$ by storing pattern $\mathbf{x}$ and M-1 random patterns

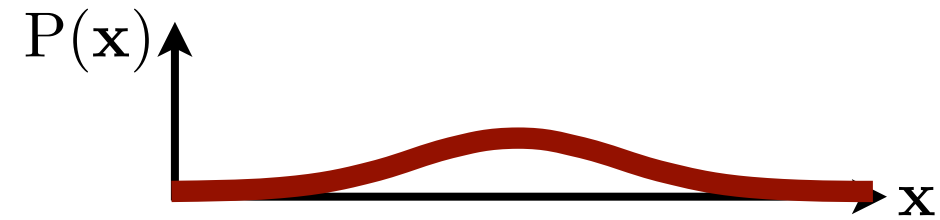




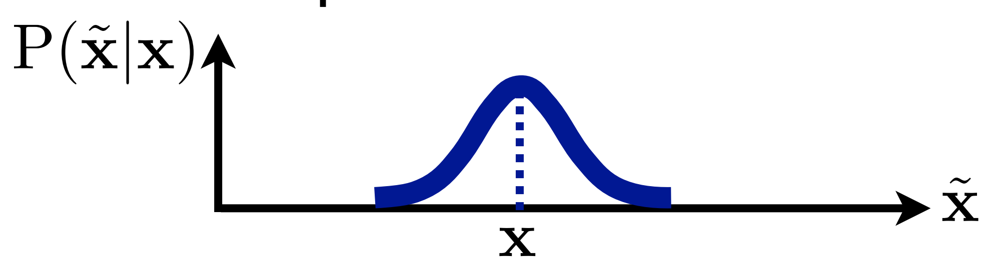
UNDERSTANDING THE POSTERIOR

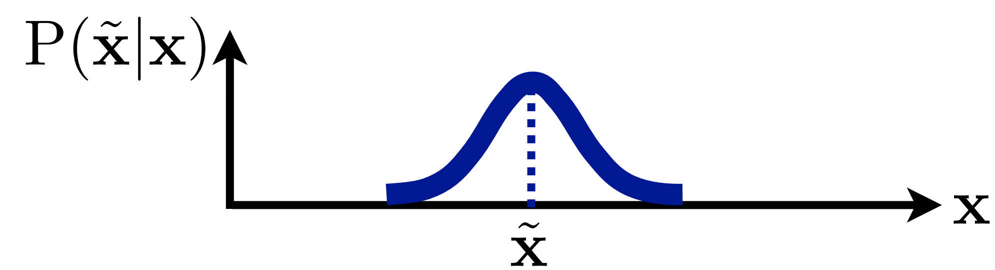
$$P(\mathbf{x}|\tilde{\mathbf{x}}, \mathbf{W}) \propto P(\mathbf{x})\ P(\tilde{\mathbf{x}}|\mathbf{x})\ P(\mathbf{W}|\mathbf{x})$$

$P(\mathbf{x})$ the probability that pattern $\mathbf{x}$ is chosen to be stored

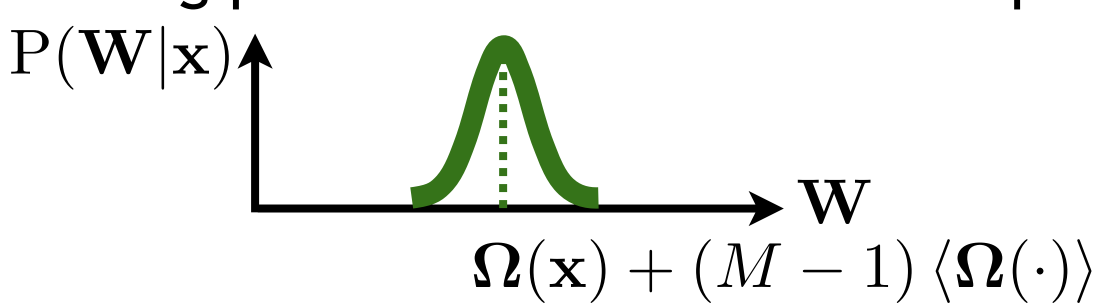


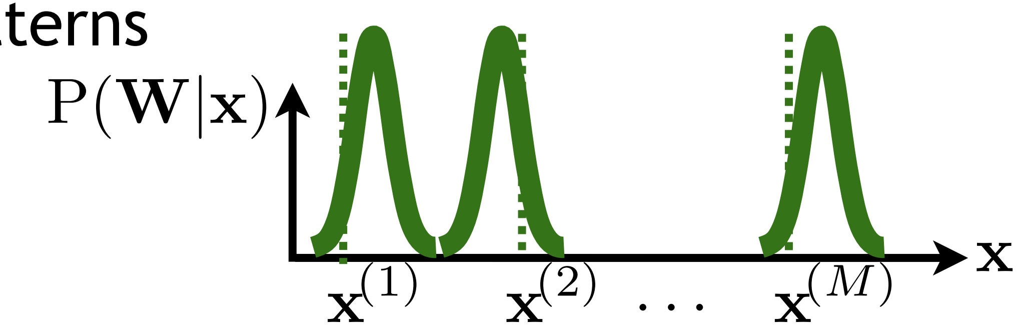
$P(\tilde{\mathbf{x}}|\mathbf{x})$ the probability that input $\tilde{\mathbf{x}}$ is received when the true pattern to be recalled is $\mathbf{x}$



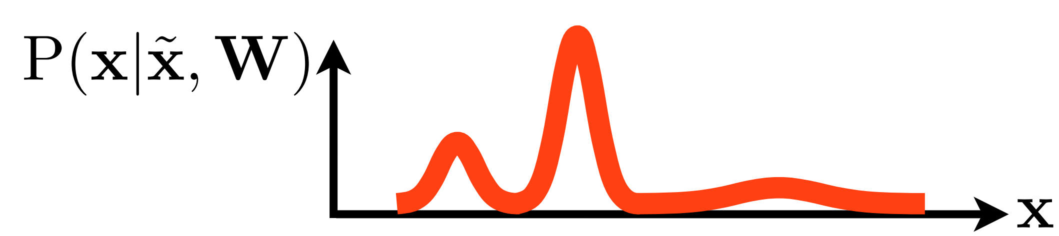


$P(\mathbf{W}|\mathbf{x})$ the probability of obtaining weight matrix $\mathbf{W}$ by storing pattern $\mathbf{x}$ and M-1 random patterns





$P(\mathbf{x}|\tilde{\mathbf{x}}, \mathbf{W})$ what is the probability of that pattern $\mathbf{x}$ is the right pattern to be recalled given that the input is $\mathbf{x}$ and the weight matrix $\mathbf{W}$



MAXIMUM A POSTERIORI INFERENCE

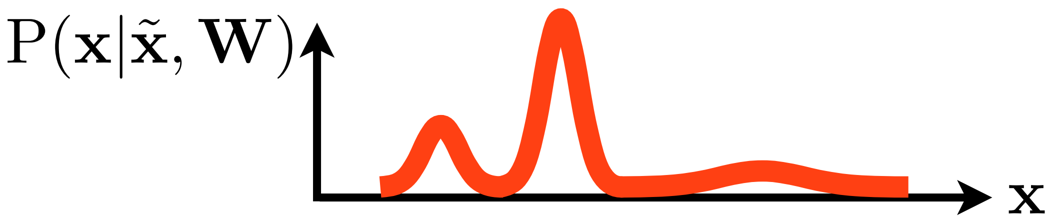
- memories are represented as distributed patterns of activity
- **assume** particular learning rule
- **assume** particular network dynamics

→ **find** energy function

MAXIMUM A POSTERIORI INFERENCE

- memories are represented as distributed patterns of activity
- **assume** particular learning rule
- **define** energy function

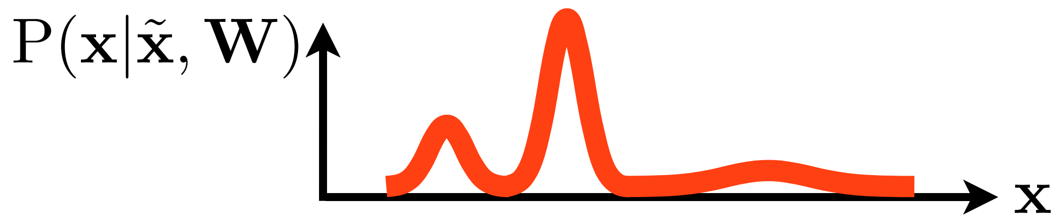
→ **derive** network dynamics



approximate **MAXIMUM A POSTERIORI INFERENCE**

- memories are represented as distributed patterns of activity
- **assume** particular learning rule
- **define** energy function

→ **derive** network dynamics



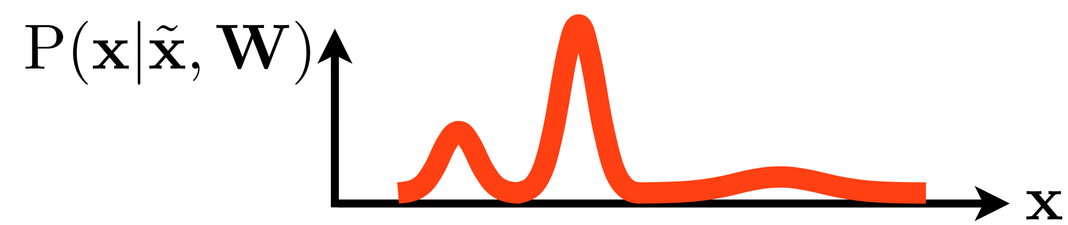
network dynamics implements gradient ascent on the (log) posterior

$$\frac{d}{dt}x_i \propto \frac{\partial}{\partial x_i} \log P(\mathbf{x}|\tilde{\mathbf{x}}, \mathbf{W})$$

approximate **MAXIMUM A POSTERIORI INFERENCE**

- memories are represented as distributed patterns of activity
- **assume** particular learning rule
- **define** energy function

→ **derive** network dynamics



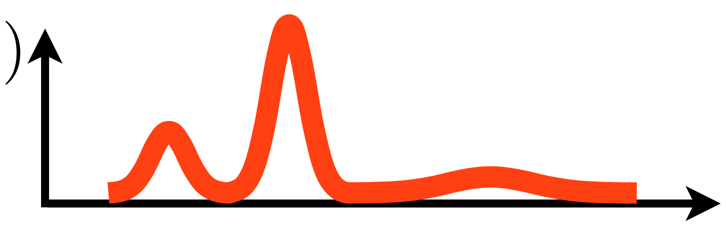
network dynamics implements gradient ascent on the (log) posterior

$$\frac{d}{dt}x_i \propto \frac{\partial}{\partial x_i}\log P(\mathbf{x}) + \frac{\partial}{\partial x_i}\log P(\tilde{\mathbf{x}}|\mathbf{x}) + \frac{\partial}{\partial x_i}\log P(\mathbf{W}|\mathbf{x})$$

approximate **MAXIMUM A POSTERIORI INFERENCE**

- memories are represented as distributed patterns of activity
- **assume** particular learning rule
- **define** energy function

→ **derive** network dynamics



network dynamics implements gradient ascent on the (log) posterior

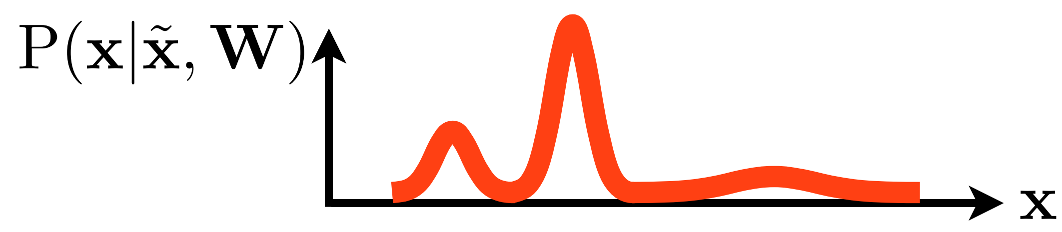
$$\frac{d}{dt}x_i \propto \frac{\partial}{\partial x_i}\log \mathrm{P}(\mathbf{x}) + \frac{\partial}{\partial x_i}\log \mathrm{P}(\tilde{\mathbf{x}}|\mathbf{x}) + \frac{\partial}{\partial x_i}\log \mathrm{P}(\mathbf{W}|\mathbf{x})$$

$$\approx \frac{\partial}{\partial x_i}\log \mathrm{P}(x_i) + \frac{\partial}{\partial x_i}\log \mathrm{P}(\tilde{x}_i|x_i) + \frac{1}{2}\sum_j \frac{\partial}{\partial x_i}\log \mathrm{P}(w_{ij}|x_i, x_j)$$

approximate **MAXIMUM A POSTERIORI INFERENCE**

- memories are represented as distributed patterns of activity
- **assume** particular learning rule
- **define** energy function

→ **derive** network dynamics



network dynamics implements gradient ascent on the (log) posterior

$$\frac{d}{dt}x_i \propto \frac{\partial}{\partial x_i}\log \mathrm{P}(\mathbf{x}) + \frac{\partial}{\partial x_i}\log \mathrm{P}(\tilde{\mathbf{x}}|\mathbf{x}) + \frac{\partial}{\partial x_i}\log \mathrm{P}(\mathbf{W}|\mathbf{x})$$

$$\approx \frac{\partial}{\partial x_i}\log \mathrm{P}(x_i) + \frac{\partial}{\partial x_i}\log \mathrm{P}(\tilde{x}_i|x_i) + \frac{1}{2}\sum_j \frac{\partial}{\partial x_i}\log \mathrm{P}(w_{ij}|x_i, x_j)$$

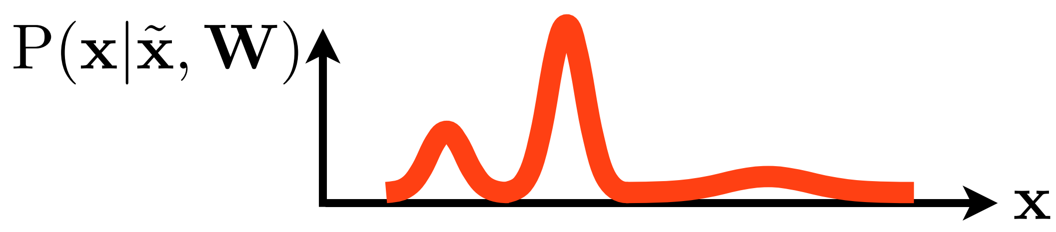
$$\frac{d}{dt}x_i \propto \ldots + \sum_j w_{ij}\,\frac{\partial}{\partial x_i}\Omega(x_i, x_j)$$

Lengyel et al, Nat Neurosci 2005

approximate **MAXIMUM A POSTERIORI INFERENCE**

- memories are represented as distributed patterns of activity
- **assume** particular learning rule
- **define** energy function

→ **derive** network dynamics



network dynamics implements gradient ascent on the (log) posterior

$$\frac{d}{dt}x_i \propto \frac{\partial}{\partial x_i}\log P(\mathbf{x}) + \frac{\partial}{\partial x_i}\log P(\tilde{\mathbf{x}}|\mathbf{x}) + \frac{\partial}{\partial x_i}\log P(\mathbf{W}|\mathbf{x})$$

$$\approx \frac{\partial}{\partial x_i}\log P(x_i) + \frac{\partial}{\partial x_i}\log P(\tilde{x}_i|x_i) + \frac{1}{2}\sum_j \frac{\partial}{\partial x_i}\log P(w_{ij}|x_i, x_j)$$

$$\frac{d}{dt}x_i \propto \ldots + \sum_j w_{ij}\frac{\partial}{\partial x_i}\Omega(x_i, x_j)$$

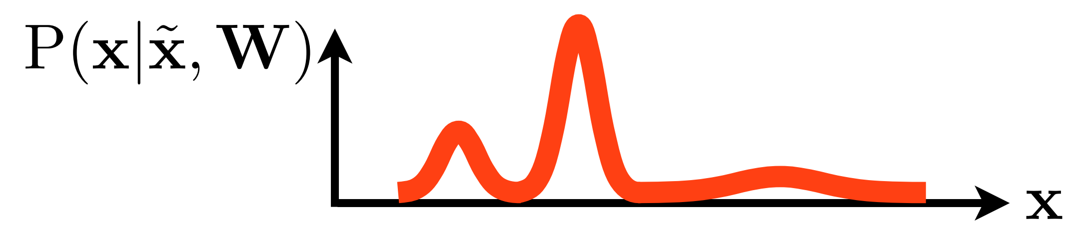
interactions should be scaled by synaptic weights

Lengyel et al, Nat Neurosci 2005

approximate **MAXIMUM A POSTERIORI INFERENCE**

- memories are represented as distributed patterns of activity
- **assume** particular learning rule
- **define** energy function

→ **derive** network dynamics



network dynamics implements gradient ascent on the (log) posterior

$$\frac{d}{dt}x_i \propto \frac{\partial}{\partial x_i}\log \mathrm{P}(\mathbf{x}) + \frac{\partial}{\partial x_i}\log \mathrm{P}(\tilde{\mathbf{x}}|\mathbf{x}) + \frac{\partial}{\partial x_i}\log \mathrm{P}(\mathbf{W}|\mathbf{x})$$

$$\approx \frac{\partial}{\partial x_i}\log \mathrm{P}(x_i) + \frac{\partial}{\partial x_i}\log \mathrm{P}(\tilde{x}_i|x_i) + \frac{1}{2}\sum_j \frac{\partial}{\partial x_i}\log \mathrm{P}(w_{ij}|x_i, x_j)$$

$$\frac{d}{dt}x_i \propto \ldots + \sum_j w_{ij}\frac{\partial}{\partial x_i}\Omega(x_i, x_j)$$

interactions should be scaled by synaptic weights

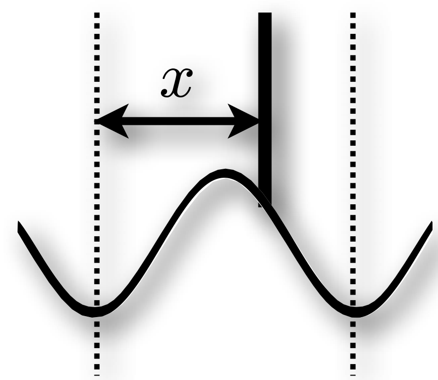
matching between storage and recall

Lengyel et al, Nat Neurosci 2005

PREDICTING INTERACTIONS BETWEEN NEURONS

representation

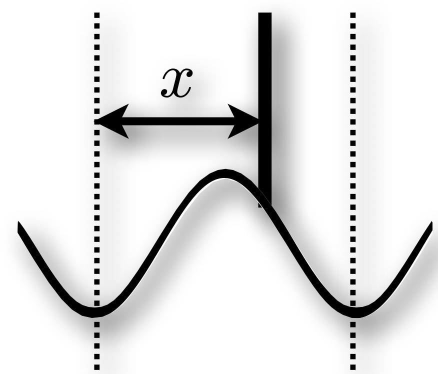
spike timing



PREDICTING INTERACTIONS BETWEEN NEURONS

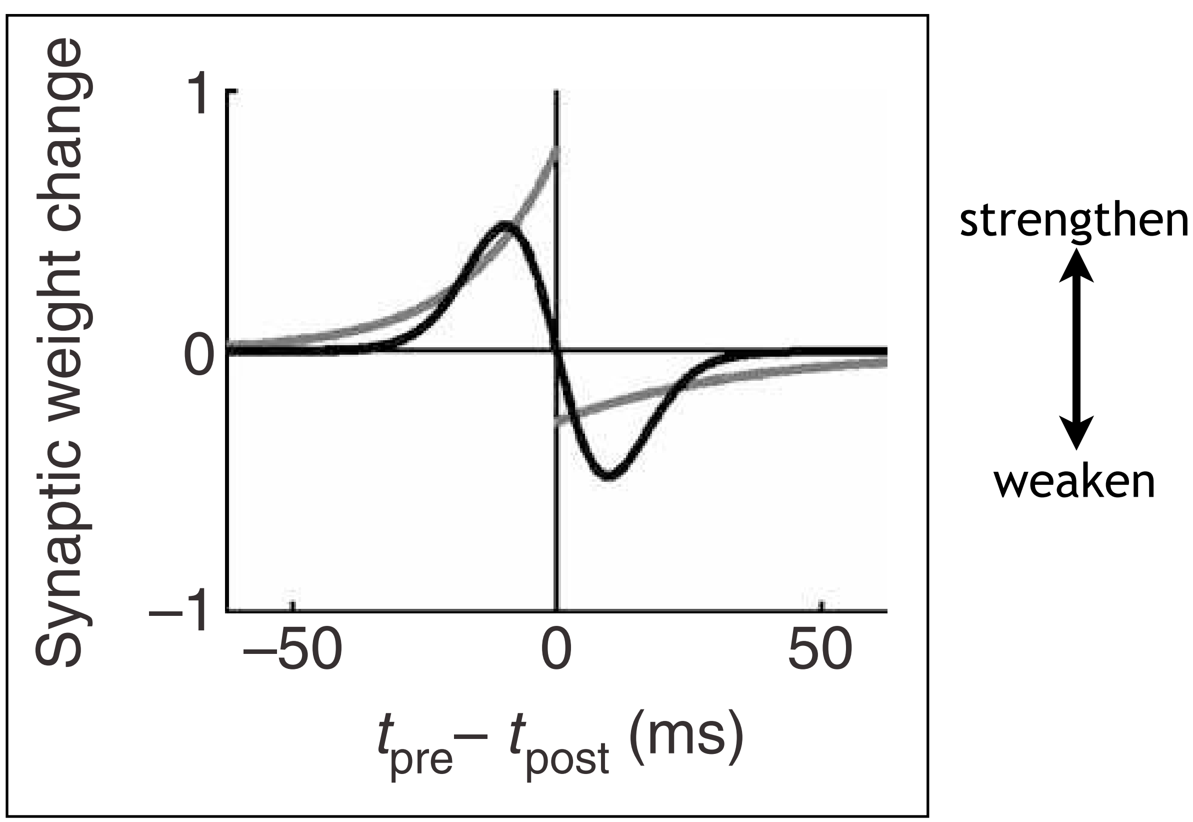
representation

spike timing



storage

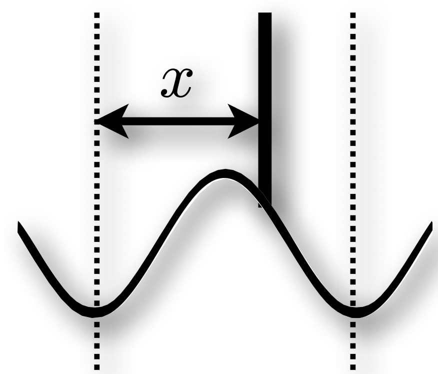
spike timing-dependent plasticity



PREDICTING INTERACTIONS BETWEEN NEURONS

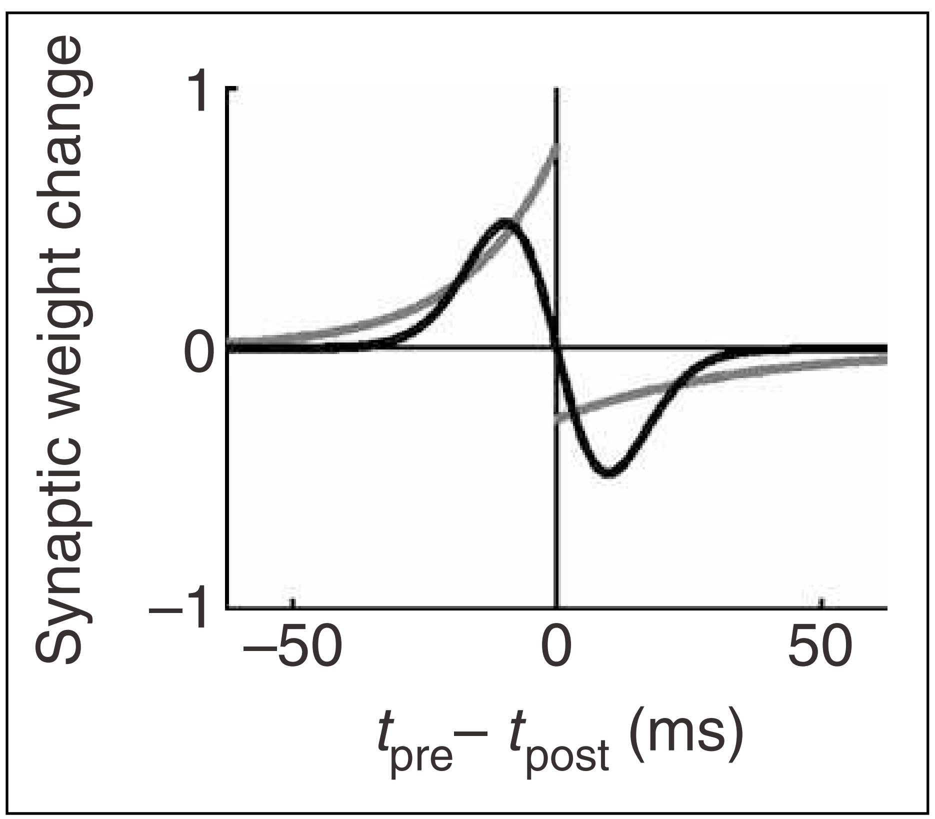
representation

spike timing



storage

spike timing-dependent plasticity

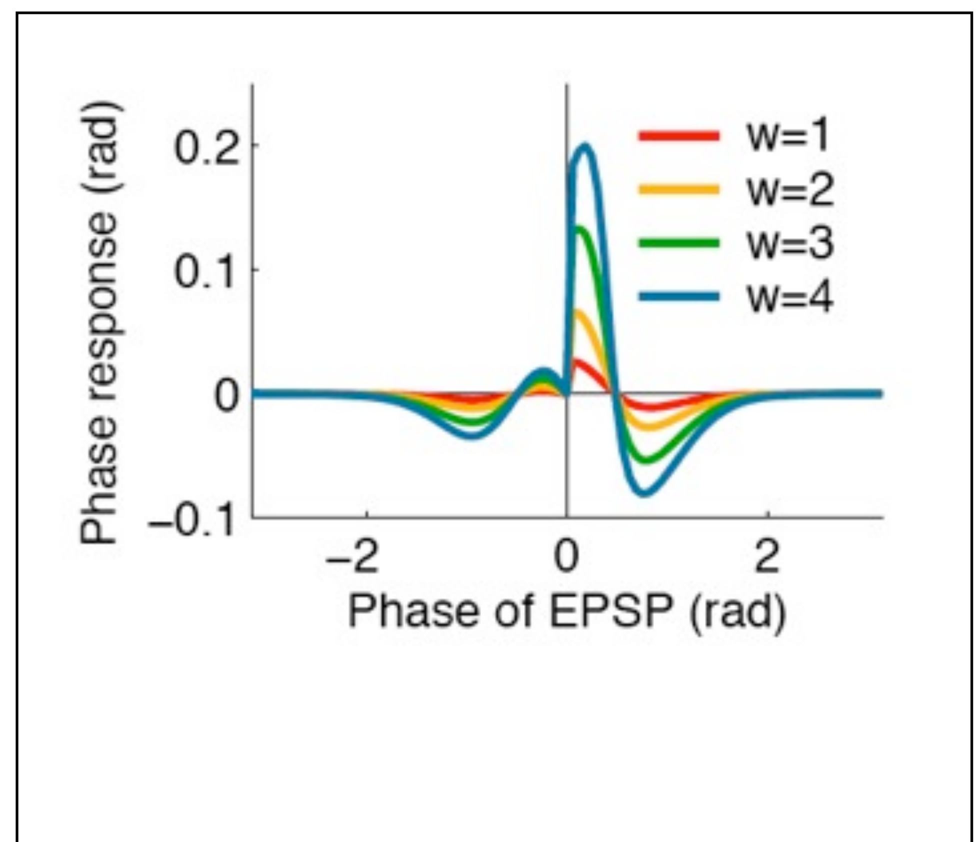


strengthen

weaken

recall

"phase response curve"



delay

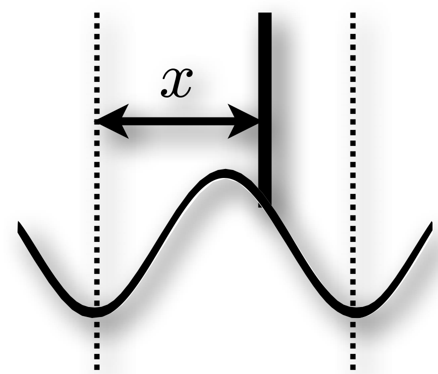
advance

Lengyel et al, Nat Neurosci 2005

PREDICTING INTERACTIONS BETWEEN NEURONS

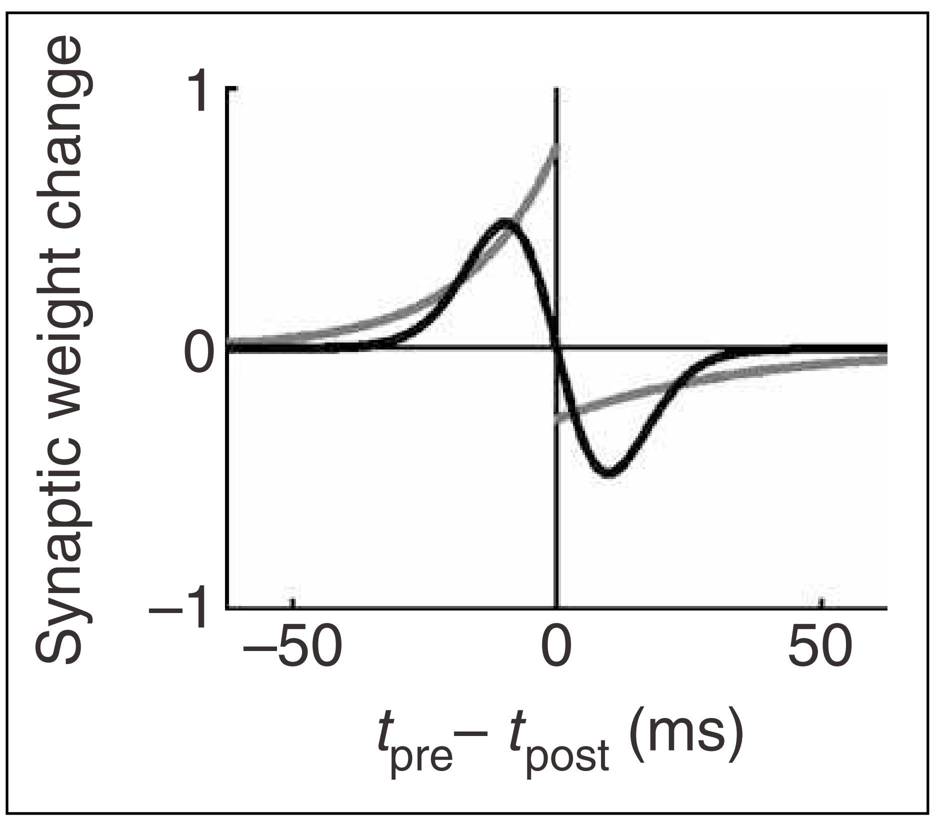
representation

spike timing



storage

spike timing-dependent plasticity

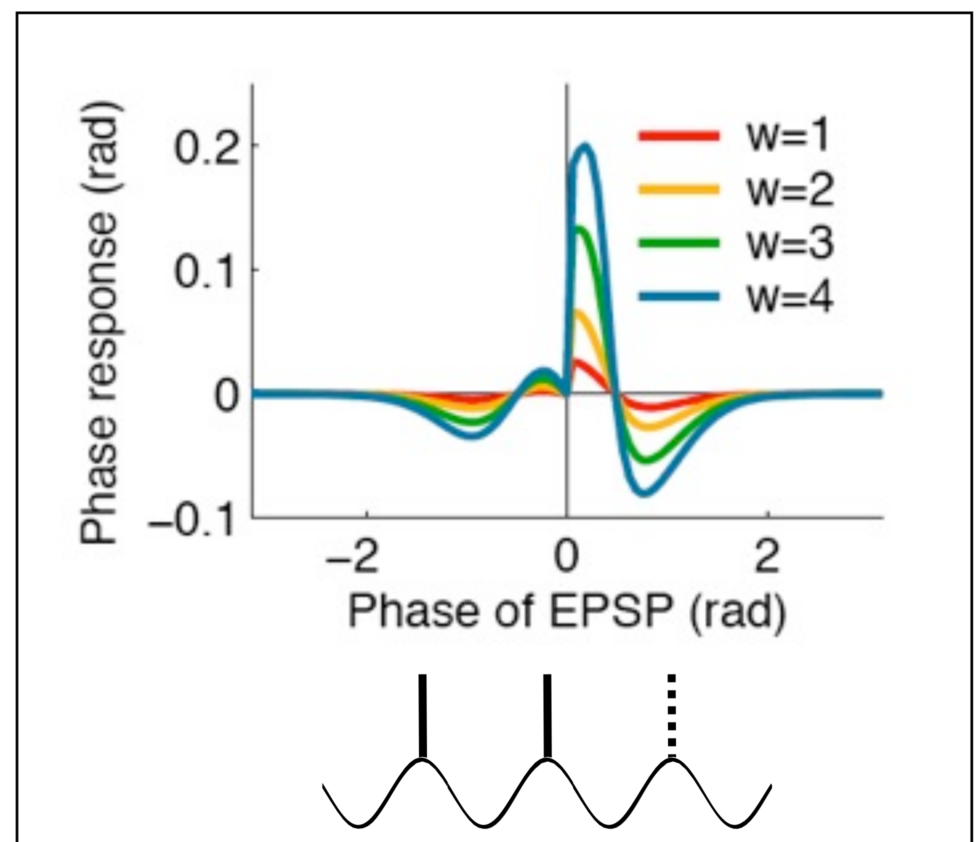


strengthen

weaken

recall

"phase response curve"



delay

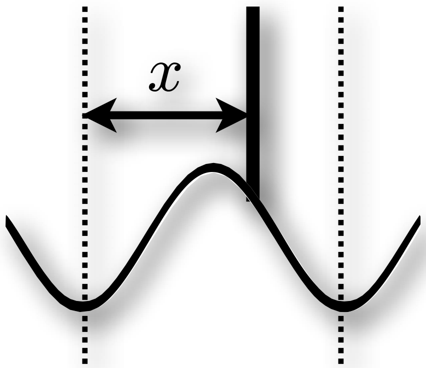
advance

Lengyel et al, Nat Neurosci 2005

PREDICTING INTERACTIONS BETWEEN NEURONS

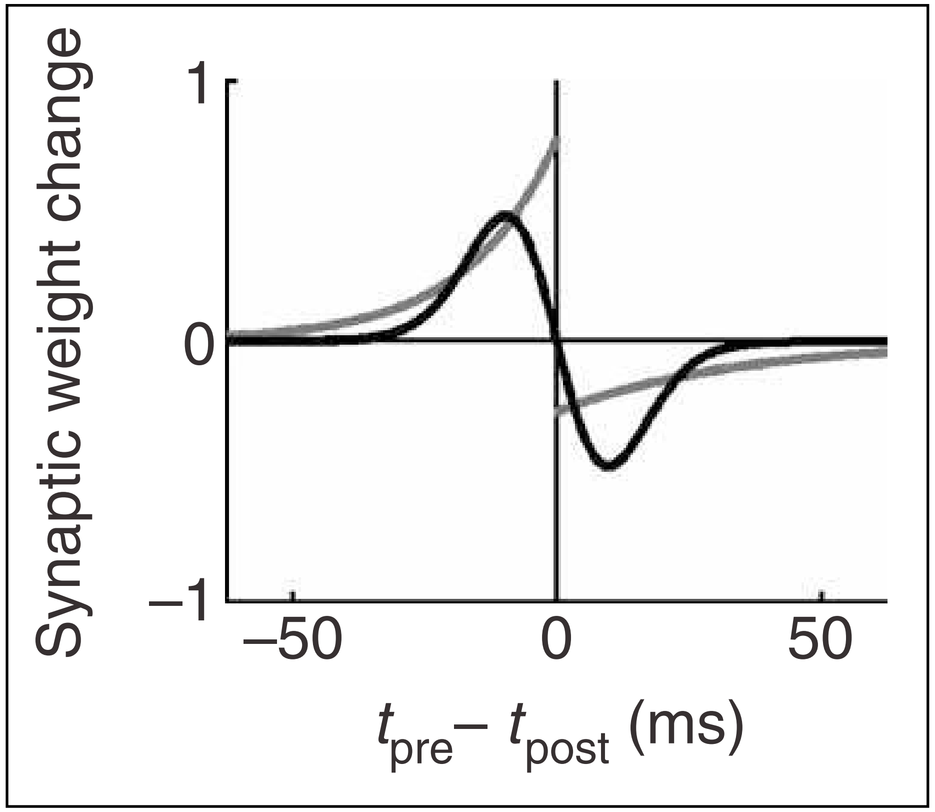
representation

spike timing



storage

spike timing-dependent plasticity

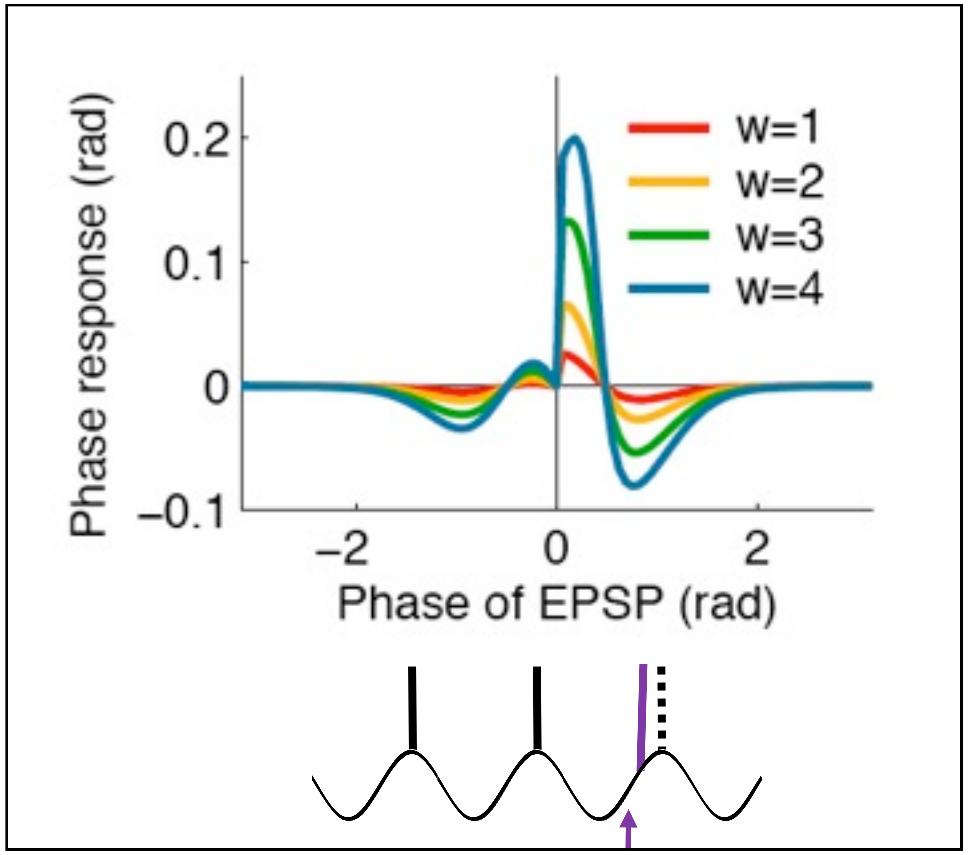


strengthen

weaken

recall

"phase response curve"



delay

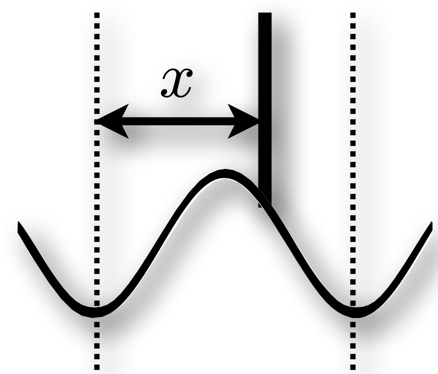
advance

Lengyel et al, Nat Neurosci 2005

PREDICTING INTERACTIONS BETWEEN NEURONS

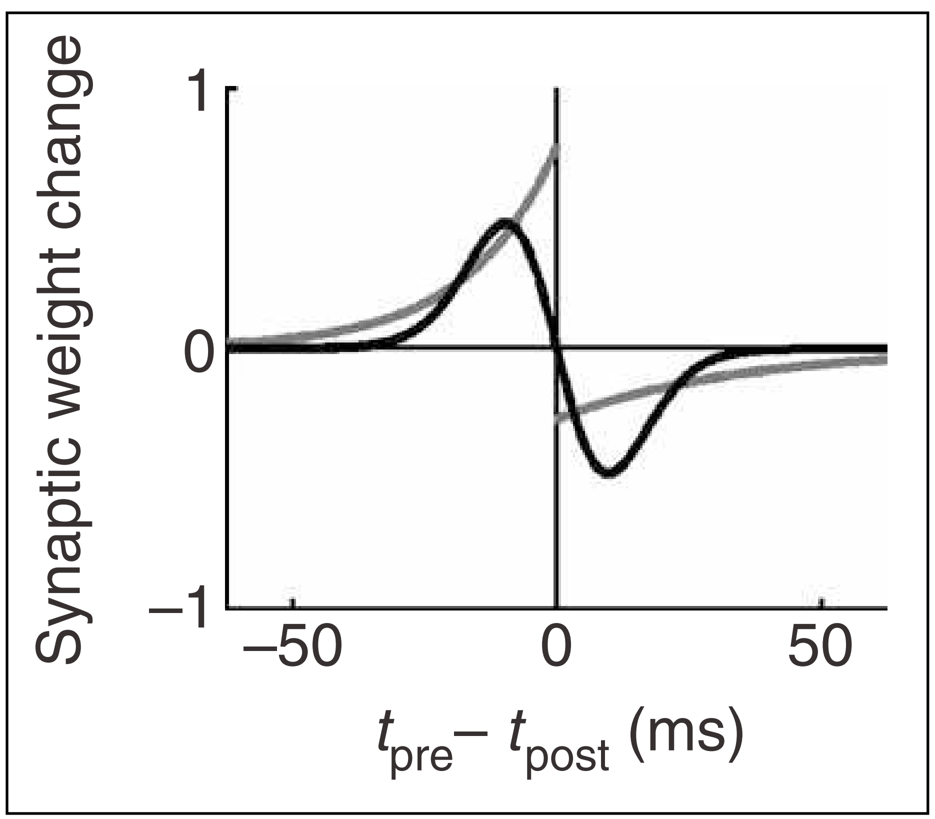
representation

spike timing



storage

spike timing-dependent plasticity

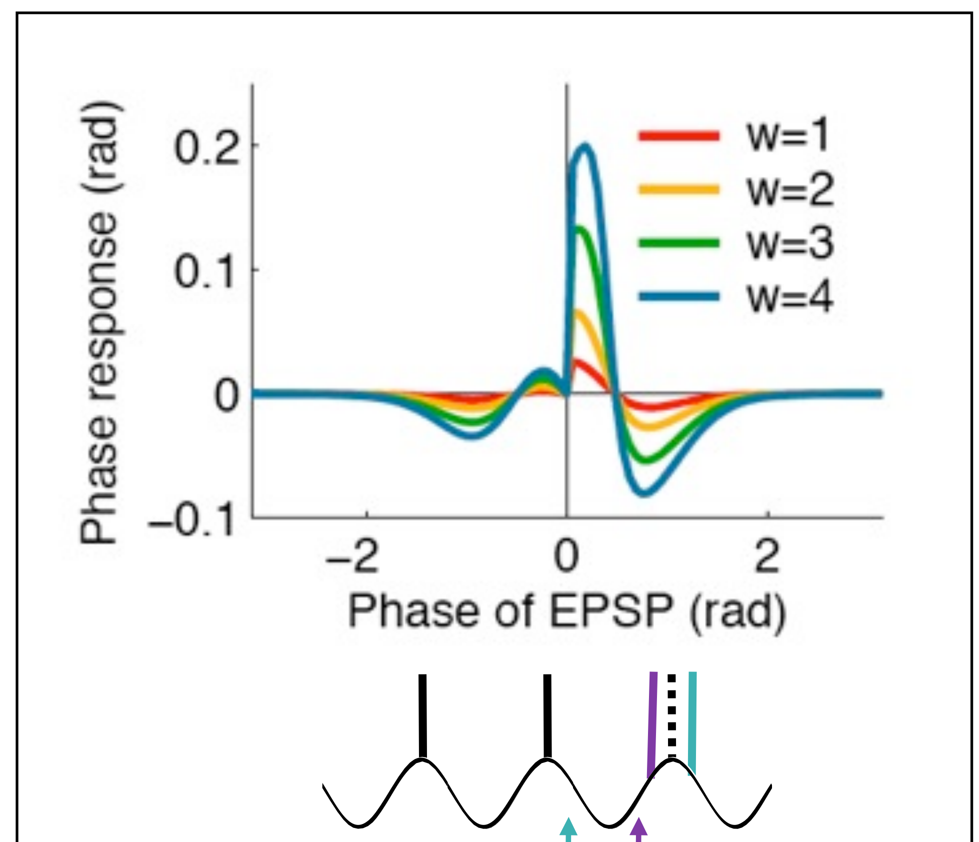


strengthen

weaken

recall

"phase response curve"

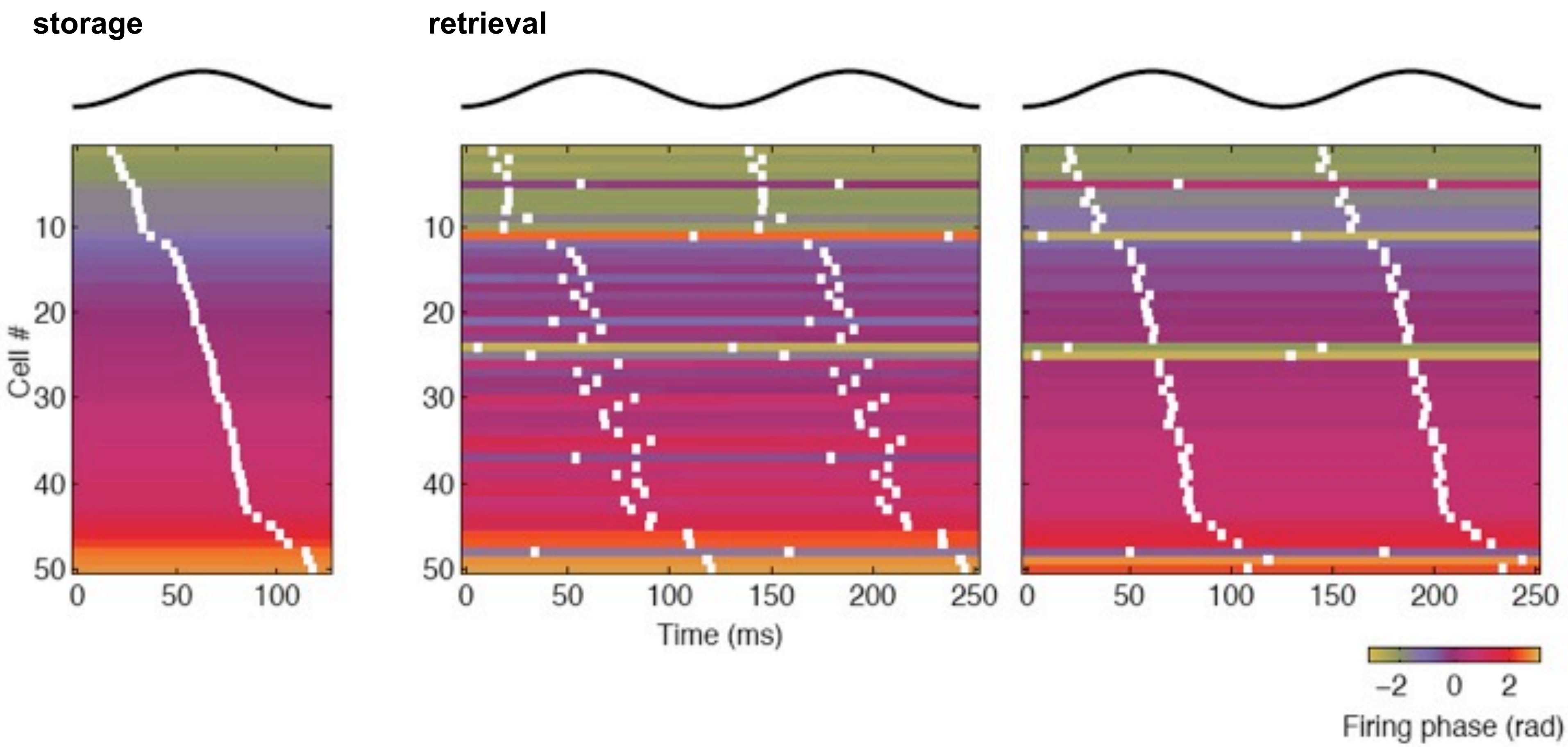


delay

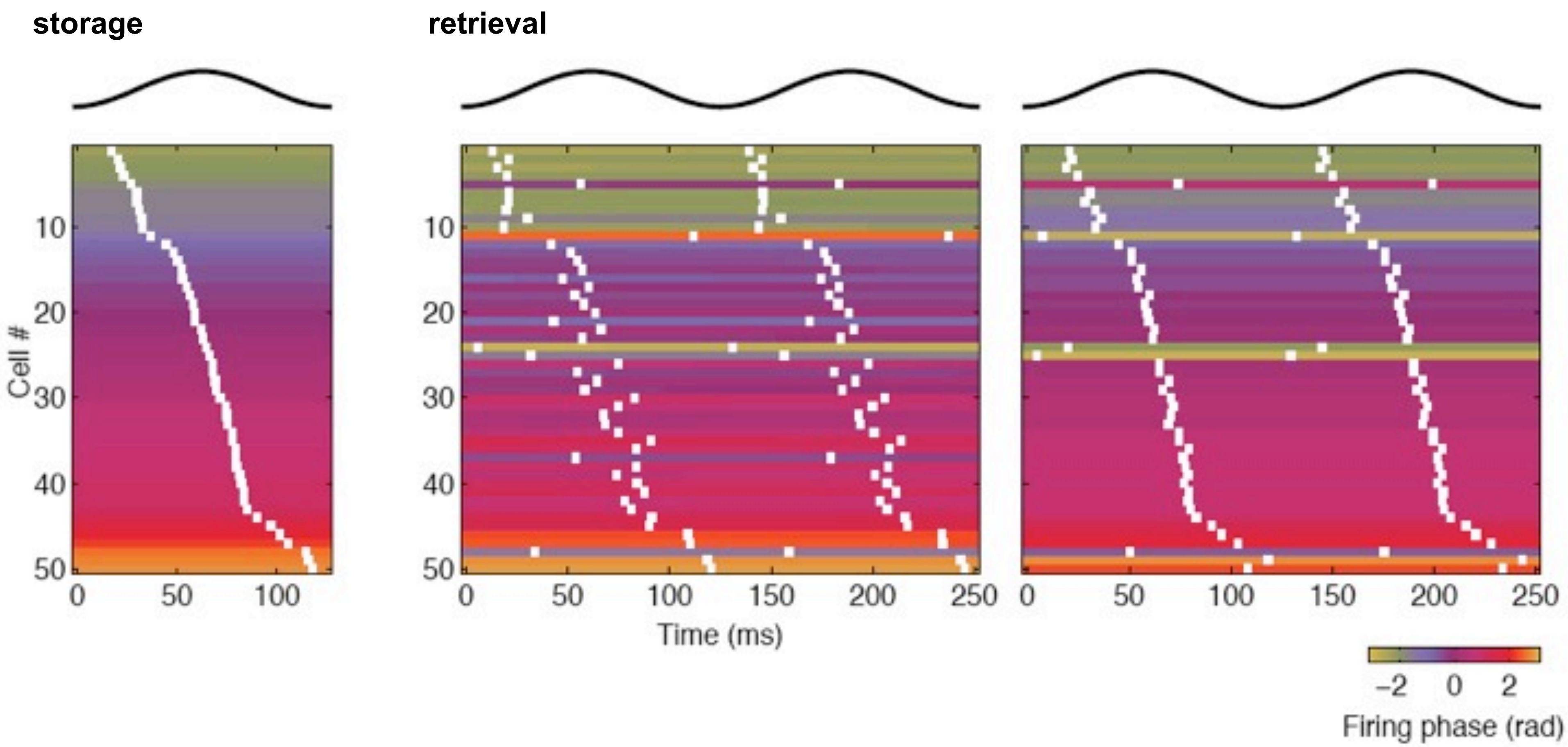
advance

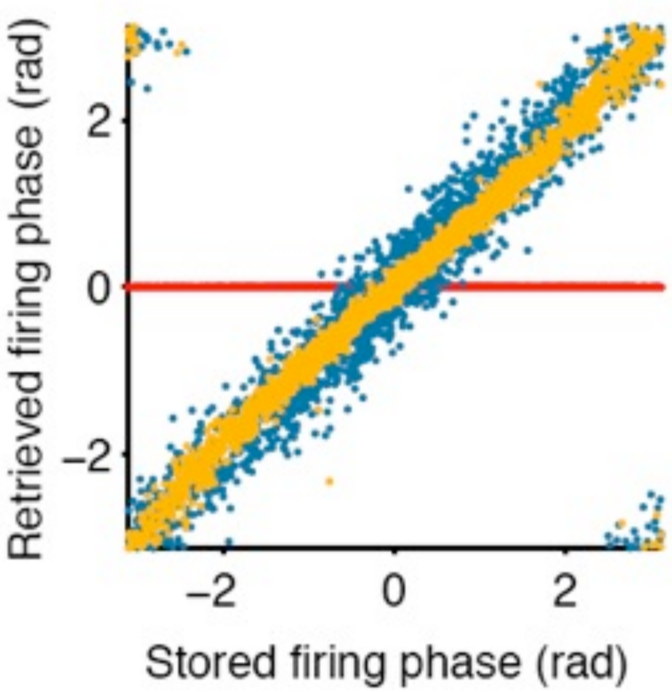
Lengyel et al, Nat Neurosci 2005

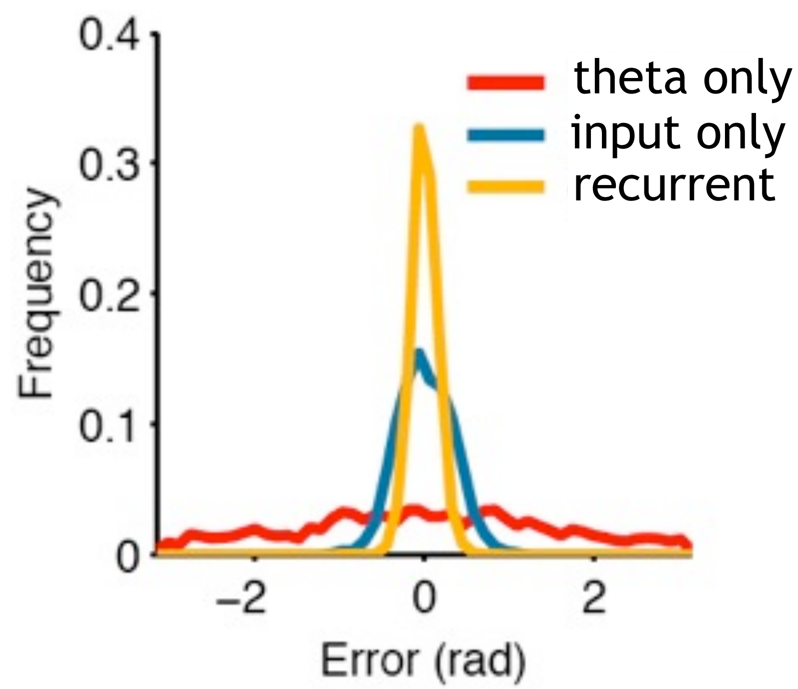
PERFORMANCE OF NETWORK

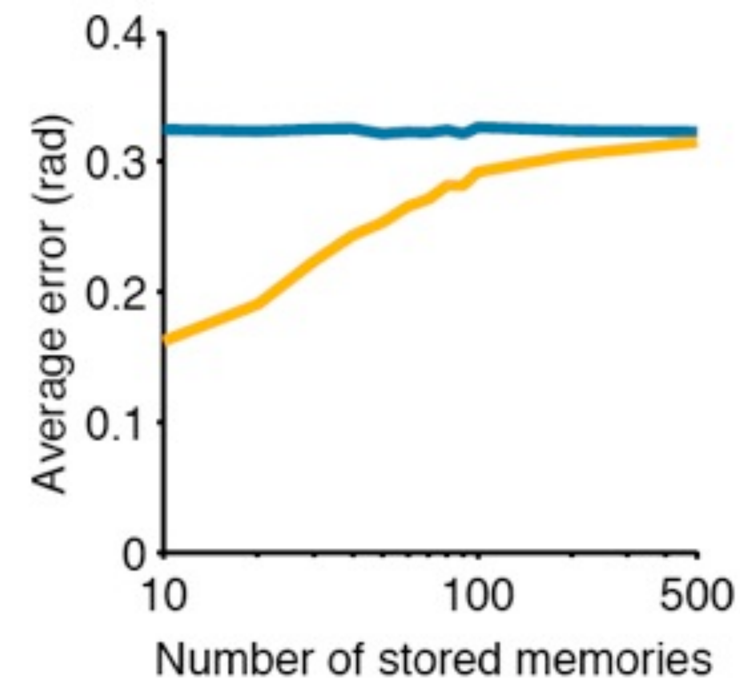


PERFORMANCE OF NETWORK





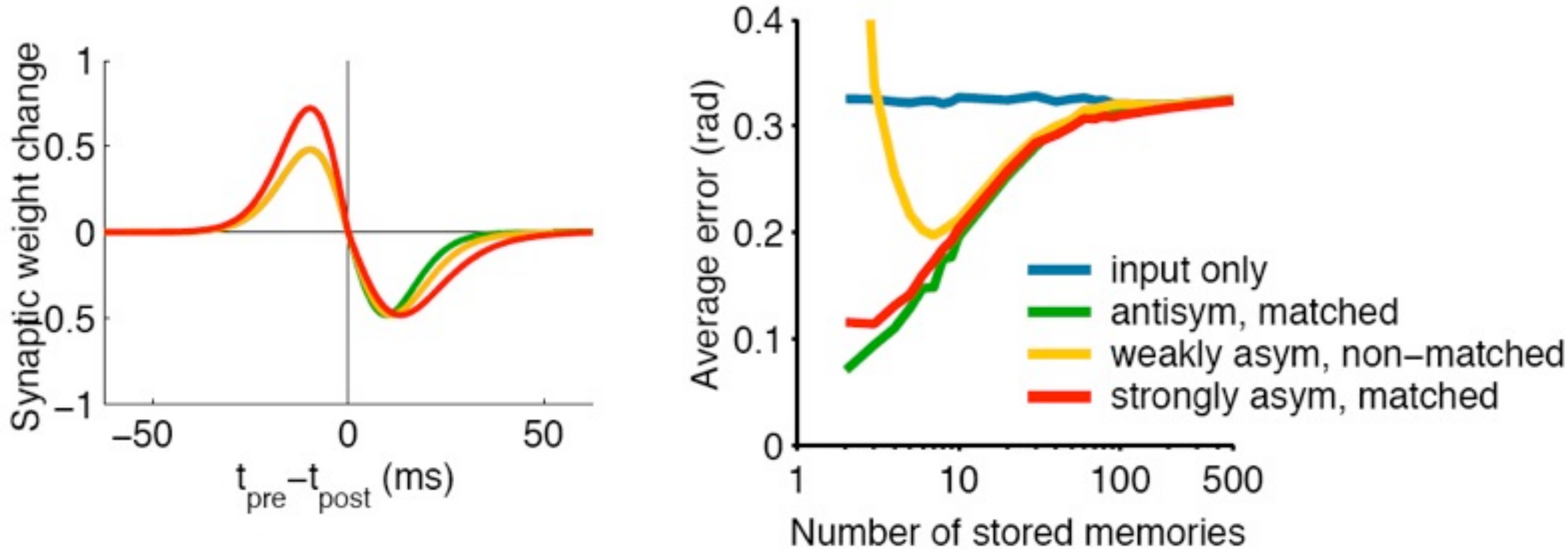




Lengyel et al, Nat Neurosci 2005

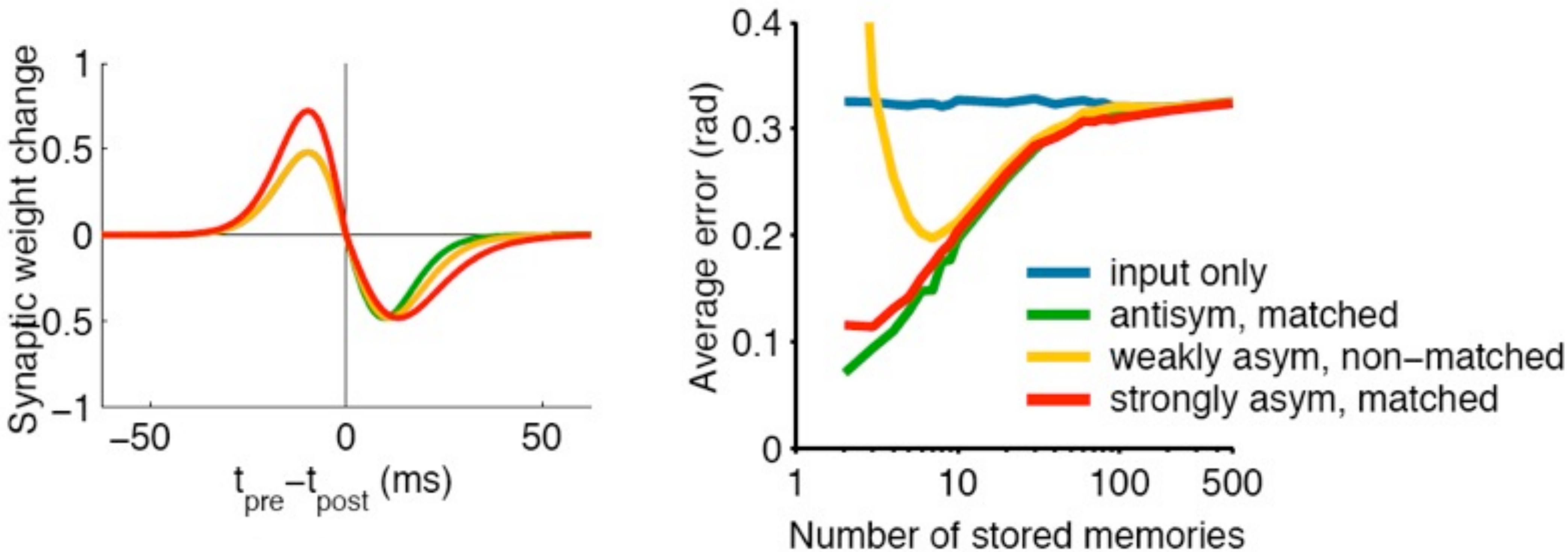
ROBUSTNESS OF RECALL PERFORMANCE

asymmetric STDP

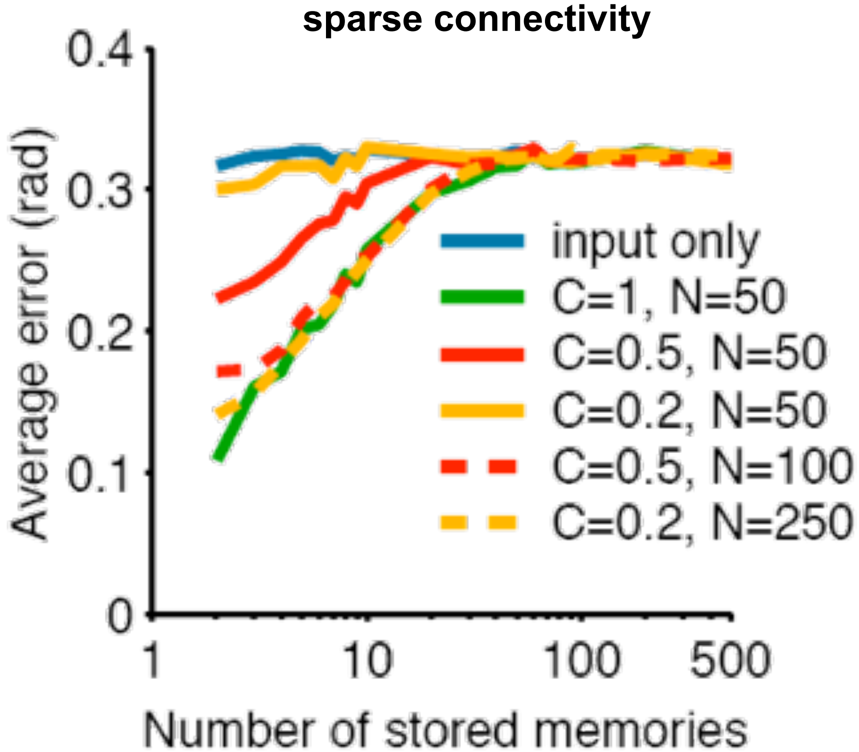


ROBUSTNESS OF RECALL PERFORMANCE

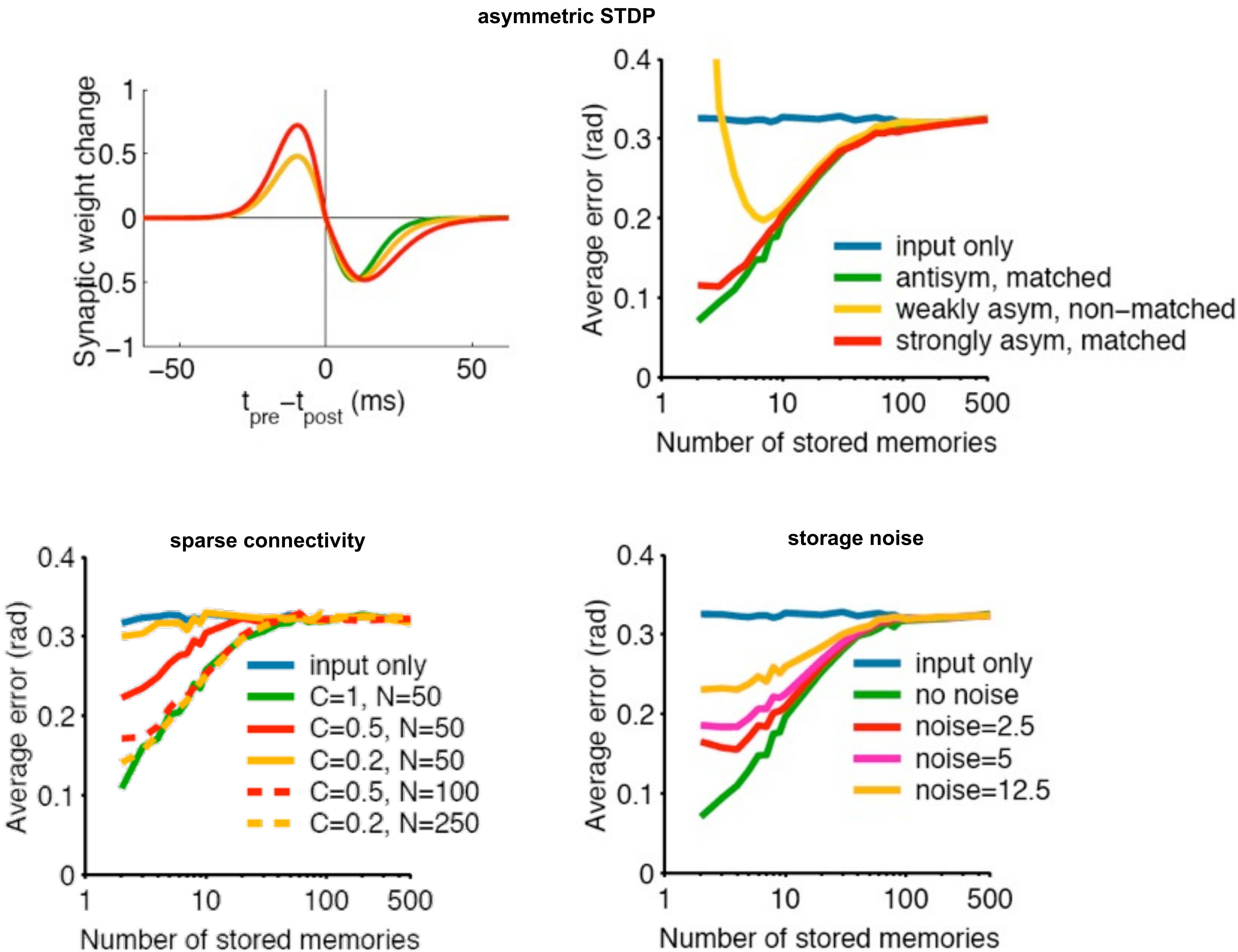
asymmetric STDP



sparse connectivity

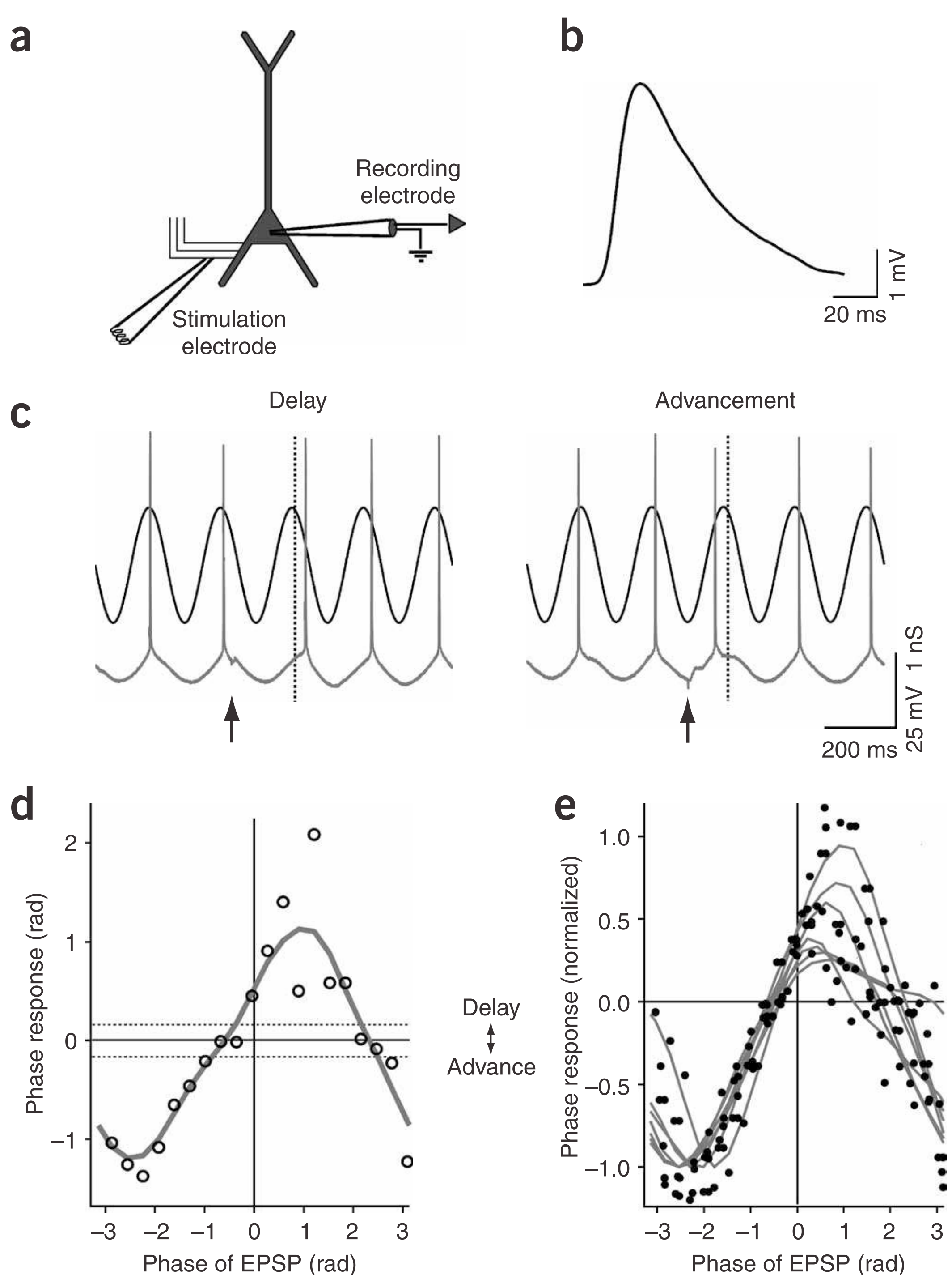


ROBUSTNESS OF RECALL PERFORMANCE

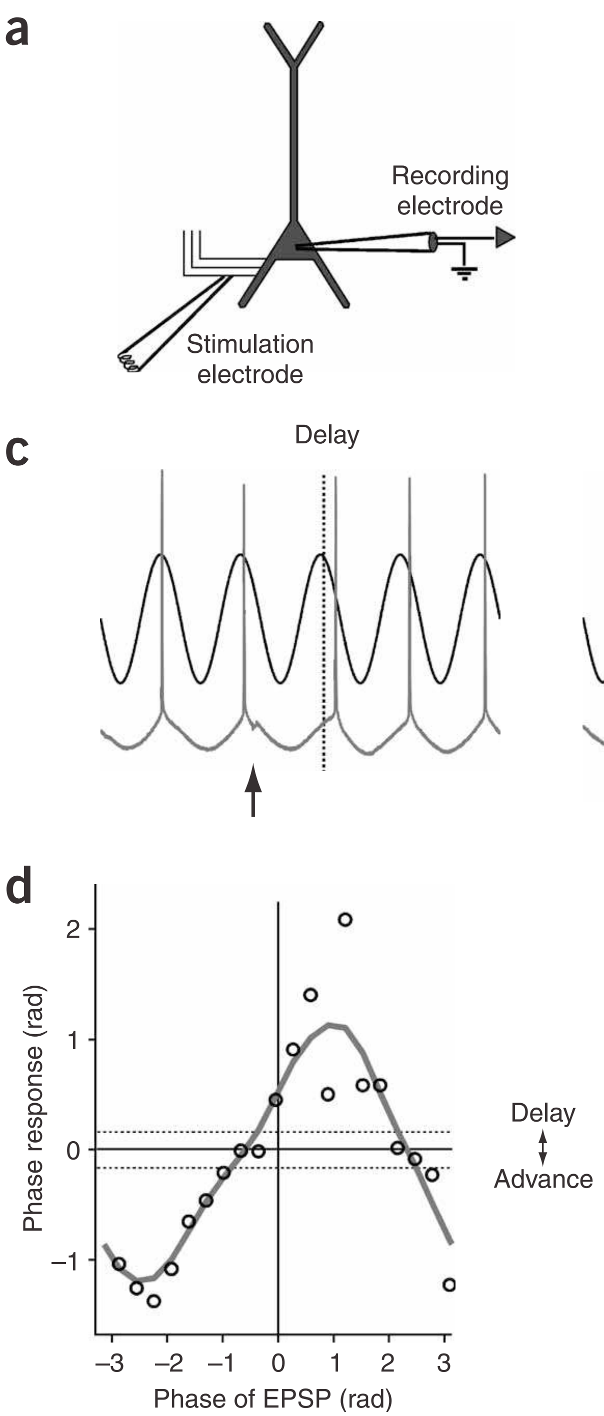


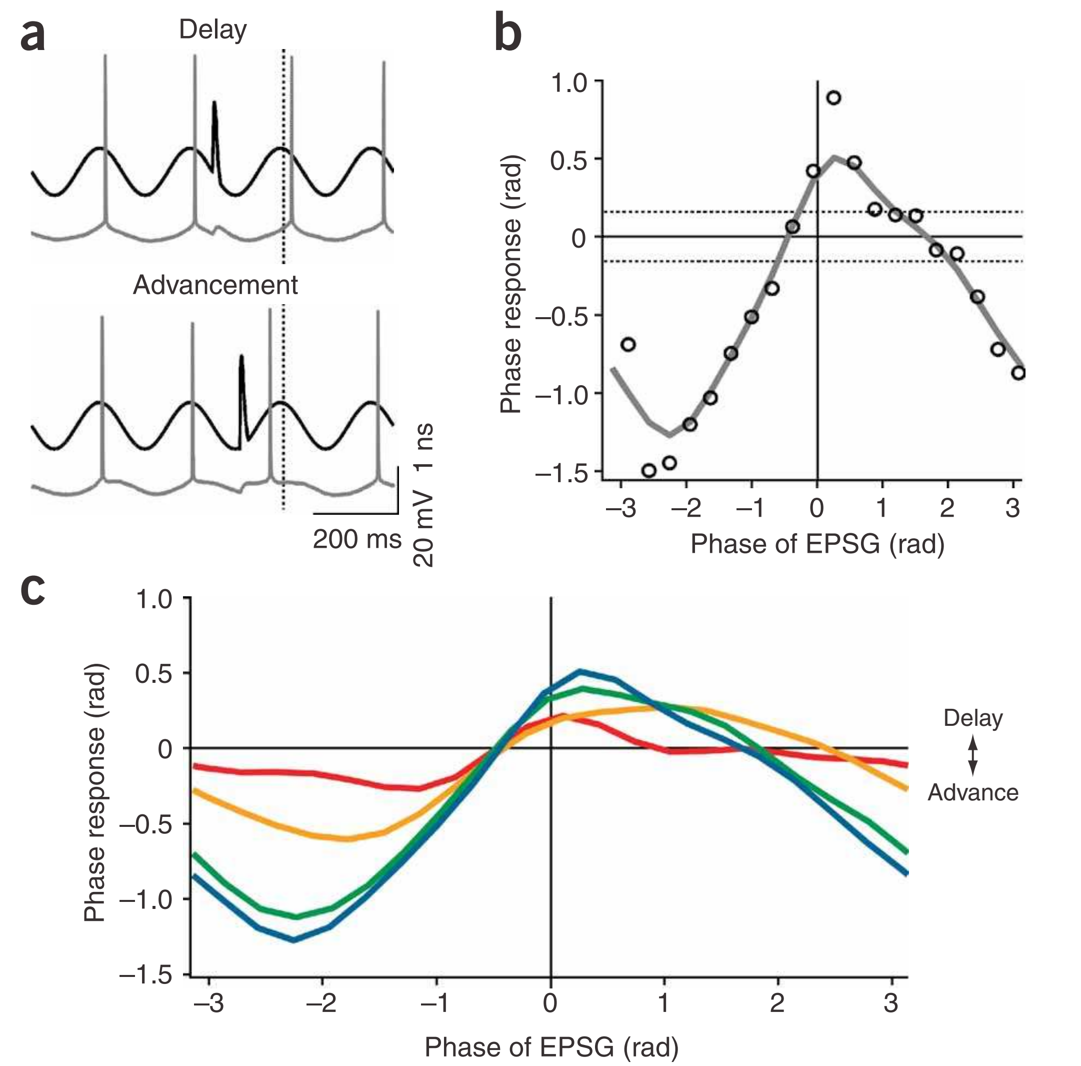
Lengyel et al, Nat Neurosci 2005

TESTING THE PREDICTION *IN VITRO*



TESTING THE PREDICTION *IN VITRO*





Lengyel et al, Nat Neurosci 2005

DISCRETE-STATE SYNAPSES

DISCRETE-STATE SYNAPSES

$$W_{ij} = \sum_{m=1}^{M} \Omega\left(x_i^{(m)}, x_j^{(m)}\right)$$

DISCRETE-STATE SYNAPSES

$$W_{ij} = \sum_{m=1}^{M} \Omega\left(x_i^{(m)}, x_j^{(m)}\right)$$ → **unlimited dynamic range …**

DISCRETE-STATE SYNAPSES

synapses with limited dynamic range

DISCRETE-STATE SYNAPSES

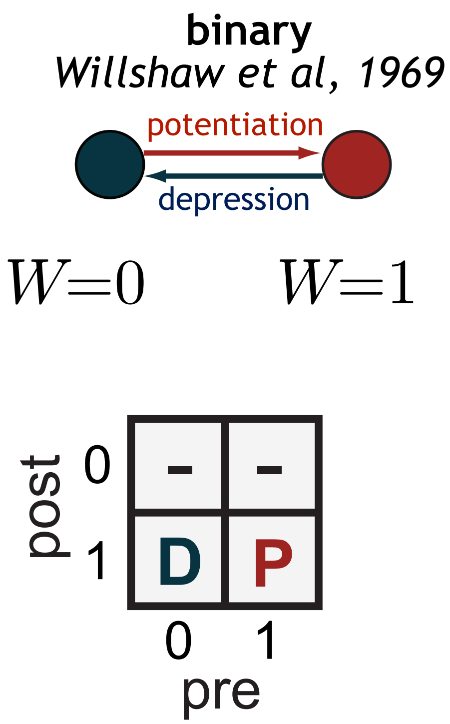
synapses with limited dynamic range

binary
Willshaw et al, 1969

$W\!=\!0$ $W\!=\!1$

DISCRETE-STATE SYNAPSES

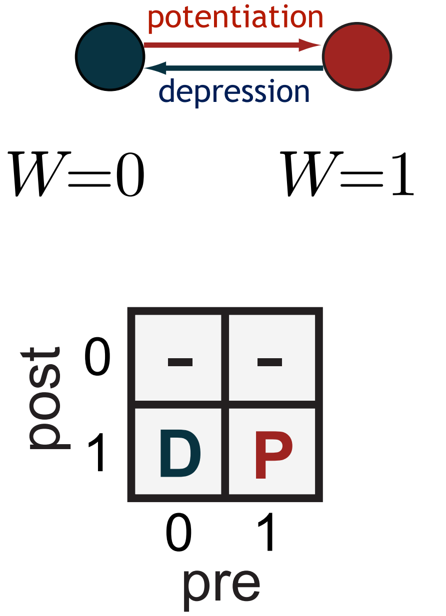
synapses with limited dynamic range



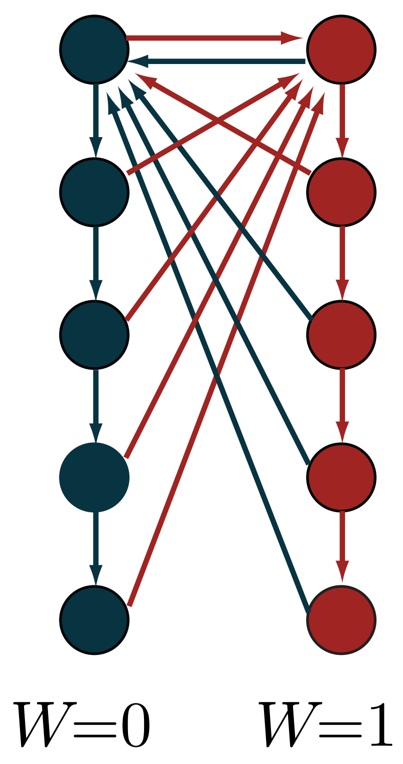
DISCRETE-STATE SYNAPSES

synapses with limited dynamic range

binary
Willshaw et al, 1969



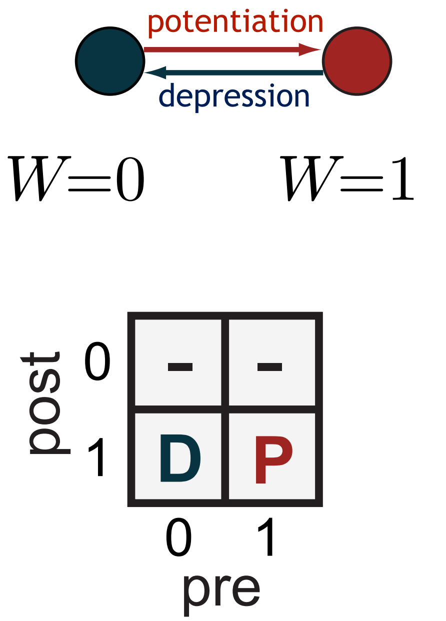
cascade
Fusi et al, 2005



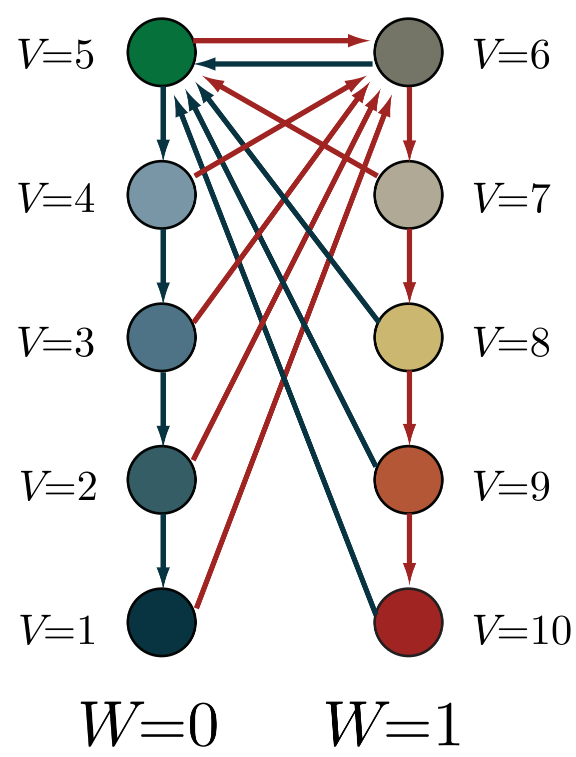
DISCRETE-STATE SYNAPSES

synapses with limited dynamic range

binary
Willshaw et al, 1969



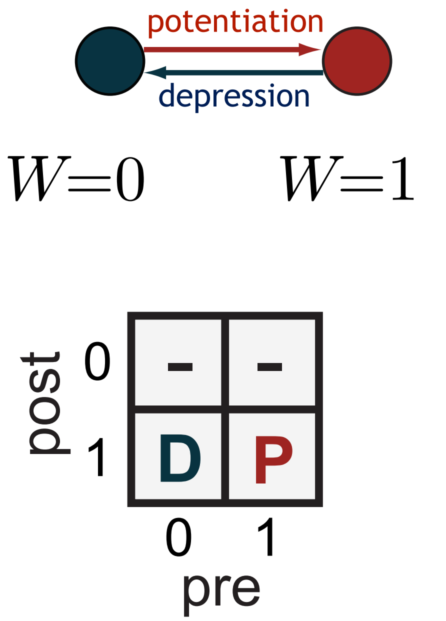
cascade
Fusi et al, 2005



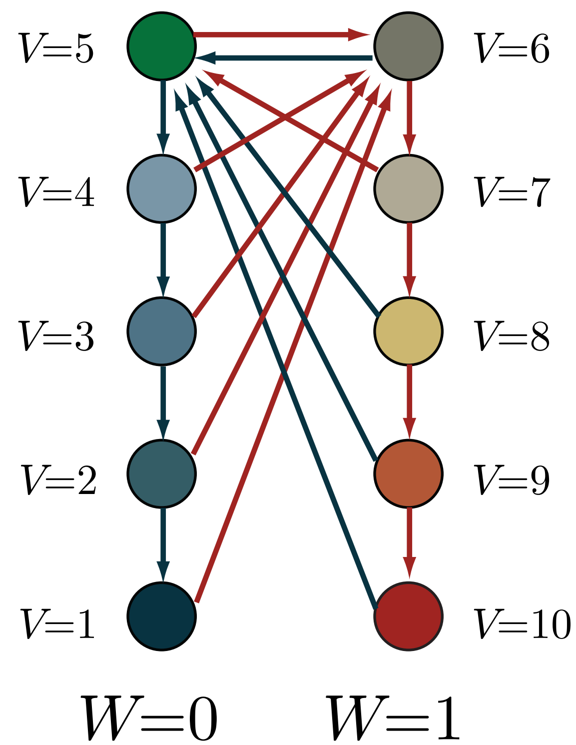
DISCRETE-STATE SYNAPSES

synapses with limited dynamic range

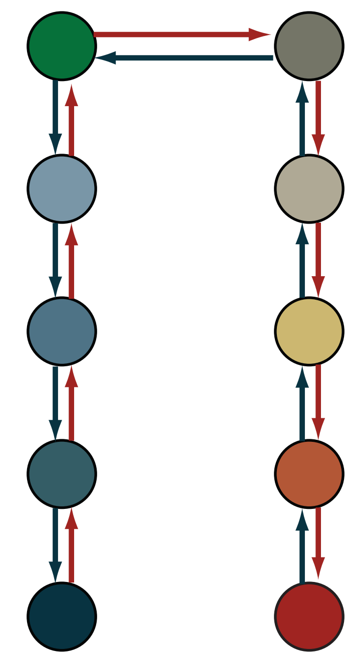
binary
Willshaw et al, 1969



cascade
Fusi et al, 2005



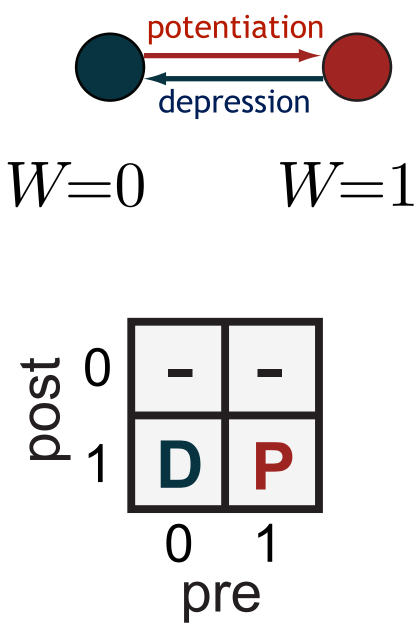
serial
Leibold & Kempter, 2008



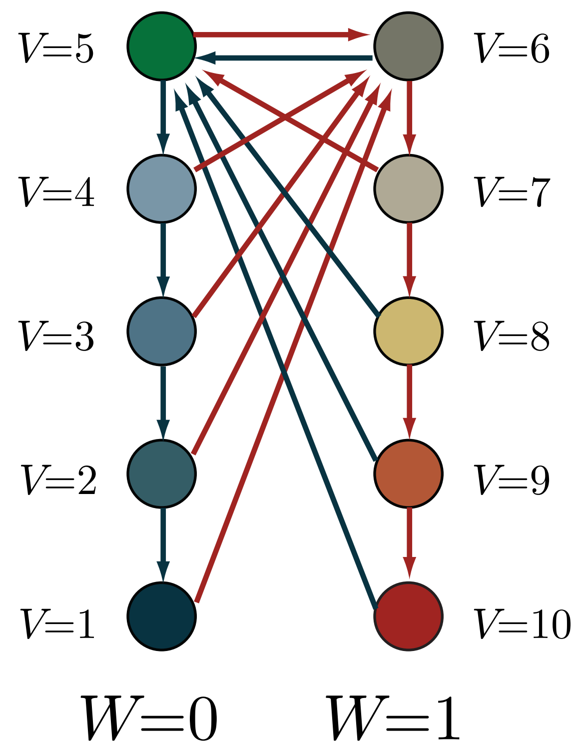
DISCRETE-STATE SYNAPSES

synapses with limited dynamic range

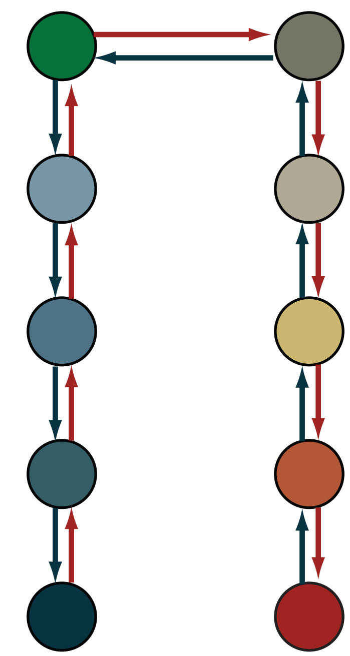
binary
Willshaw et al, 1969



cascade
Fusi et al, 2005

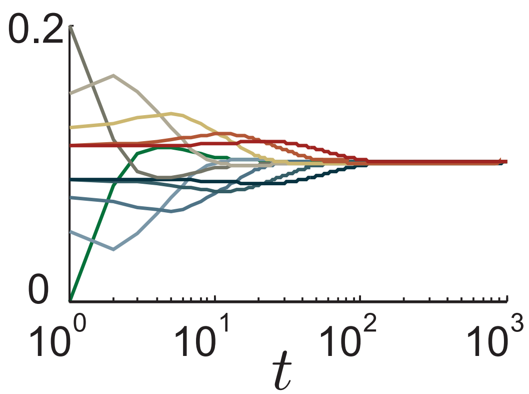


serial
Leibold & Kempter, 2008



... lead to palimpsest memories

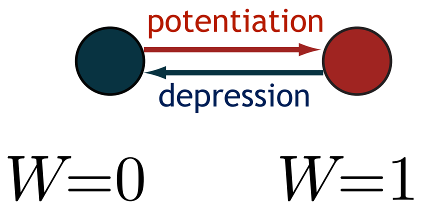
$$\mathrm{P}(V_{ij}|t, x_i = 1, x_j = 1)$$

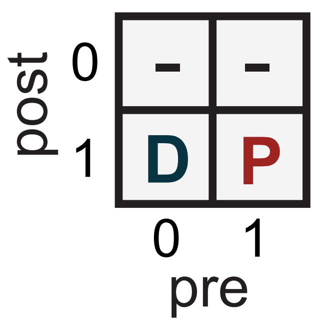


DISCRETE-STATE SYNAPSES

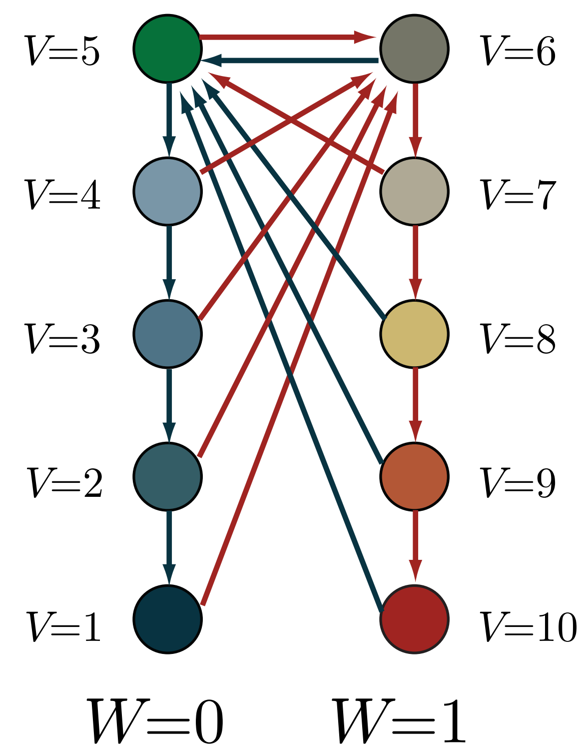
synapses with limited dynamic range

binary
Willshaw et al, 1969

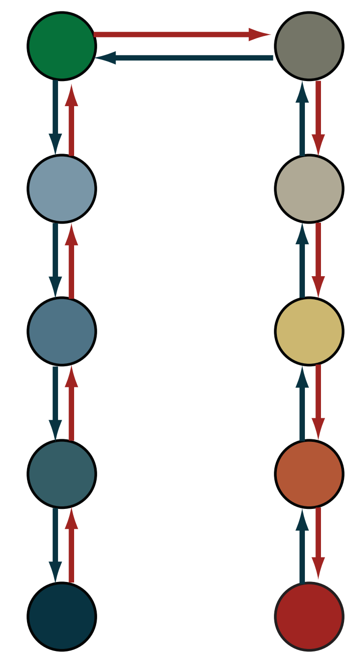




cascade
Fusi et al, 2005

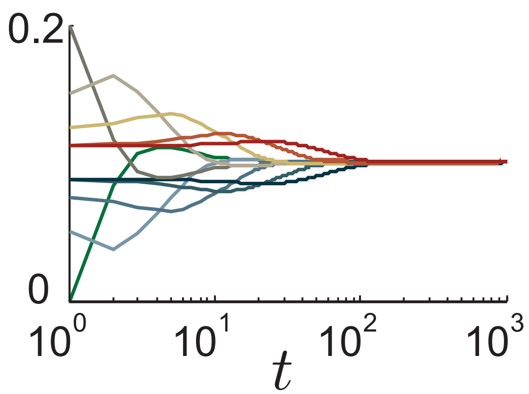


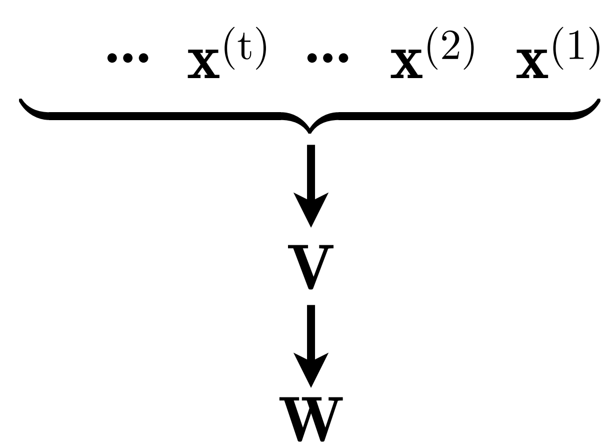
serial
Leibold & Kempter, 2008



... lead to palimpsest memories

$$\mathrm{P}(V_{ij}|t, x_i = 1, x_j = 1)$$

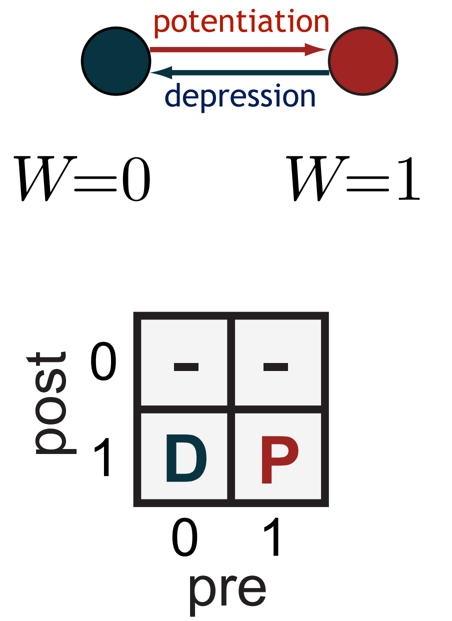




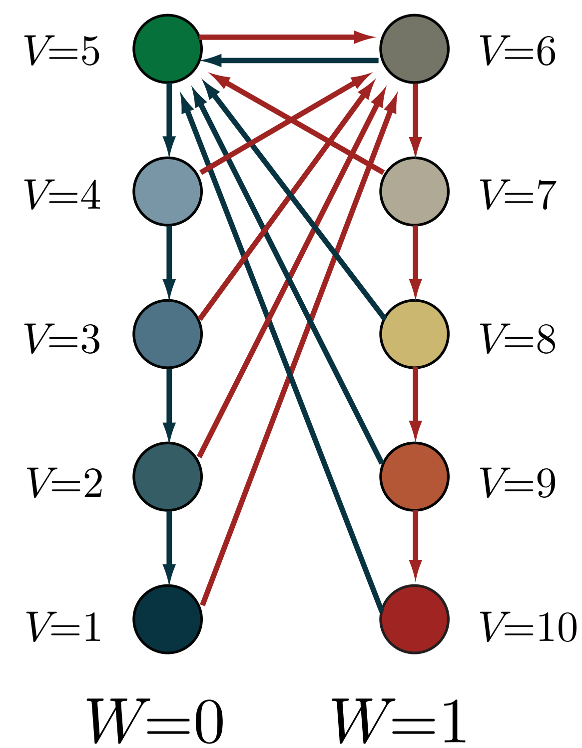
DISCRETE-STATE SYNAPSES

synapses with limited dynamic range

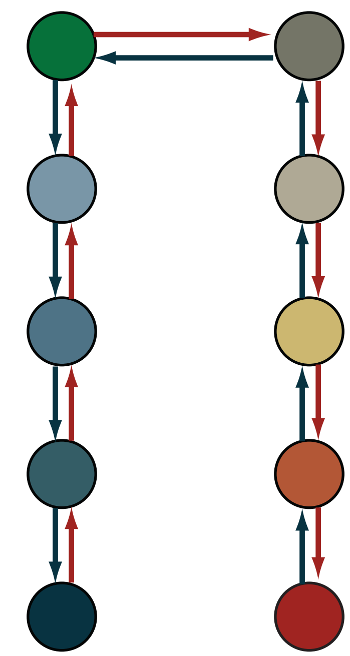
binary
Willshaw et al, 1969



cascade
Fusi et al, 2005

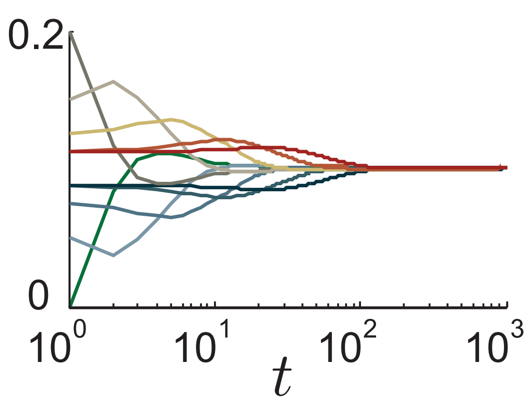


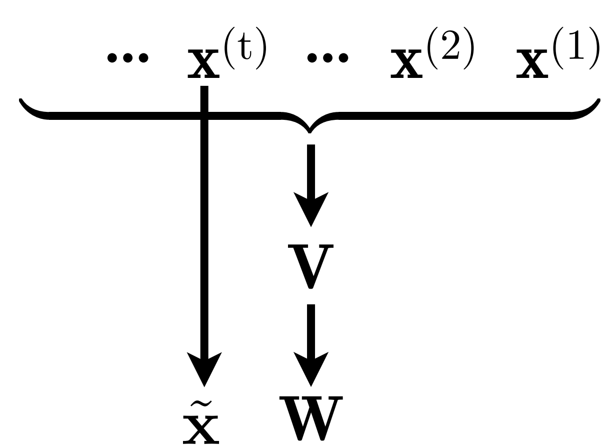
serial
Leibold & Kempter, 2008



... lead to palimpsest memories

$$\mathrm{P}(V_{ij}|t, x_i = 1, x_j = 1)$$

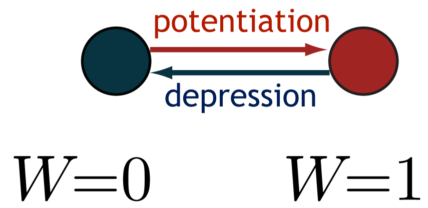


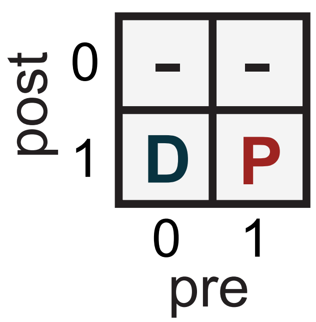


DISCRETE-STATE SYNAPSES

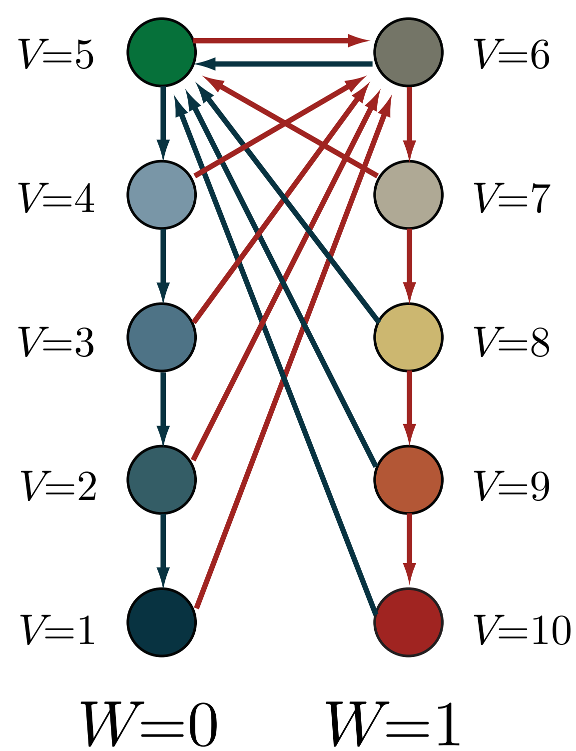
synapses with limited dynamic range

binary
Willshaw et al, 1969

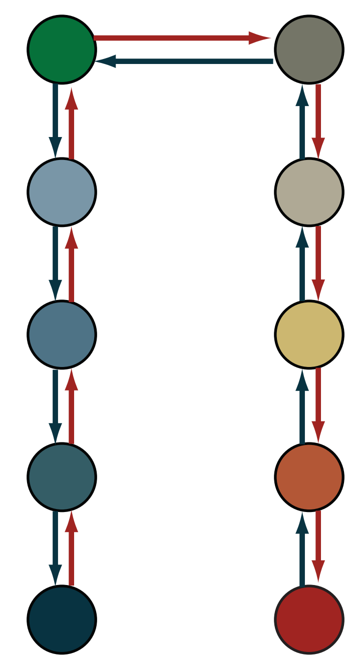




cascade
Fusi et al, 2005

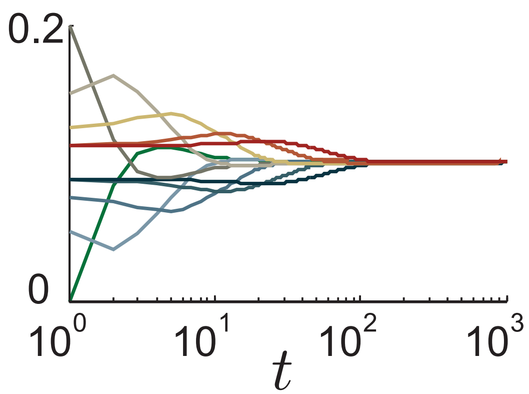


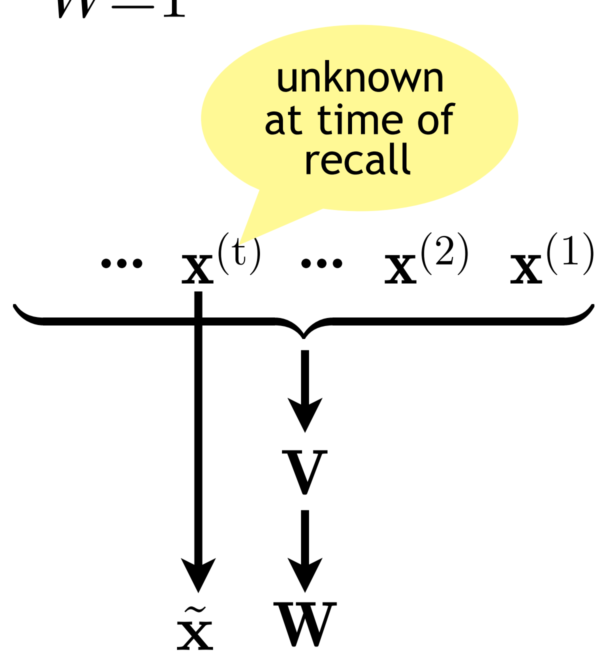
serial
Leibold & Kempter, 2008



... lead to palimpsest memories

$$\mathrm{P}(V_{ij}|t, x_i = 1, x_j = 1)$$

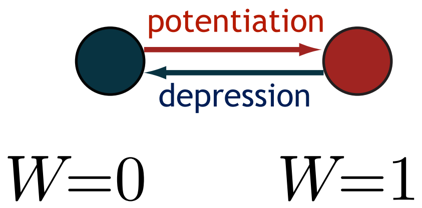


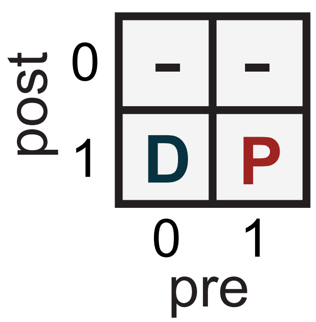


DISCRETE-STATE SYNAPSES

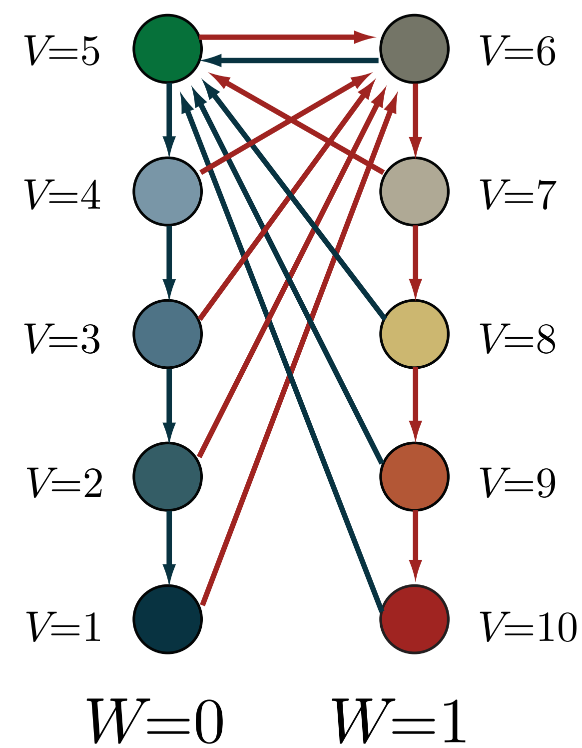
synapses with limited dynamic range

binary
Willshaw et al, 1969

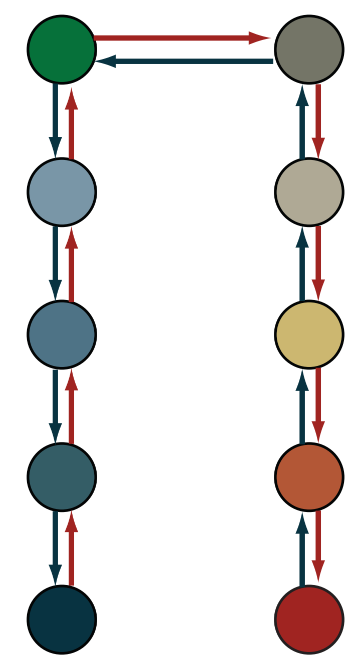




cascade
Fusi et al, 2005

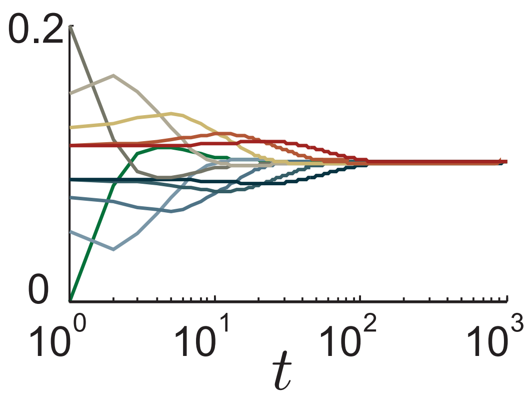


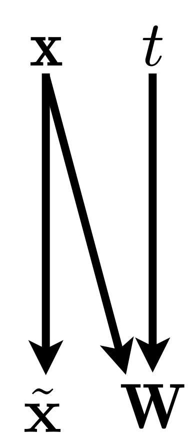
serial
Leibold & Kempter, 2008



... lead to palimpsest memories

$$\mathrm{P}(V_{ij}|t, x_i = 1, x_j = 1)$$

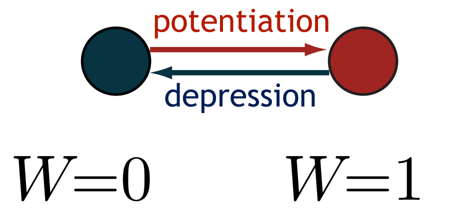


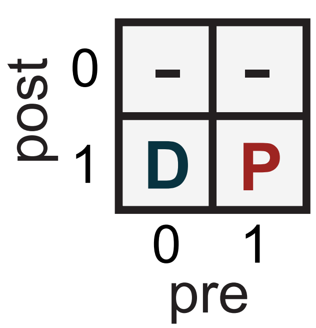


DISCRETE-STATE SYNAPSES

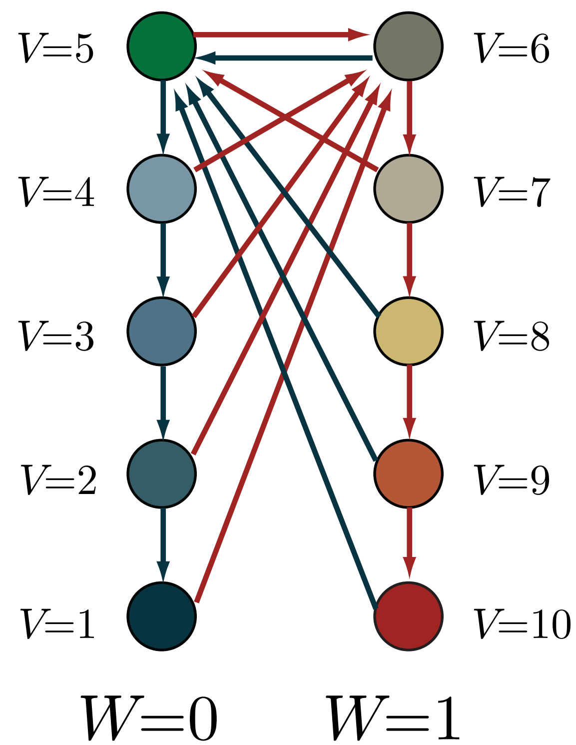
synapses with limited dynamic range

binary
Willshaw et al, 1969

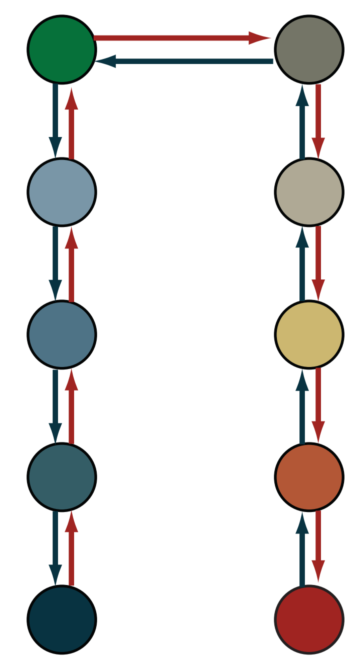




cascade
Fusi et al, 2005

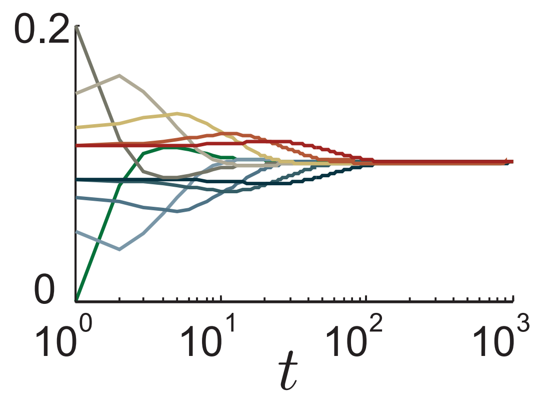


serial
Leibold & Kempter, 2008

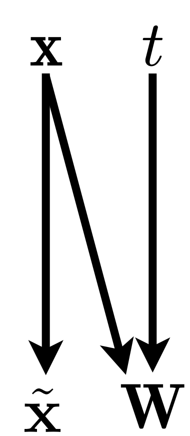


... lead to palimpsest memories

$$\mathrm{P}(V_{ij}|t, x_i = 1, x_j = 1)$$



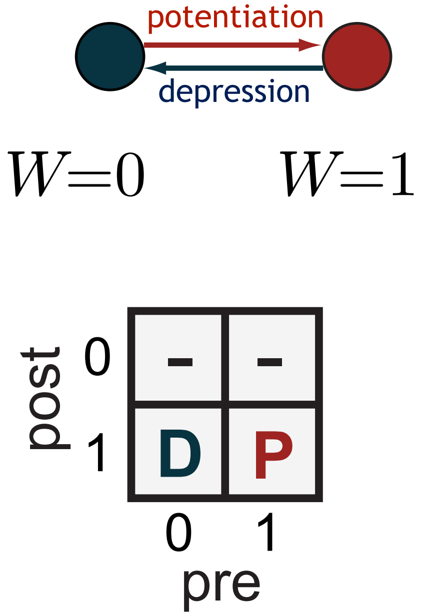
$$\mathrm{P}(\mathbf{x}|\tilde{\mathbf{x}}, \mathbf{W}) \propto \mathrm{P}(\mathbf{x})\ \mathrm{P}(\tilde{\mathbf{x}}|\mathbf{x})\ \mathrm{P}(\mathbf{W}|\mathbf{x})$$



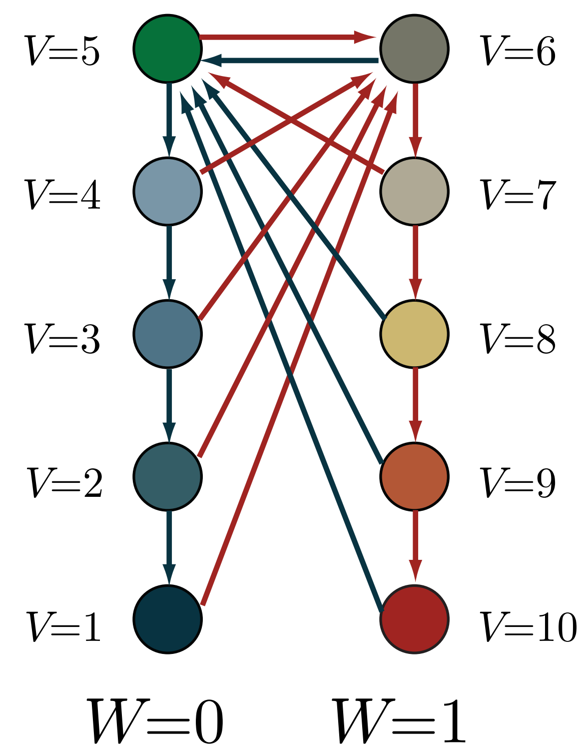
DISCRETE-STATE SYNAPSES

synapses with limited dynamic range

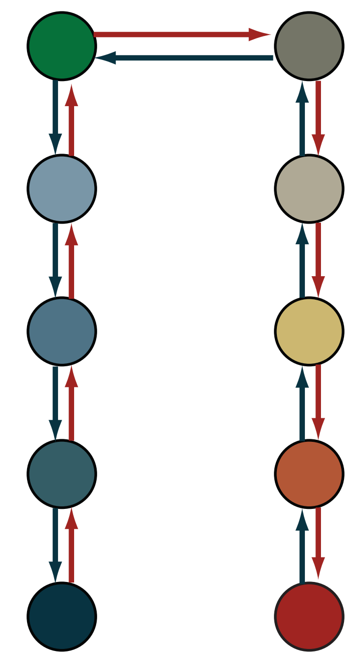
binary
Willshaw et al, 1969



cascade
Fusi et al, 2005

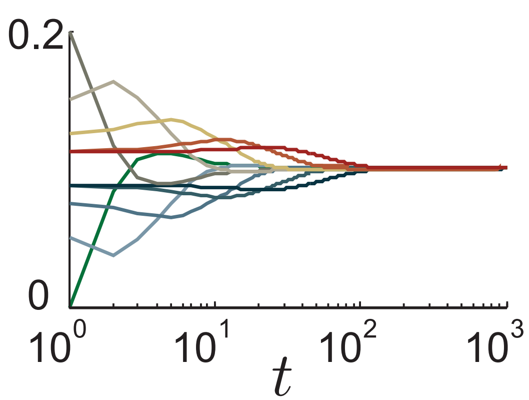


serial
Leibold & Kempter, 2008

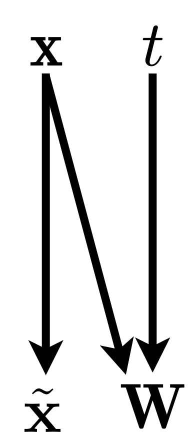


… lead to palimpsest memories

$$\mathrm{P}(V_{ij}|t, x_i = 1, x_j = 1)$$

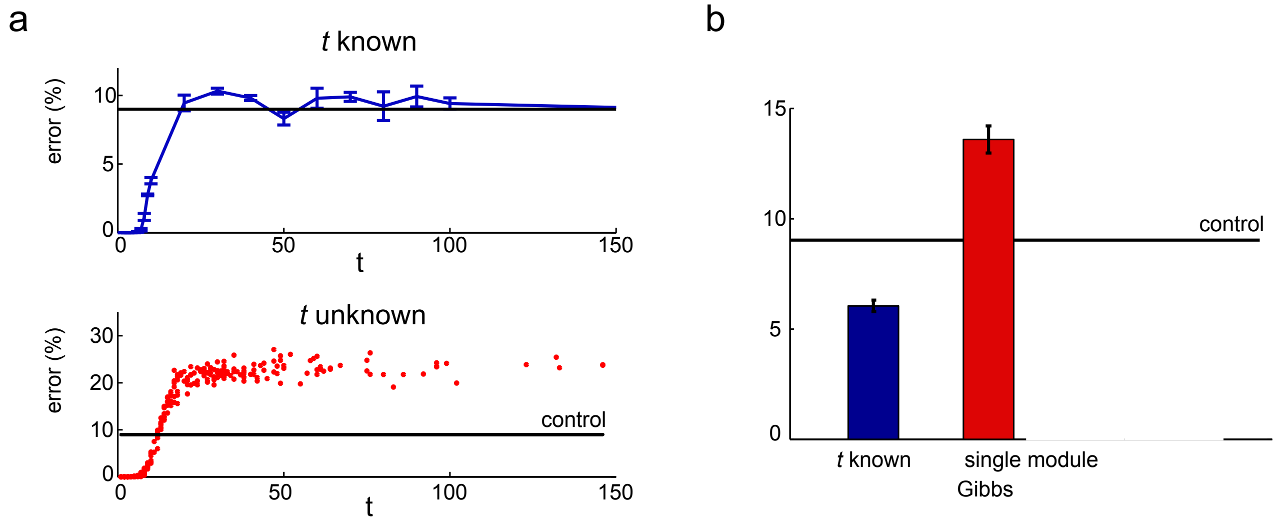


$$\mathrm{P}(\mathbf{x}|\tilde{\mathbf{x}}, \mathbf{W}) \propto \mathrm{P}(\mathbf{x})\ \mathrm{P}(\tilde{\mathbf{x}}|\mathbf{x}) \sum_t \mathrm{P}(\mathbf{W}|\mathbf{x}, t)\ \mathrm{P}(t)$$

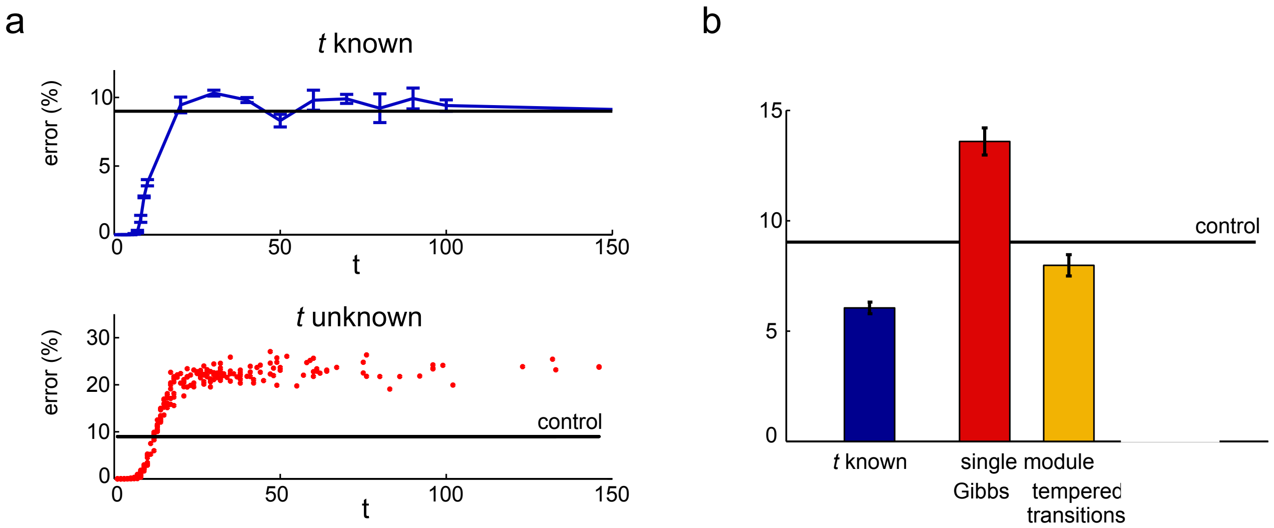


FAMILIARITY VS RECOLLECTION:
A SYSTEMS-LEVEL SOLUTION TO A SYNAPTIC-LEVEL PROBLEM?

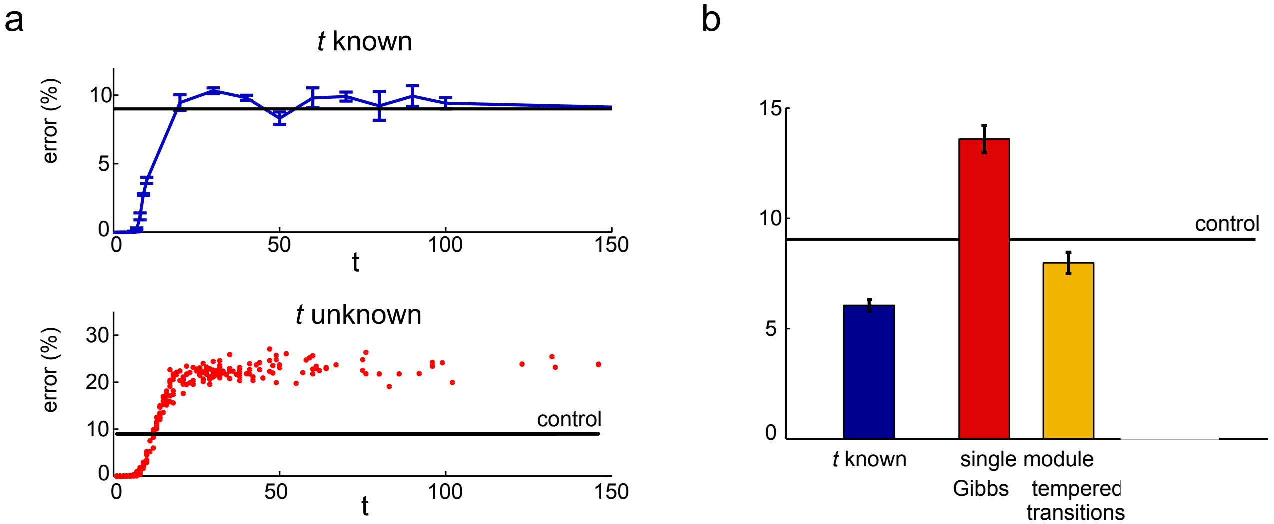
FAMILIARITY VS RECOLLECTION:
A SYSTEMS-LEVEL SOLUTION TO A SYNAPTIC-LEVEL PROBLEM?

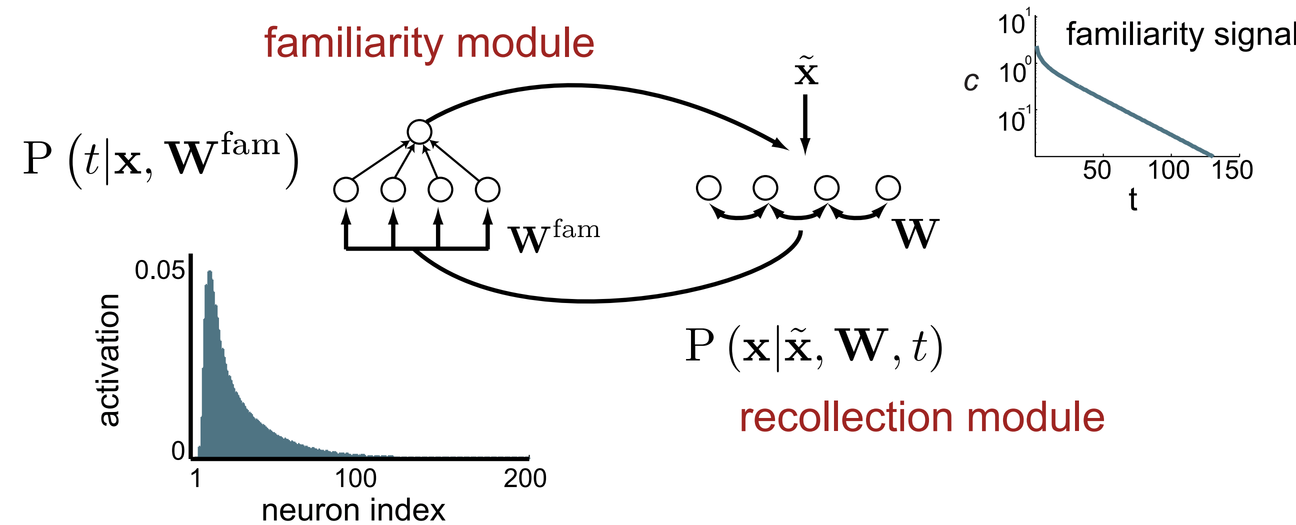


FAMILIARITY VS RECOLLECTION:
A SYSTEMS-LEVEL SOLUTION TO A SYNAPTIC-LEVEL PROBLEM?

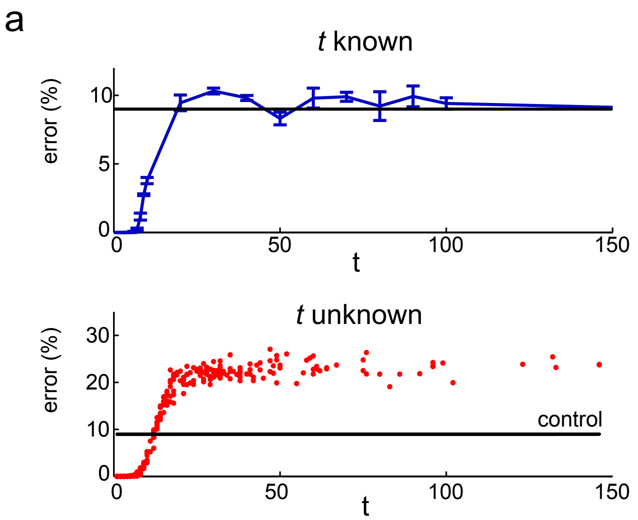


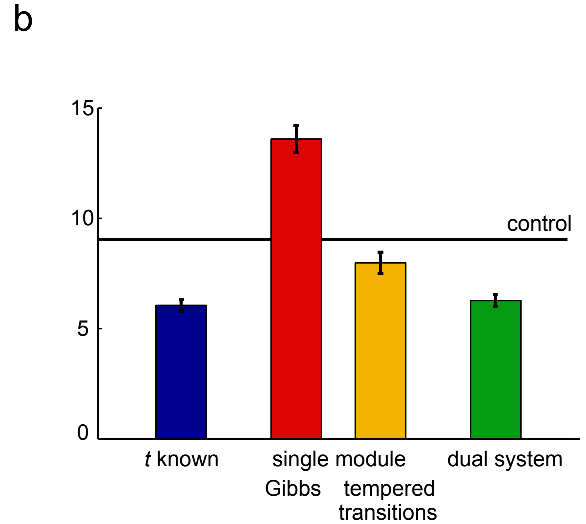
FAMILIARITY VS RECOLLECTION:
A SYSTEMS-LEVEL SOLUTION TO A SYNAPTIC-LEVEL PROBLEM?

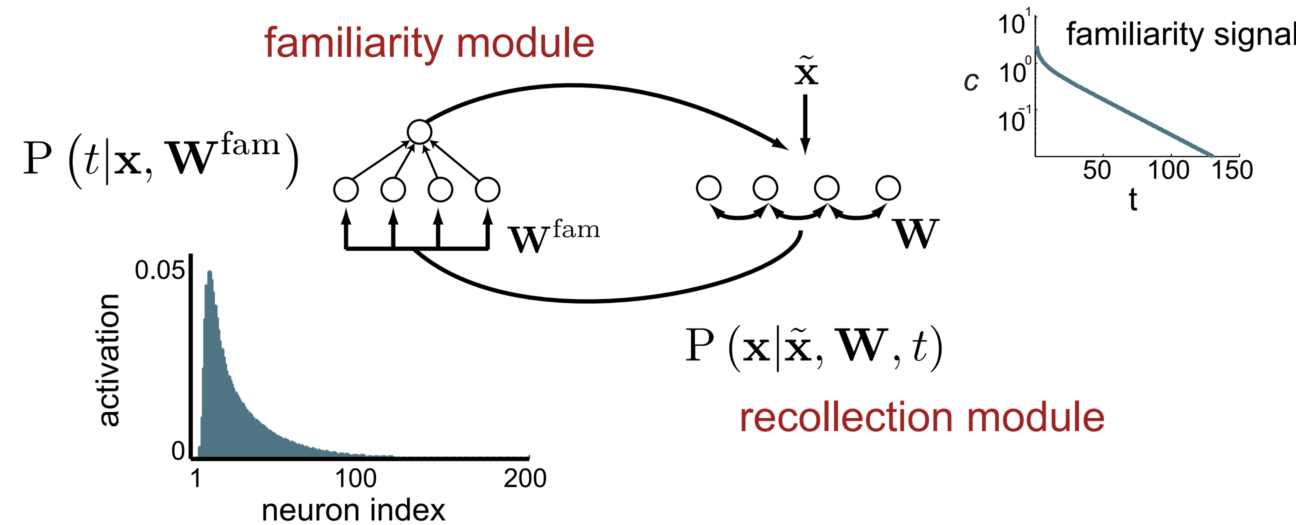




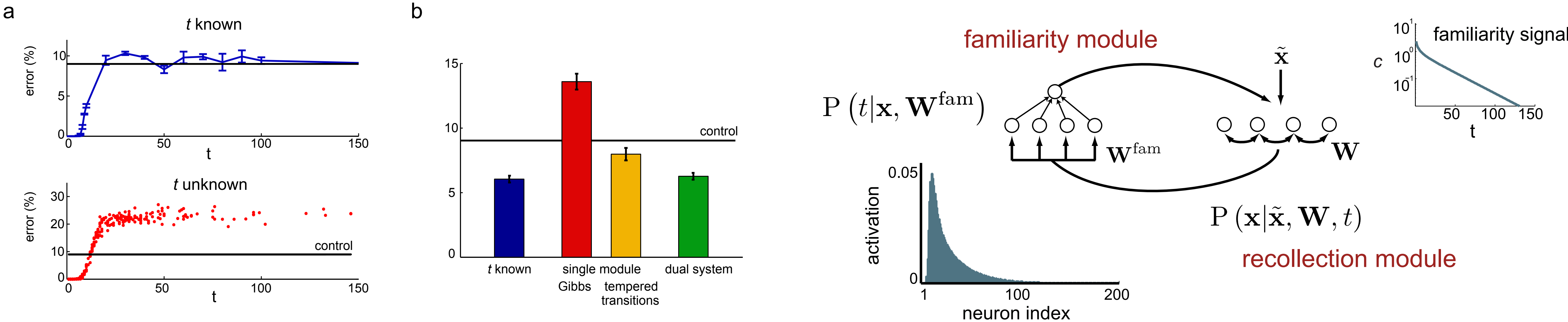
FAMILIARITY VS RECOLLECTION:
A SYSTEMS-LEVEL SOLUTION TO A SYNAPTIC-LEVEL PROBLEM?



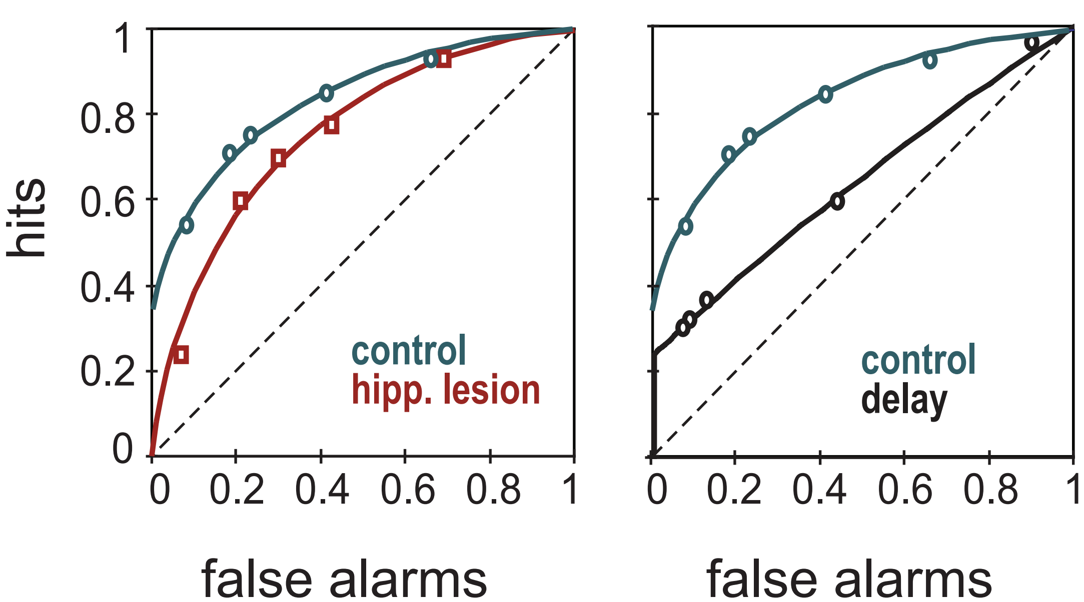




FAMILIARITY VS RECOLLECTION:
A SYSTEMS-LEVEL SOLUTION TO A SYNAPTIC-LEVEL PROBLEM?

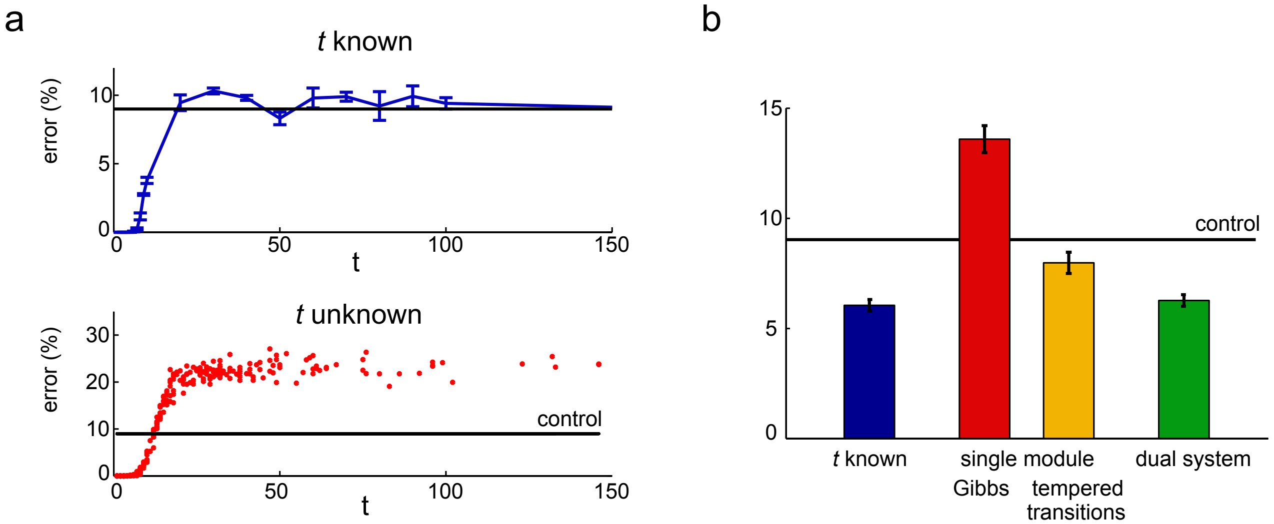


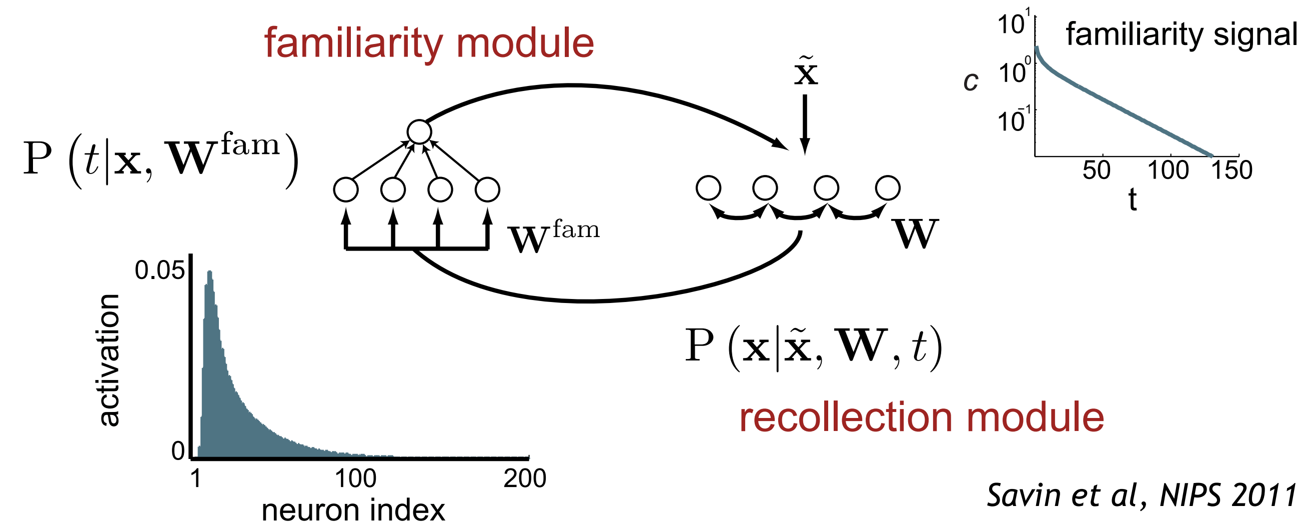
EXPERIMENTS



Fortin et al, Nature 2004

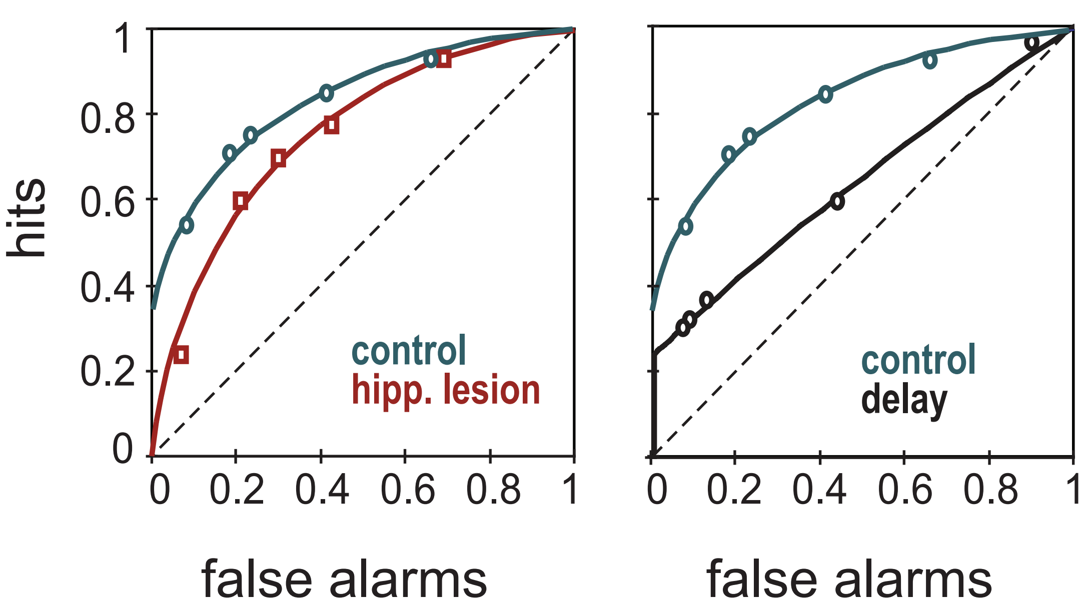
FAMILIARITY VS RECOLLECTION:
A SYSTEMS-LEVEL SOLUTION TO A SYNAPTIC-LEVEL PROBLEM?





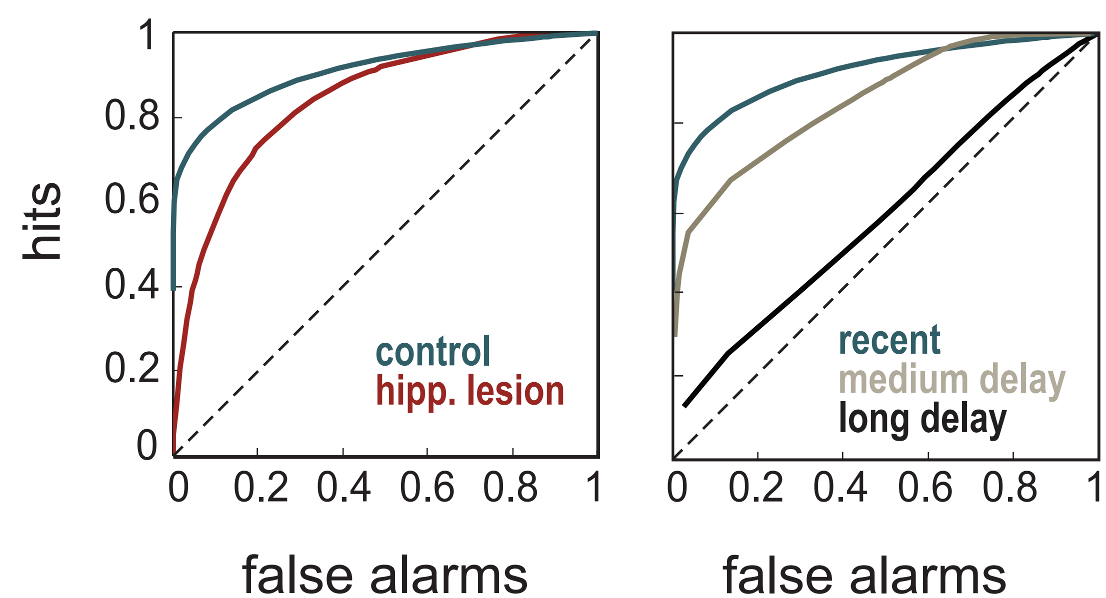
Savin et al, NIPS 2011

EXPERIMENTS



Fortin et al, Nature 2004

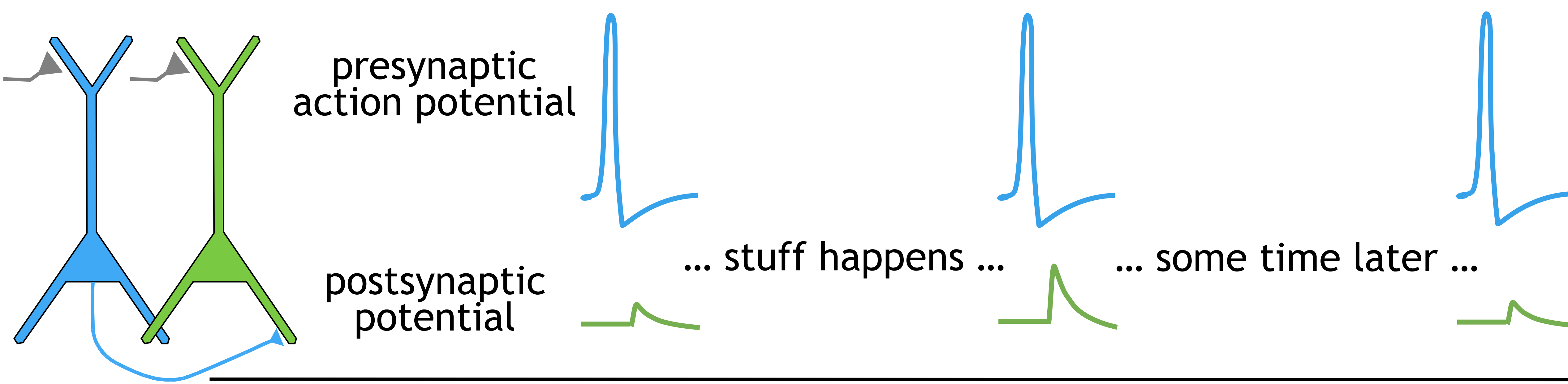
MODEL



SYNAPTIC PLASTICITY

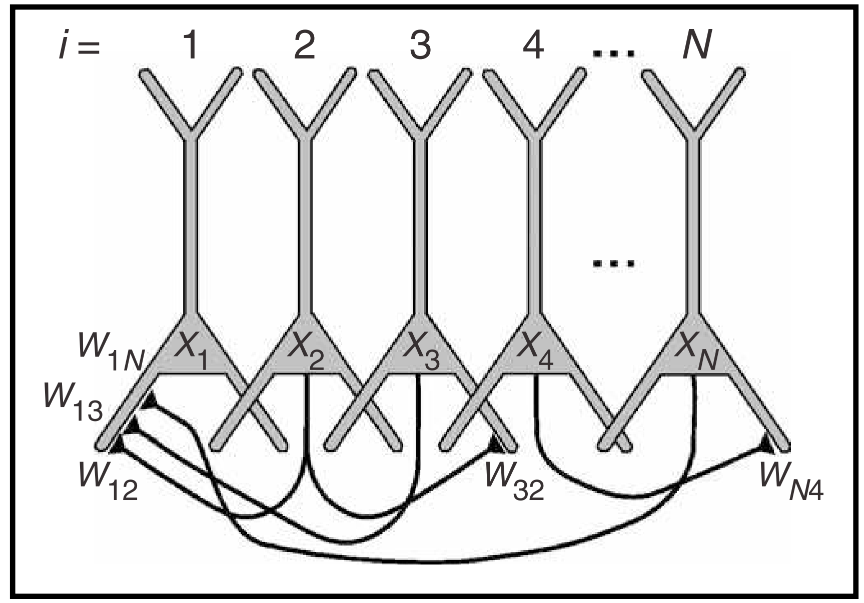
Synapses are computational devices that not only transmit action potential-encoded information, but also transform it. Neuronal information is often encoded by bursts or trains of action potentials. Synapses process such action potential bursts or trains in a synapse-specific manner that involves use-dependent changes in neurotransmitter release during the burst or train (referred to as short-term plasticity). In addition, synapses experience use-dependent long-term changes in synaptic transmission that adjust the “gain” of a synapse, and operate either pre- and/or postsynaptically (referred to as long-term plasticity)

Südhof, 2012

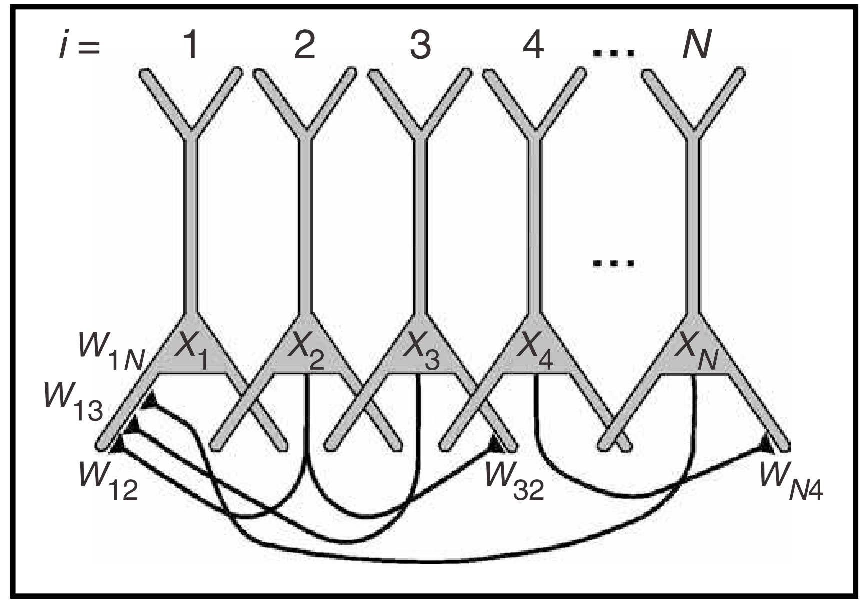


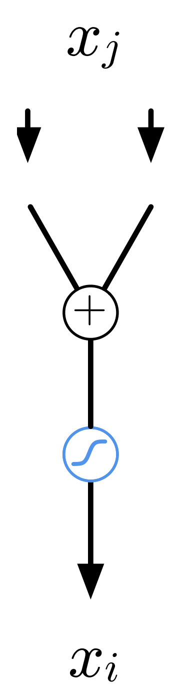
short-term plasticity	*presynaptically*	*10 — 100 ms*
long-term plasticity	*pre- and postsynaptically*	*10 min —*

CIRCUIT COMPUTATIONS → SINGLE CELL OPERATIONS



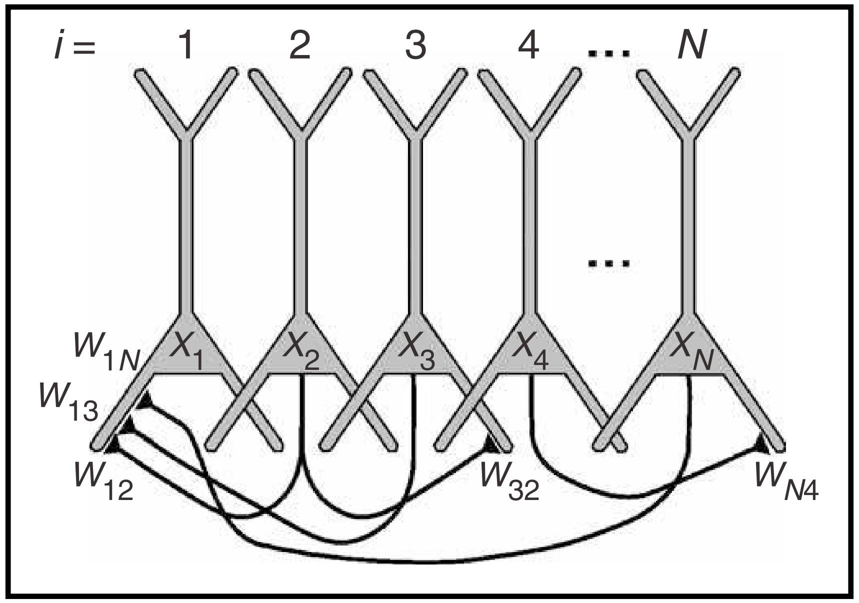
CIRCUIT COMPUTATIONS → SINGLE CELL OPERATIONS

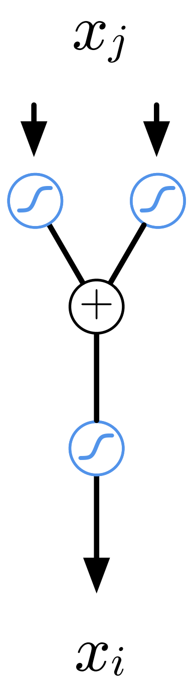




$$\frac{dx_i}{dt} \propto -\frac{x}{\tau_x} + F\left(\sum_j x_j\right)$$

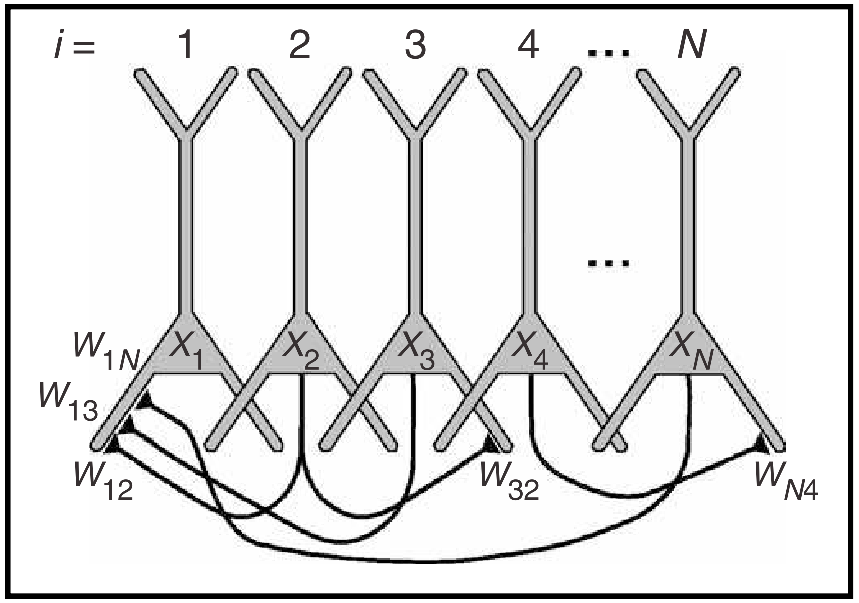
CIRCUIT COMPUTATIONS → SINGLE CELL OPERATIONS

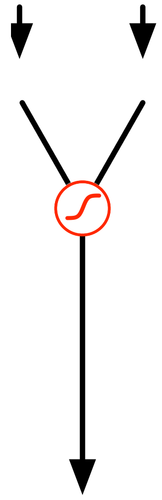




$$\frac{dx_i}{dt} \propto -\frac{x}{\tau_x} + F\left(\sum_j G(x_j)\right)$$

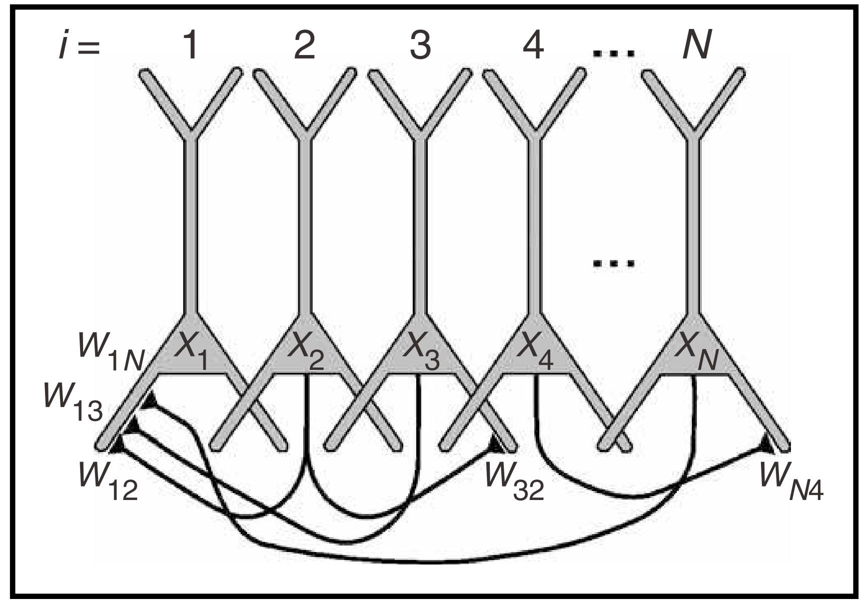
CIRCUIT COMPUTATIONS → SINGLE CELL OPERATIONS

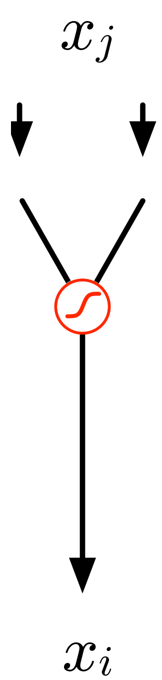




$$\frac{dx_i}{dt} \propto -\frac{x}{\tau_x} + F\Big(\{x_j\}_{j \neq i}\Big)$$

CIRCUIT COMPUTATIONS → SINGLE CELL OPERATIONS

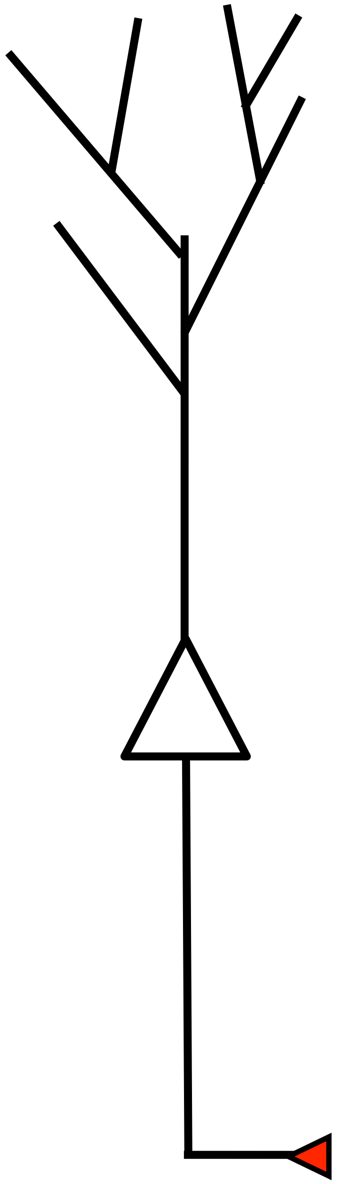




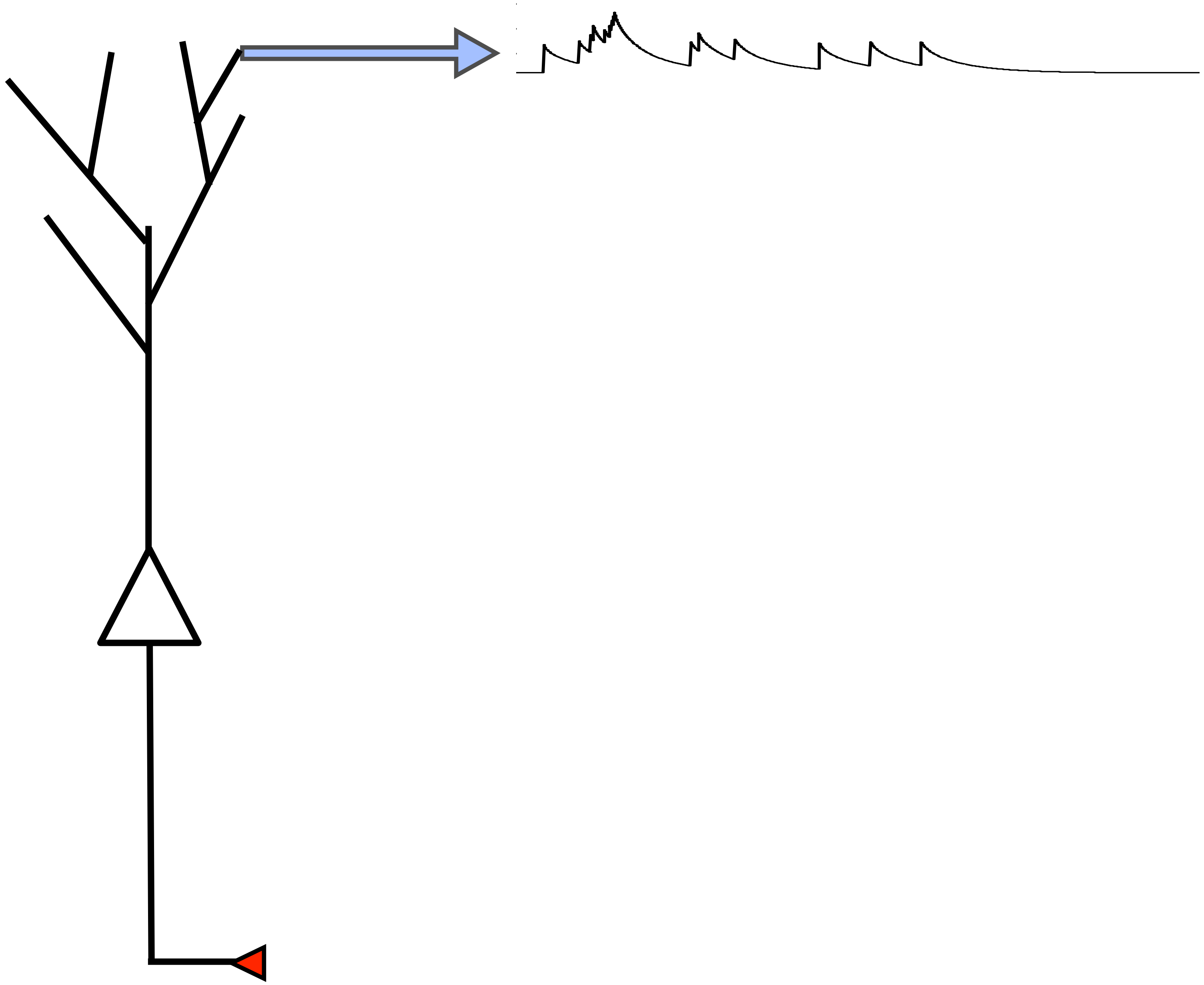
assumption: same kind of quantity

$$\frac{dx_i}{dt} \propto -\frac{x}{\tau_x} + F\Big(\{x_j\}_{j \neq i}\Big)$$

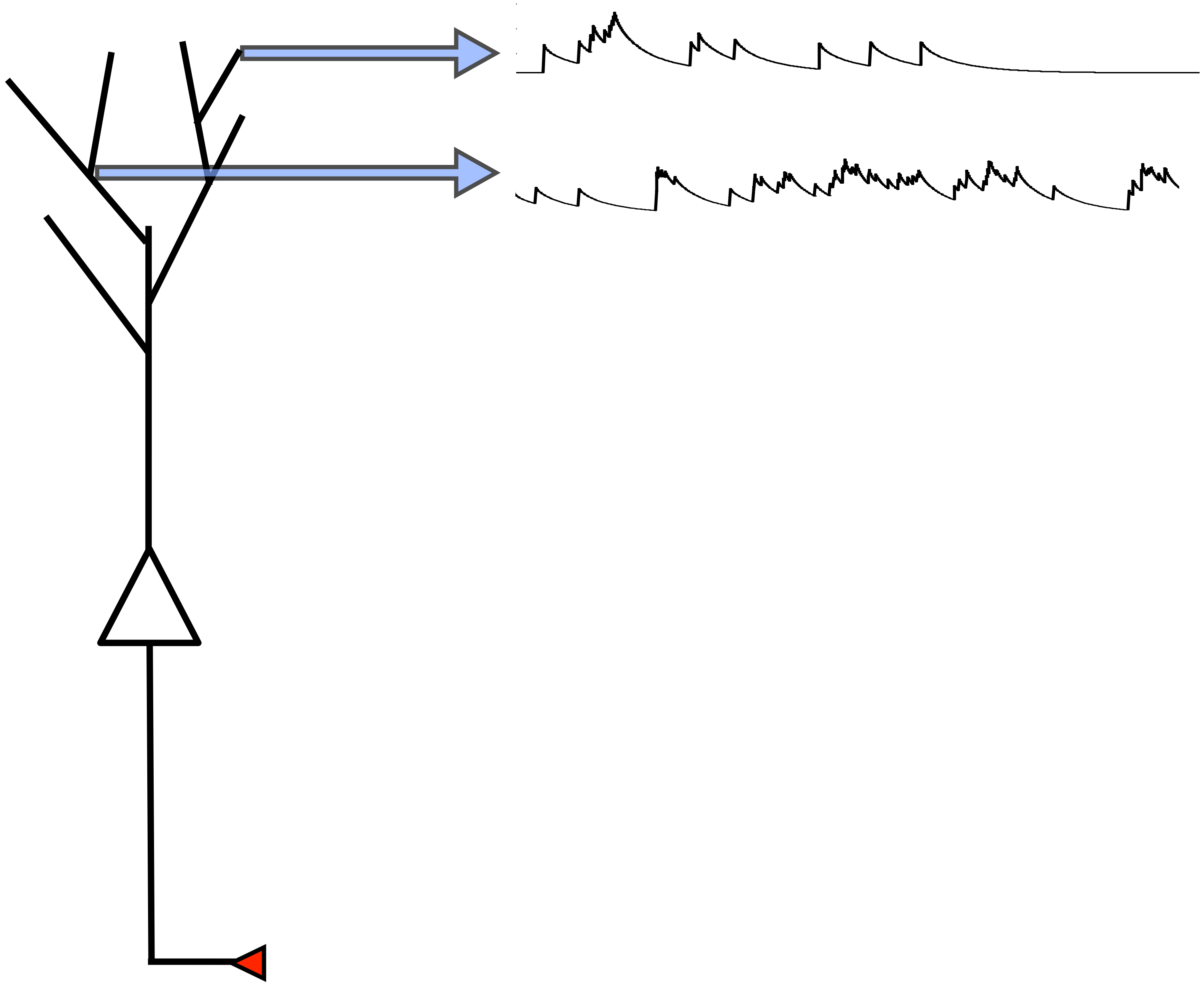
INFORMATION LOSS IN SYNAPTIC TRANSMISSION



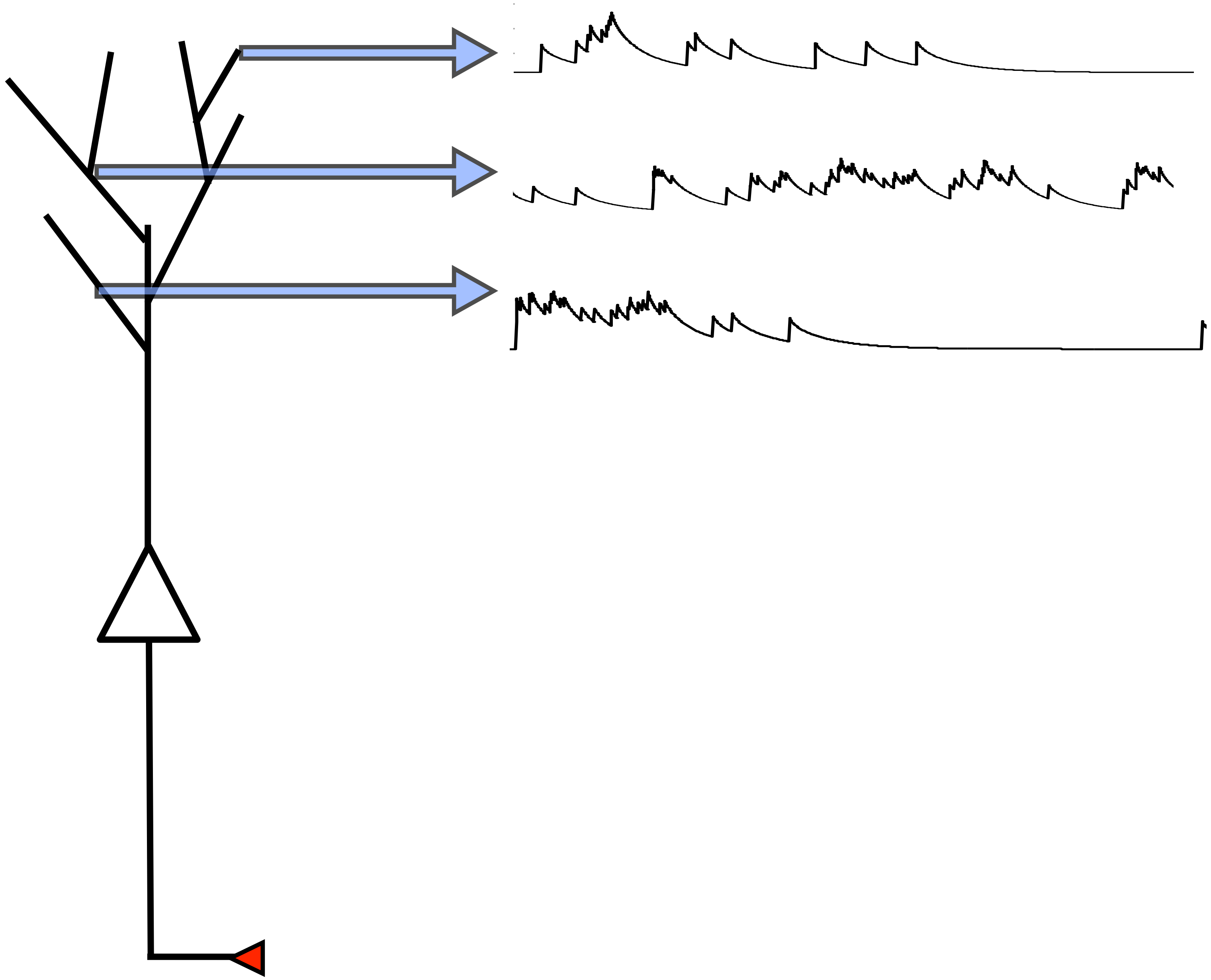
INFORMATION LOSS IN SYNAPTIC TRANSMISSION



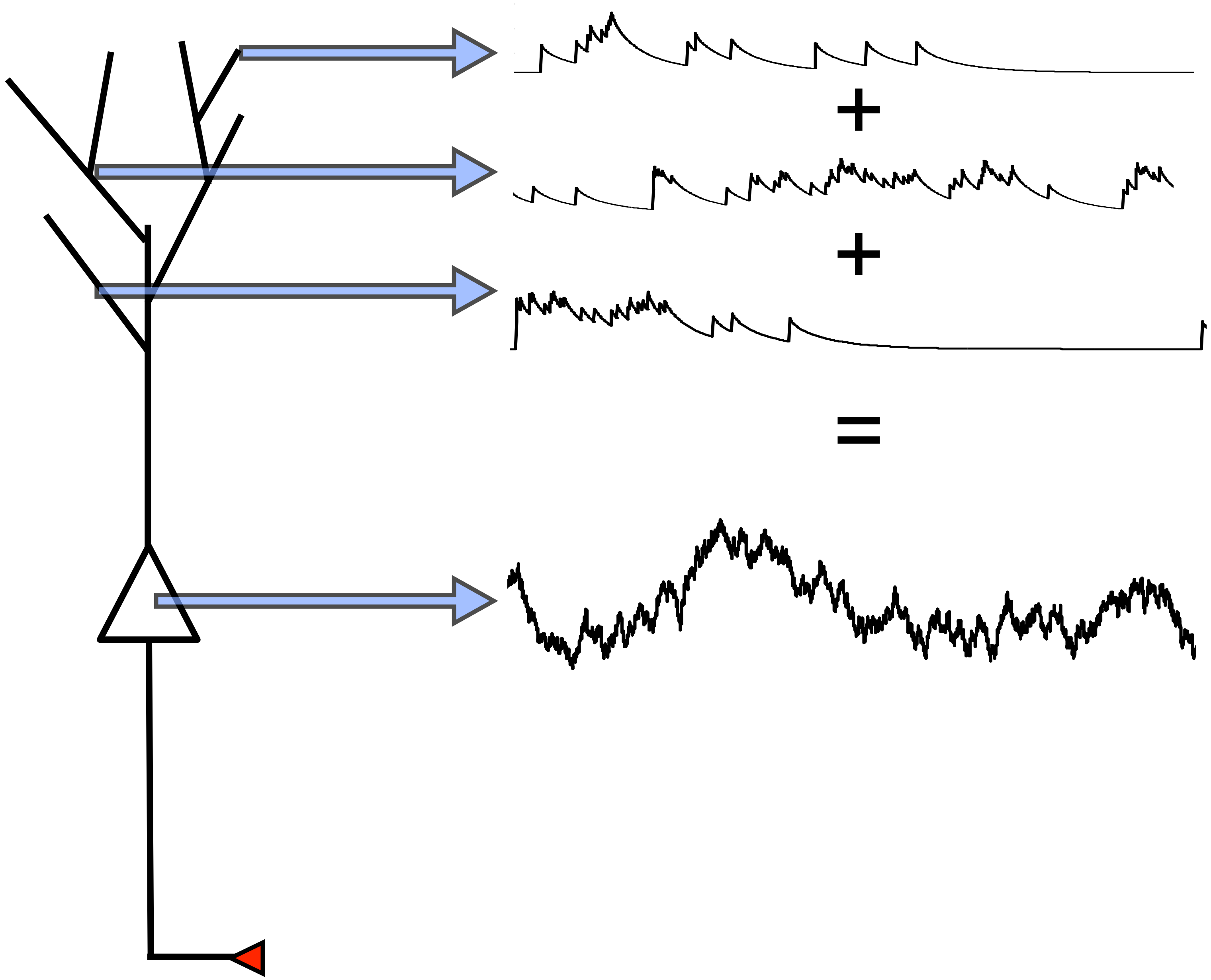
INFORMATION LOSS IN SYNAPTIC TRANSMISSION



INFORMATION LOSS IN SYNAPTIC TRANSMISSION

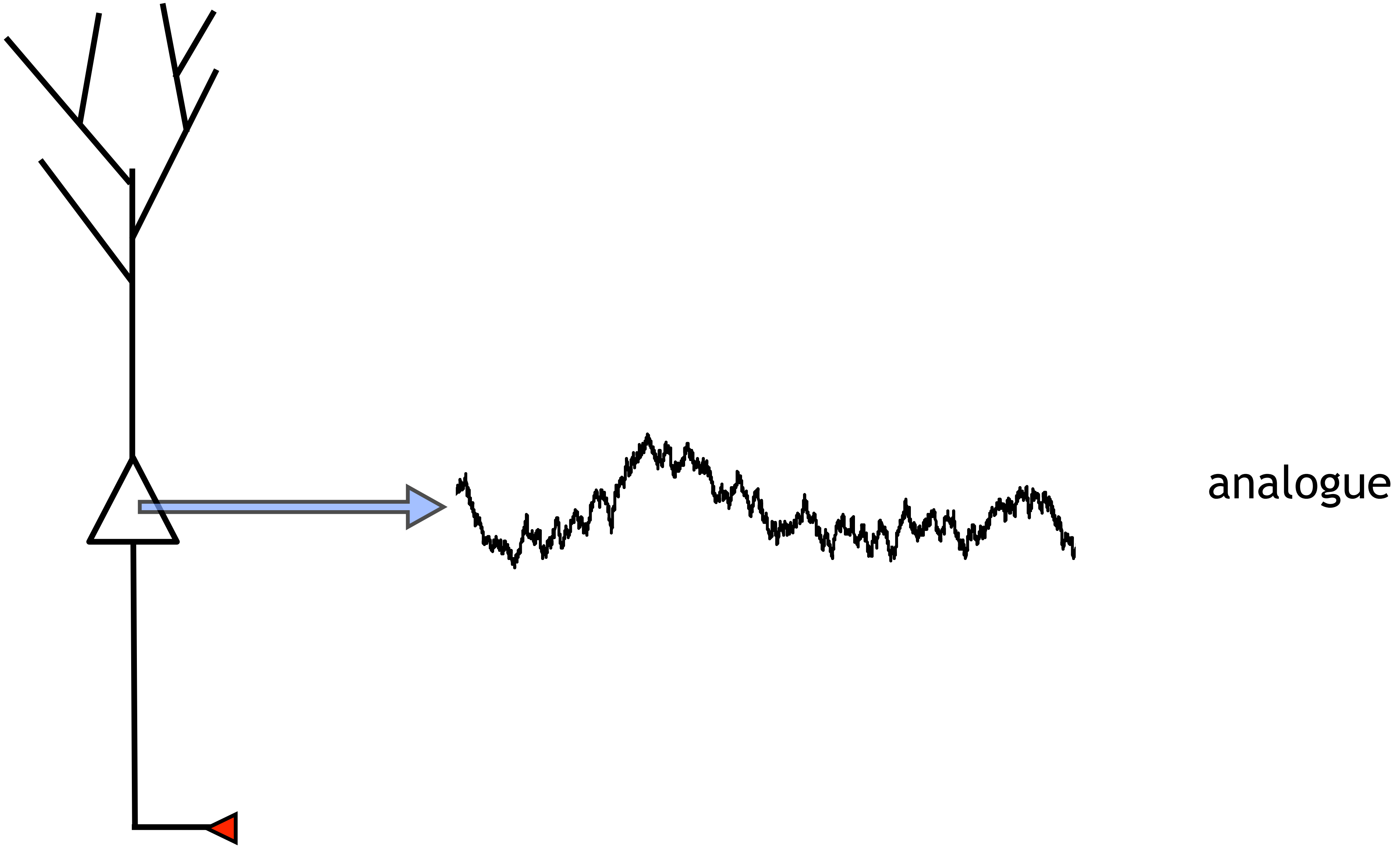


INFORMATION LOSS IN SYNAPTIC TRANSMISSION

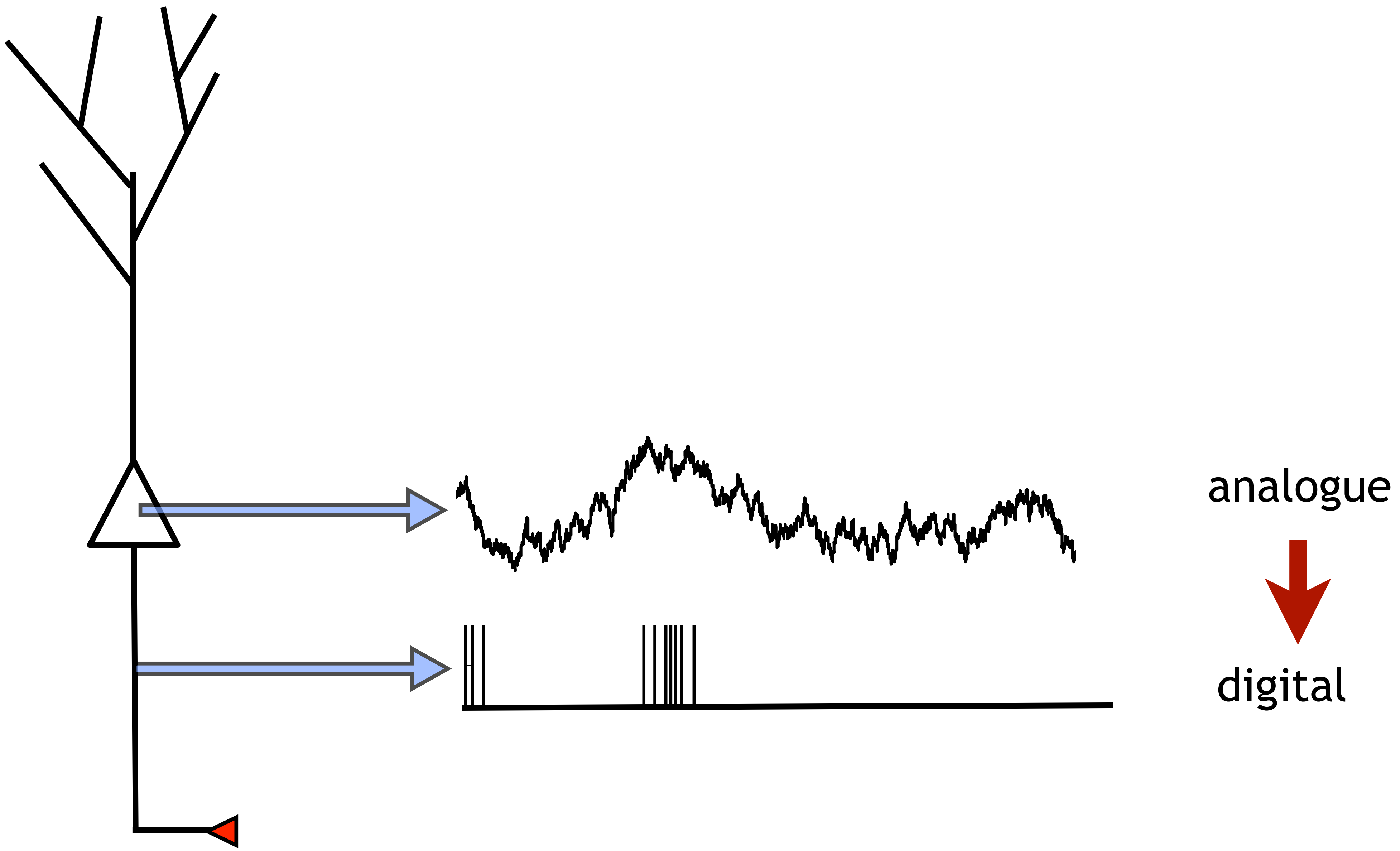


analogue

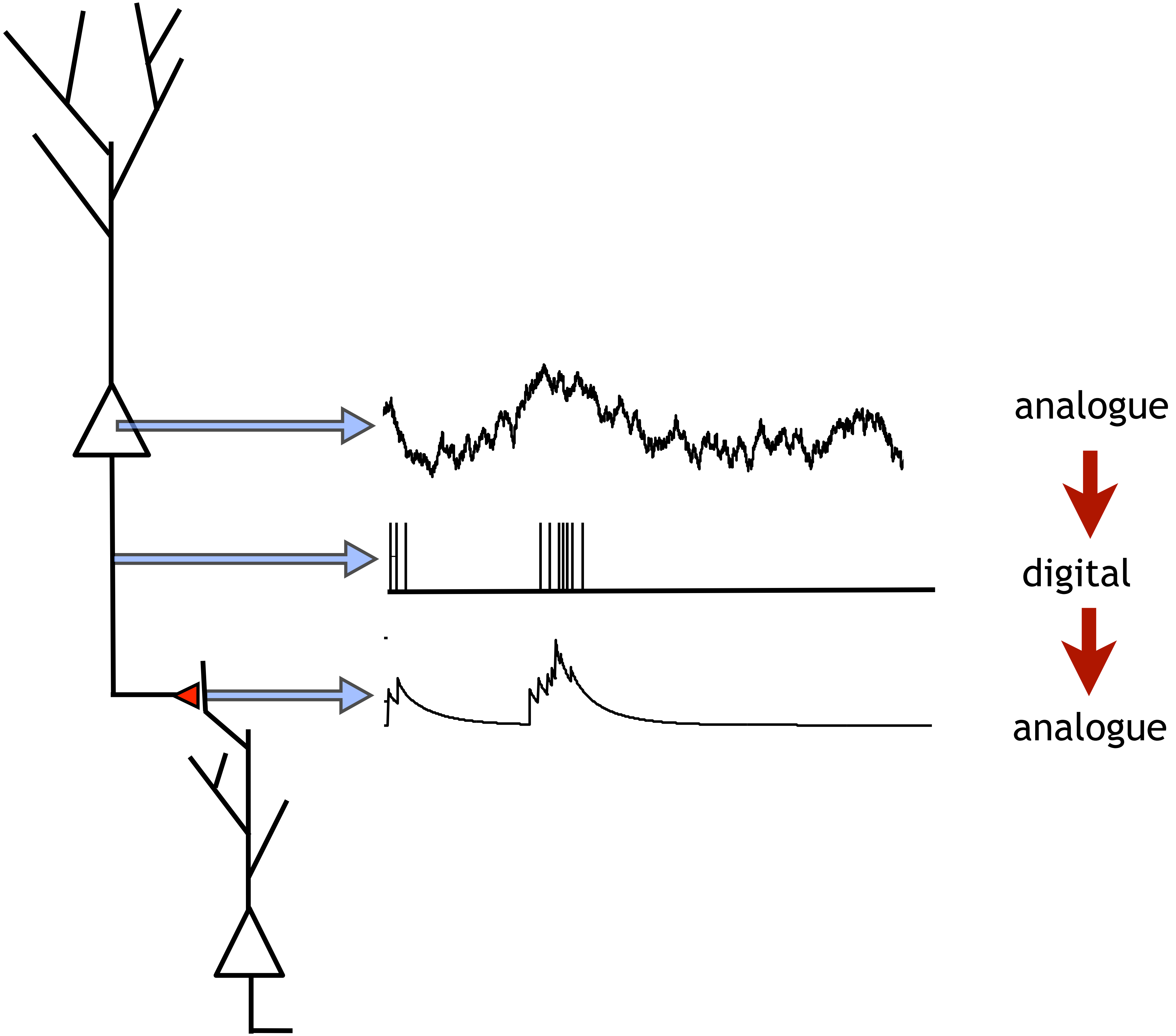
INFORMATION LOSS IN SYNAPTIC TRANSMISSION



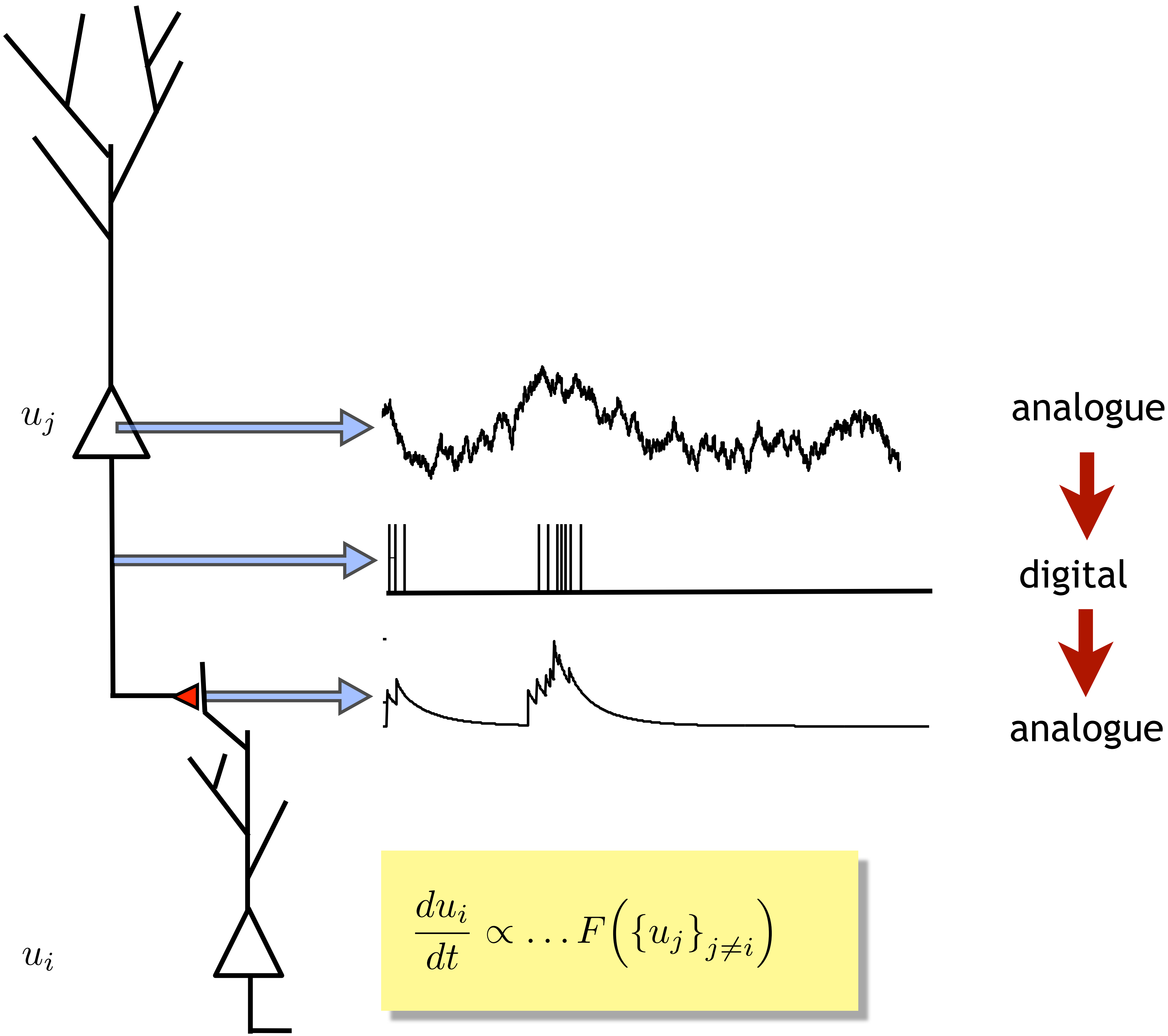
INFORMATION LOSS IN SYNAPTIC TRANSMISSION



INFORMATION LOSS IN SYNAPTIC TRANSMISSION

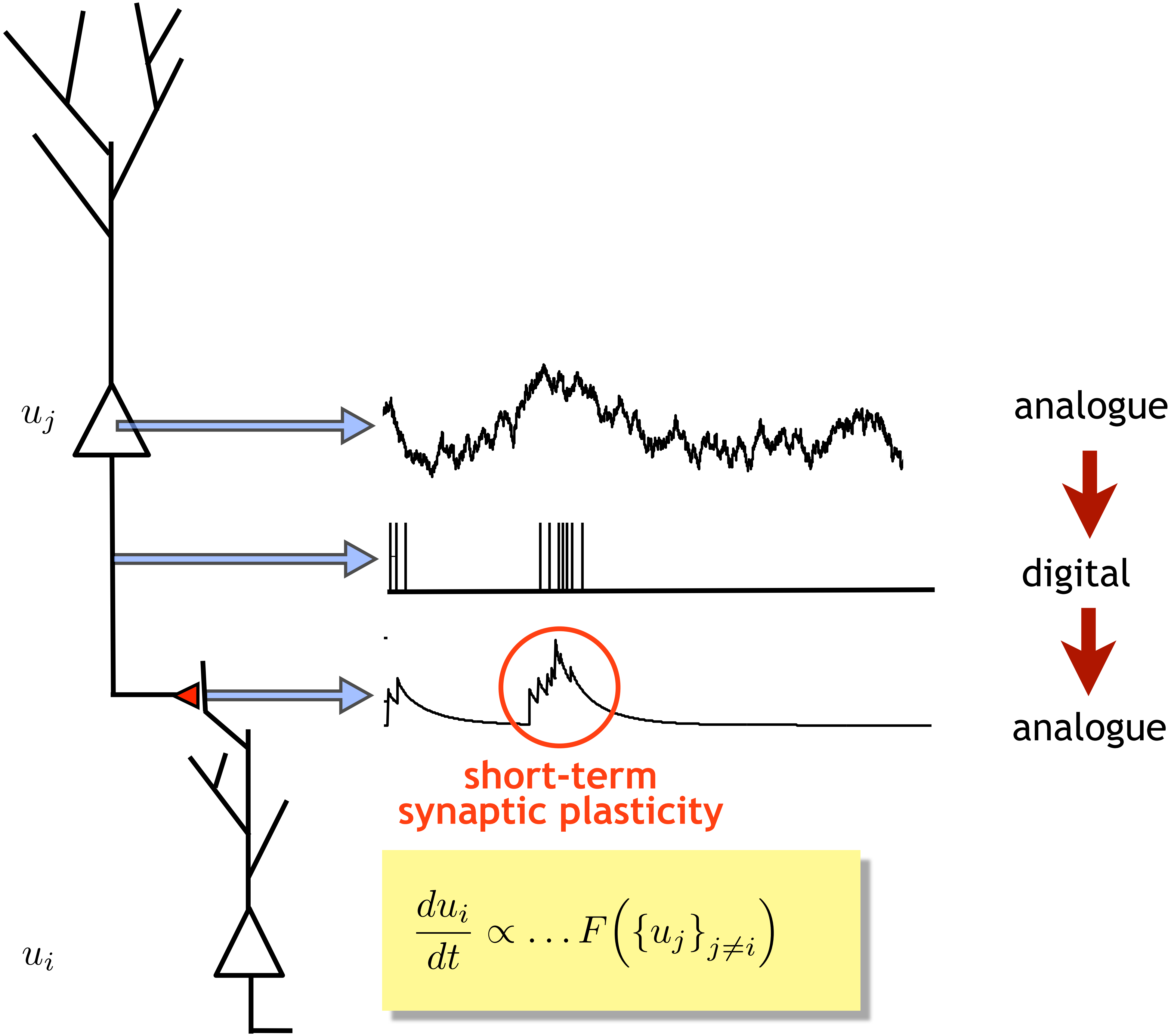


INFORMATION LOSS IN SYNAPTIC TRANSMISSION



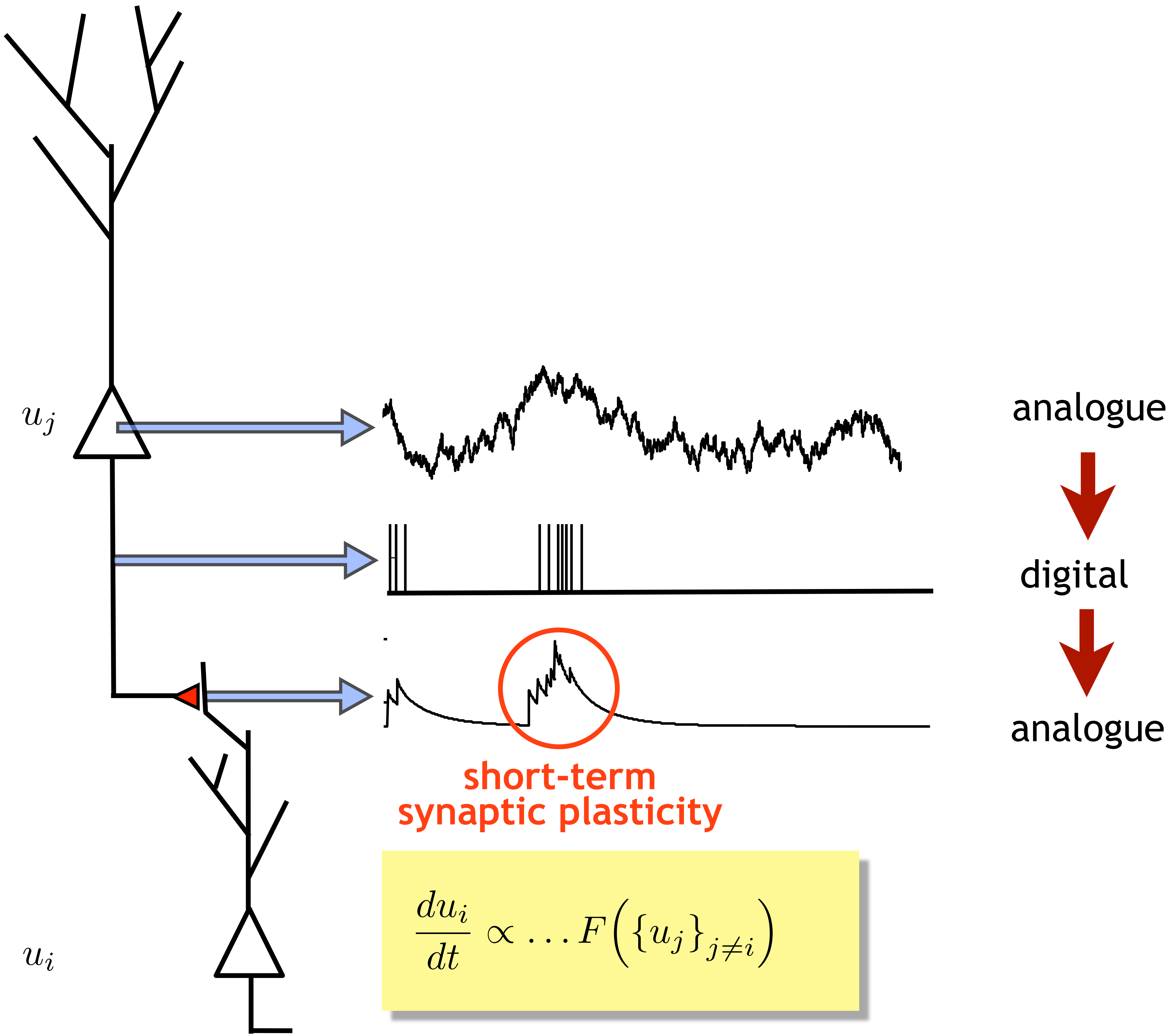
$$\frac{du_i}{dt} \propto \ldots F\Big(\{u_j\}_{j \neq i}\Big)$$

INFORMATION LOSS IN SYNAPTIC TRANSMISSION



$$\frac{du_i}{dt} \propto \ldots F\Big(\{u_j\}_{j \neq i}\Big)$$

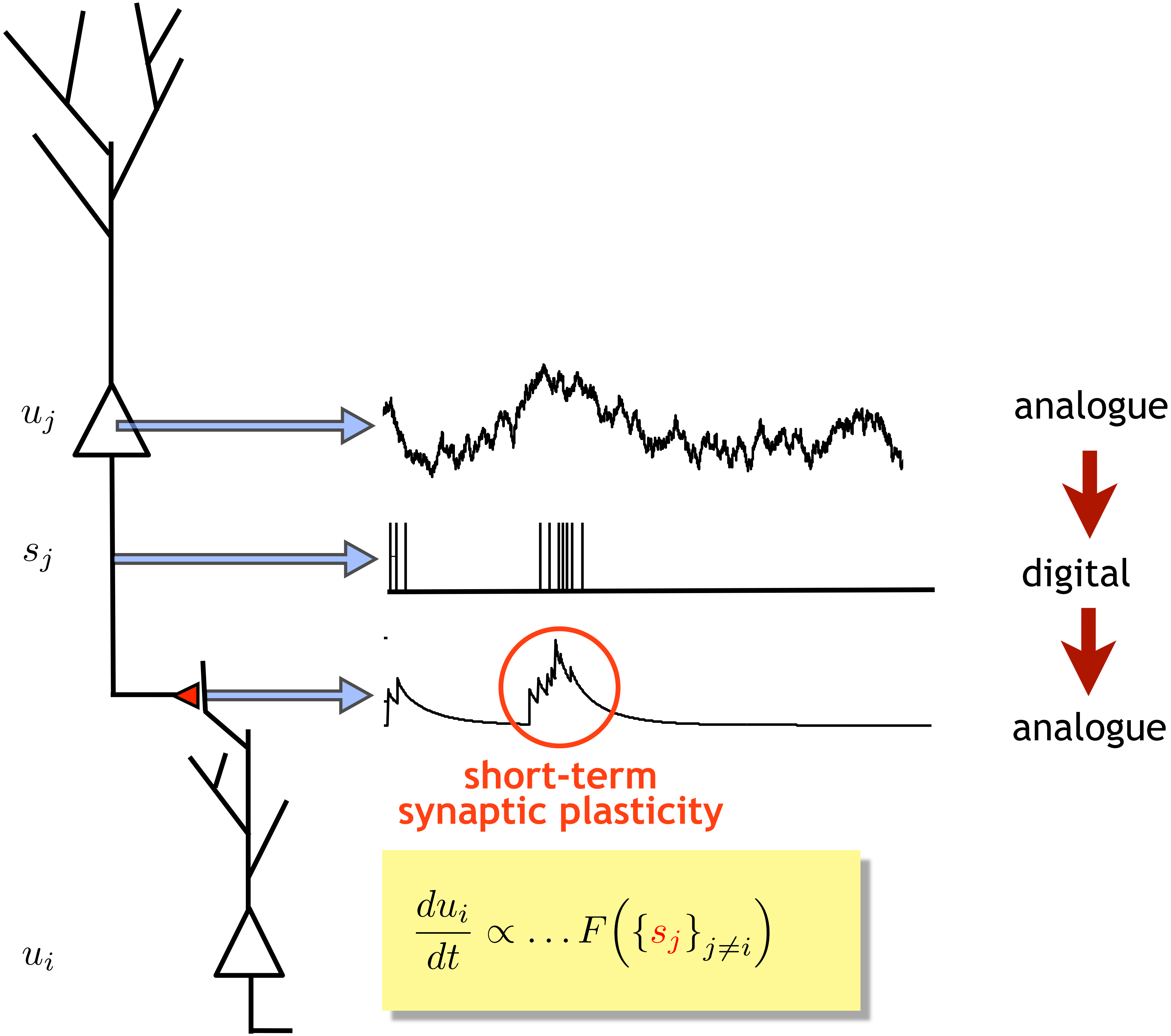
INFORMATION LOSS IN SYNAPTIC TRANSMISSION



loss of information !

$$\frac{du_i}{dt} \propto \ldots F\Big(\{u_j\}_{j \neq i}\Big)$$

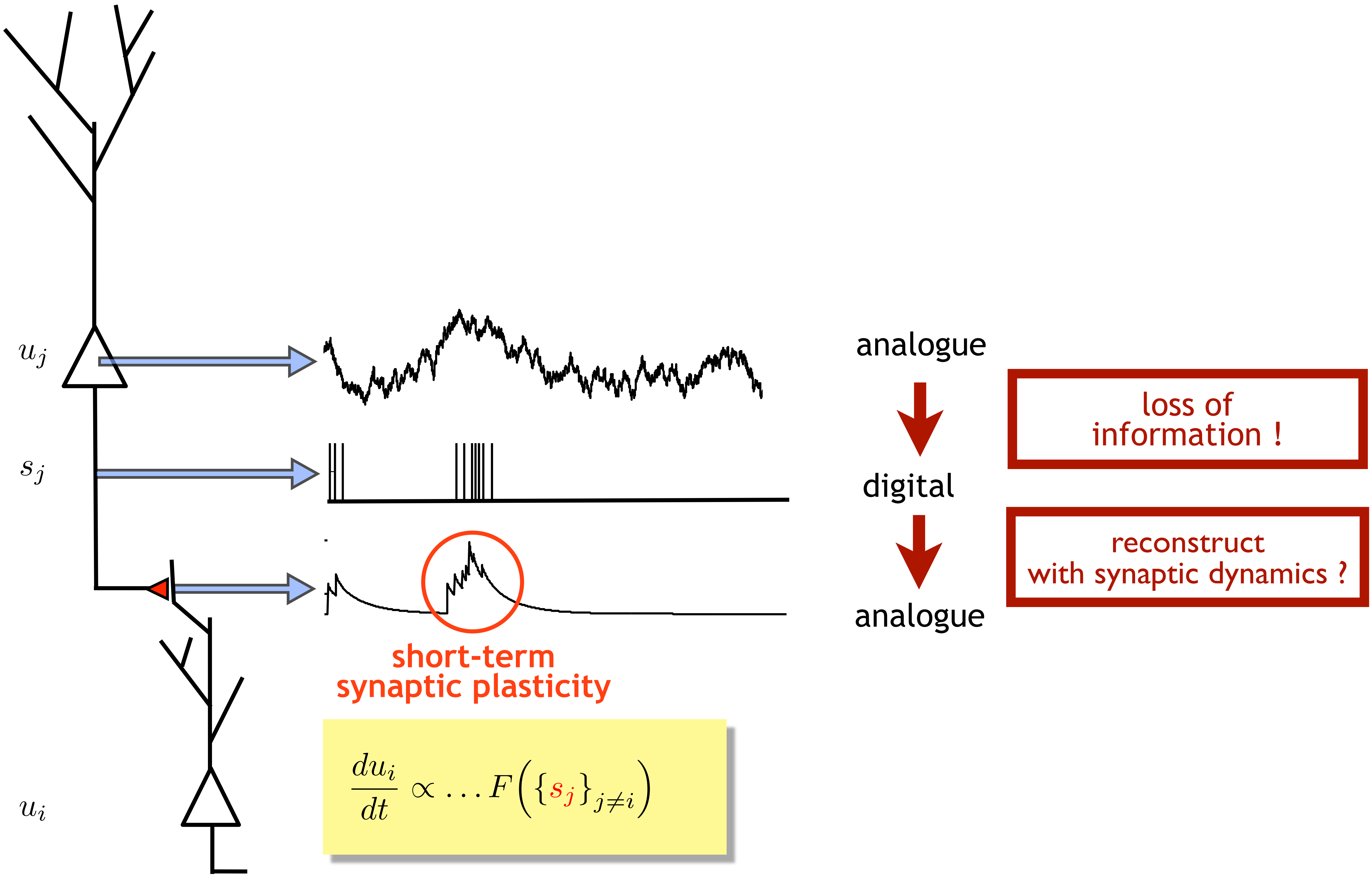
INFORMATION LOSS IN SYNAPTIC TRANSMISSION



$$\frac{du_i}{dt} \propto \ldots F\Big(\{s_j\}_{j \neq i}\Big)$$

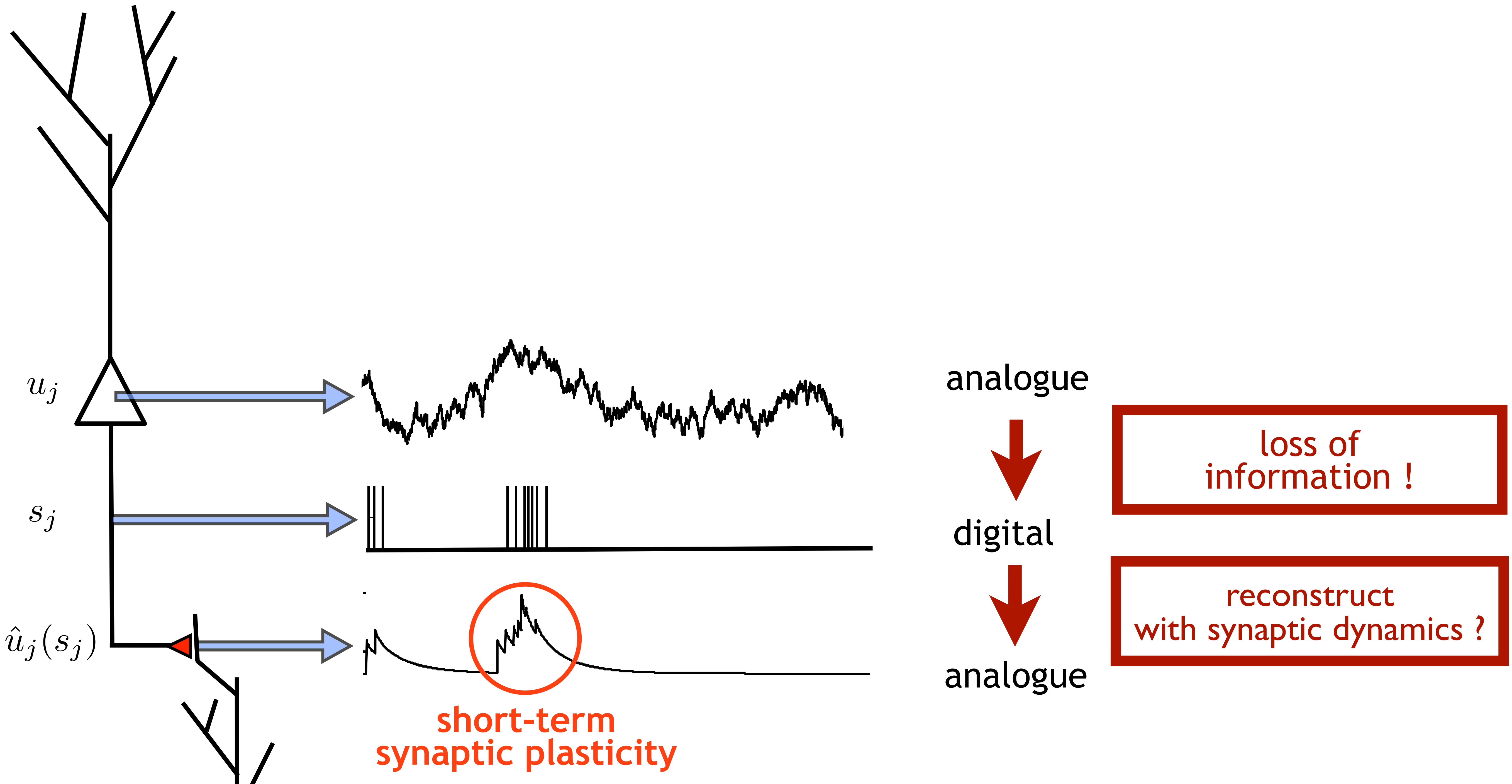
loss of information !

INFORMATION LOSS IN SYNAPTIC TRANSMISSION



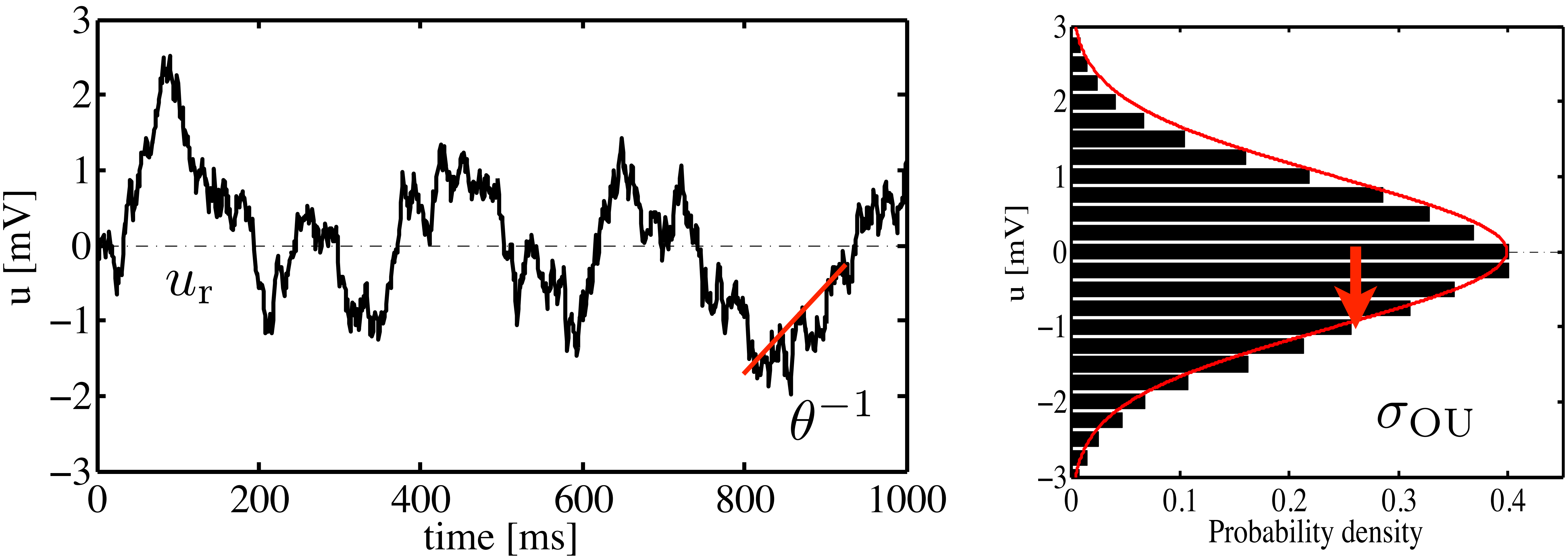
$$\frac{du_i}{dt} \propto \ldots F\Big(\{s_j\}_{j \neq i}\Big)$$

INFORMATION LOSS IN SYNAPTIC TRANSMISSION



$$\frac{du_i}{dt} \propto \ldots F\left(\{\hat{u}_j(s_j)\}_{j \neq i}\right)$$

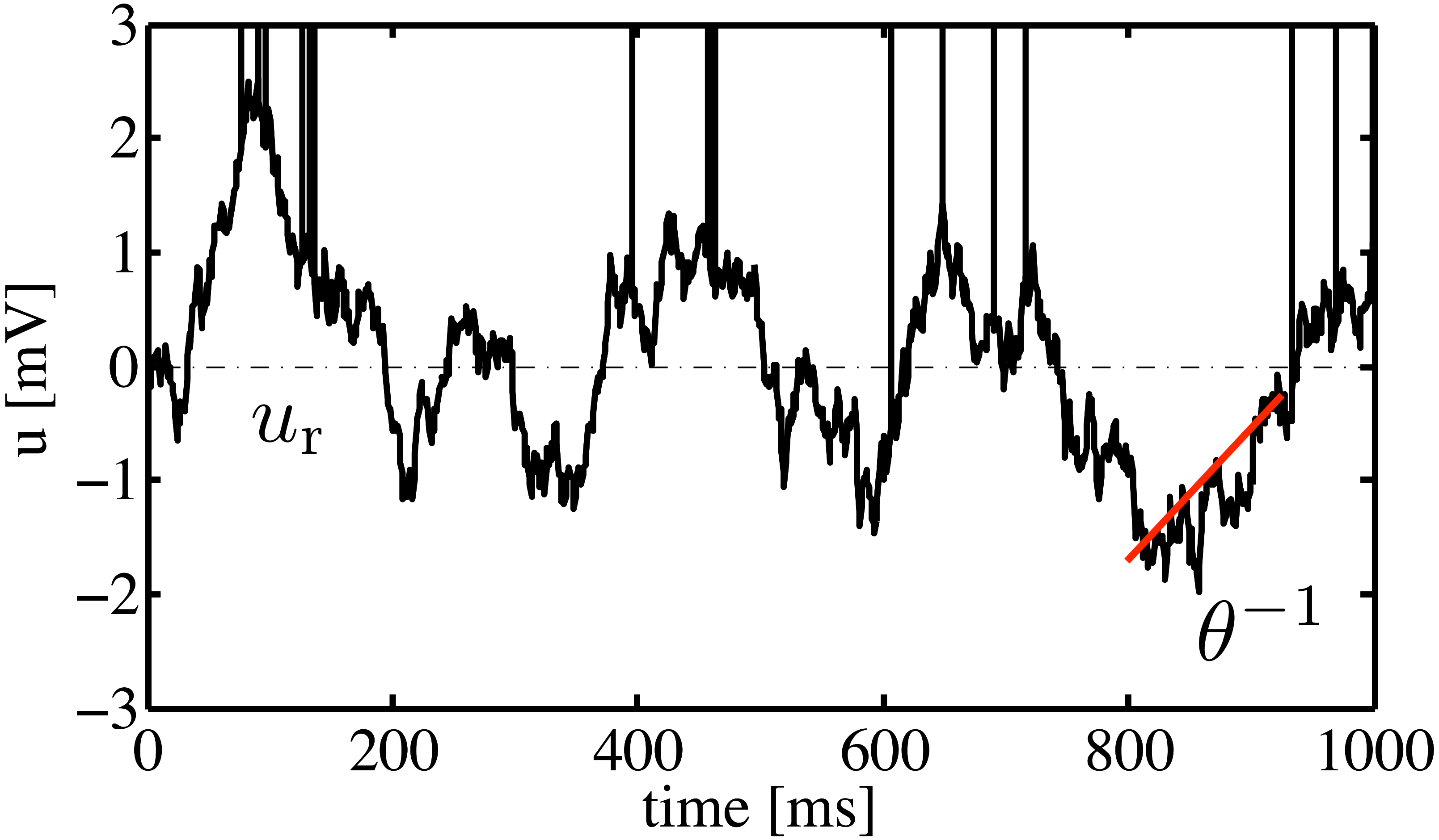
MODEL OF PRESYNAPTIC DYNAMICS

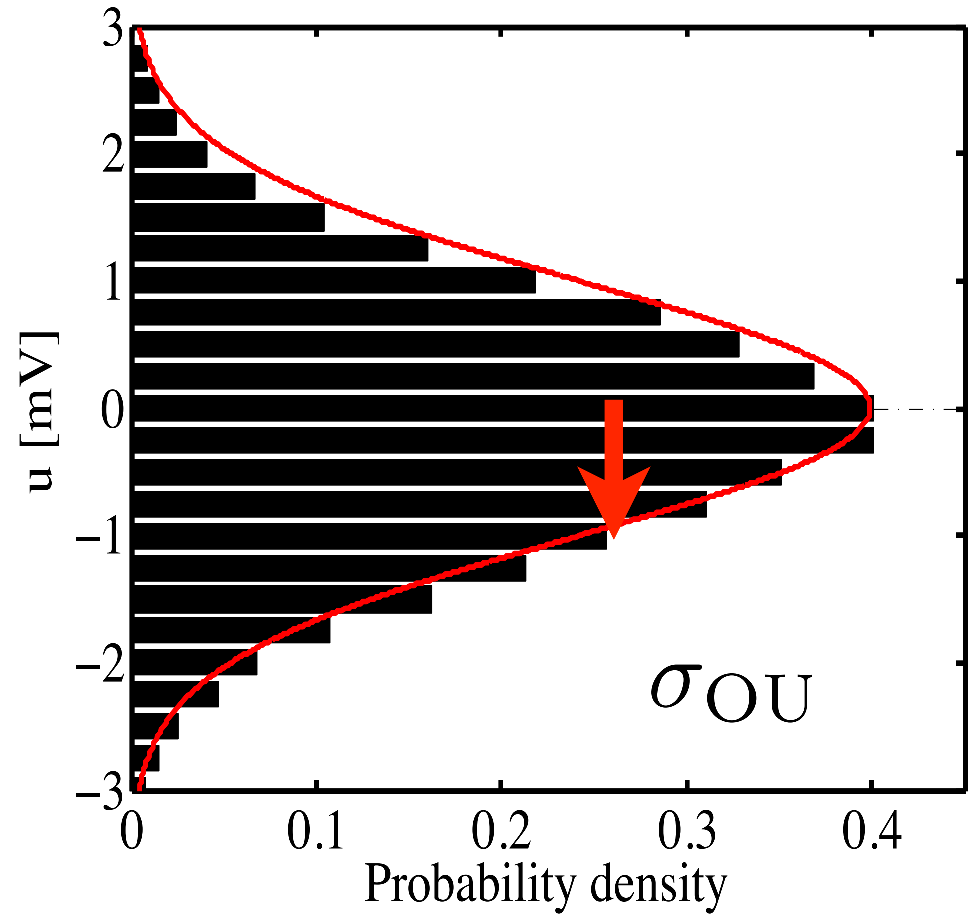


Ornstein-Uhlenbeck process:

$$u_t = u_{t-1} - \theta(u_{t-1} - u_{\mathrm{r}})\Delta t + W_t\sqrt{\Delta t}$$

MODEL OF PRESYNAPTIC DYNAMICS





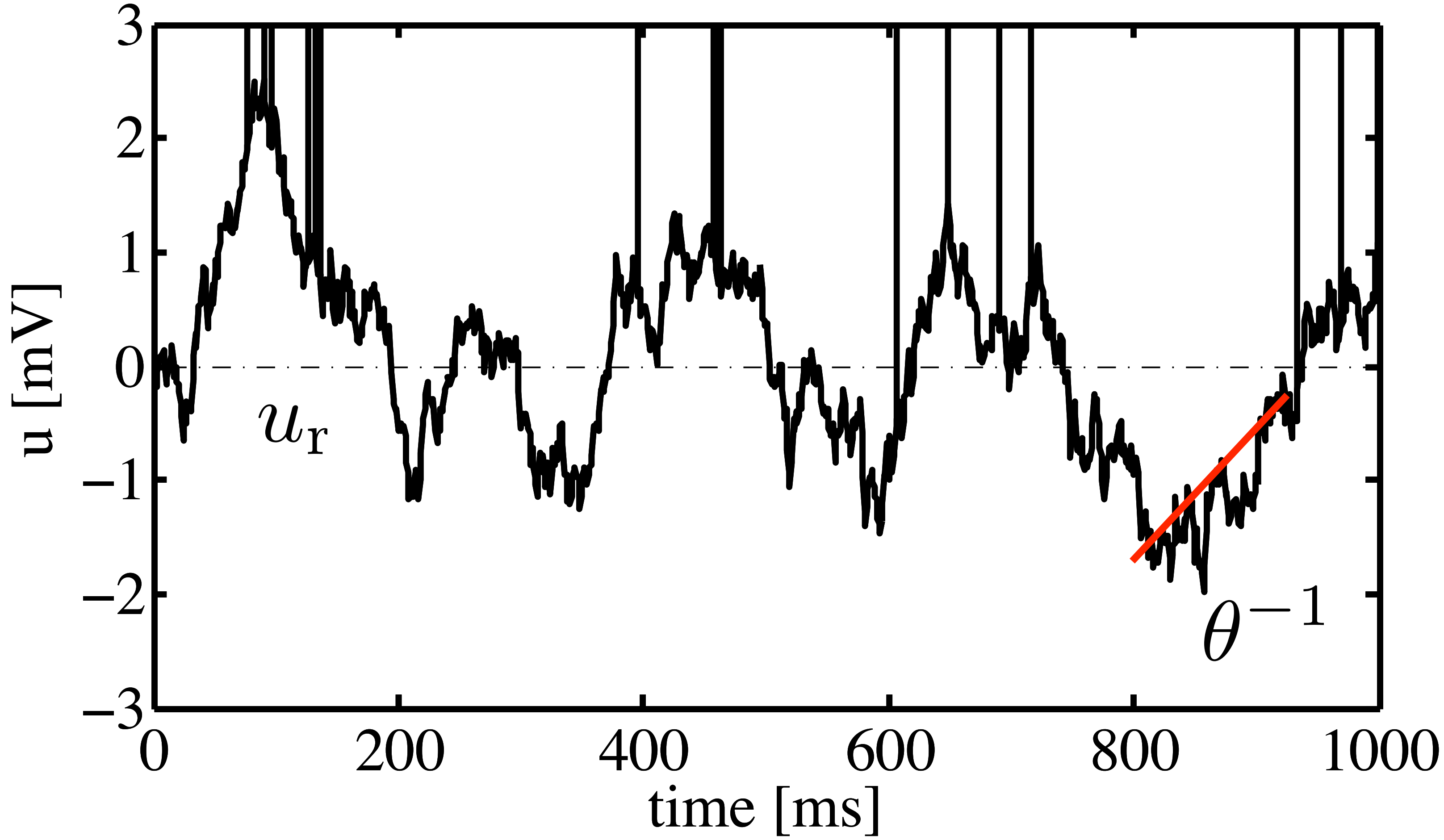
Ornstein-Uhlenbeck process:

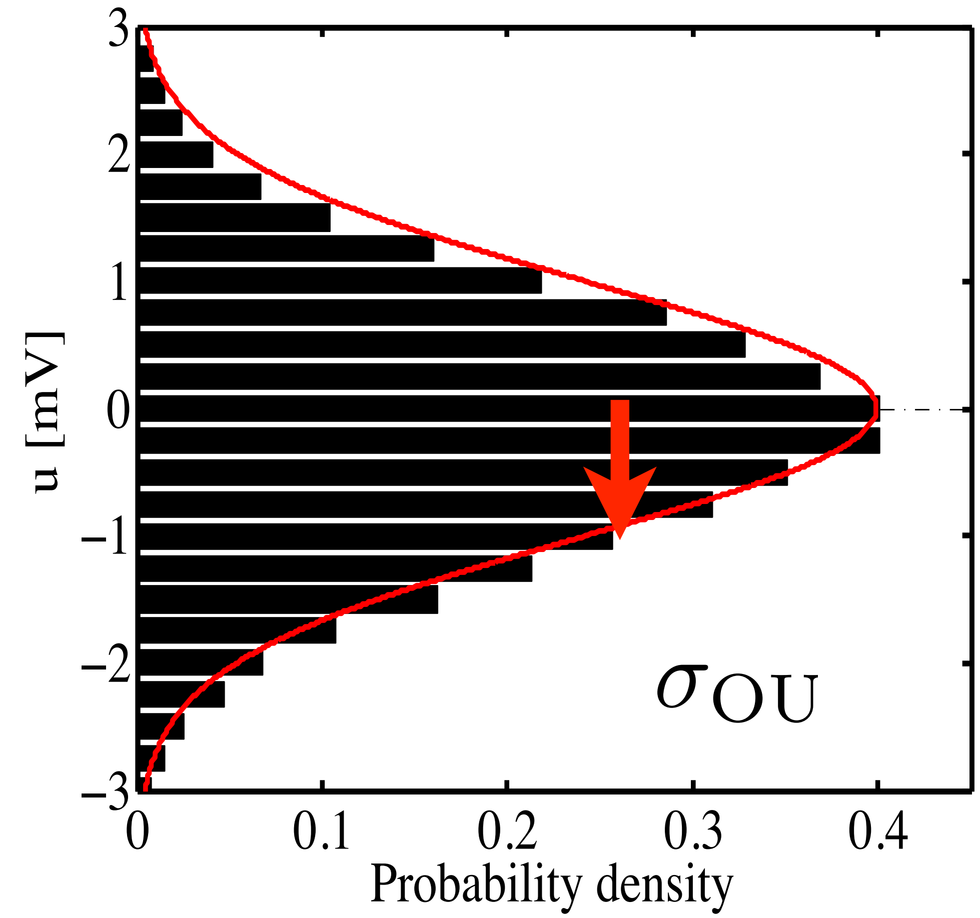
$$u_t = u_{t-1} - \theta(u_{t-1} - u_r)\Delta t + W_t\sqrt{\Delta t}$$

prob. spike:

$$p(s_t = 1|u_t) = g(u_t)\Delta t$$

MODEL OF PRESYNAPTIC DYNAMICS

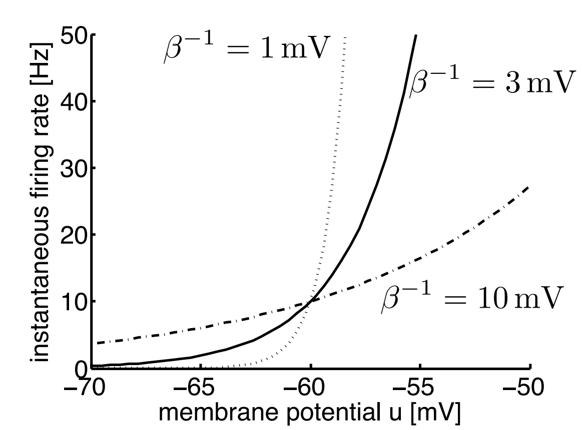




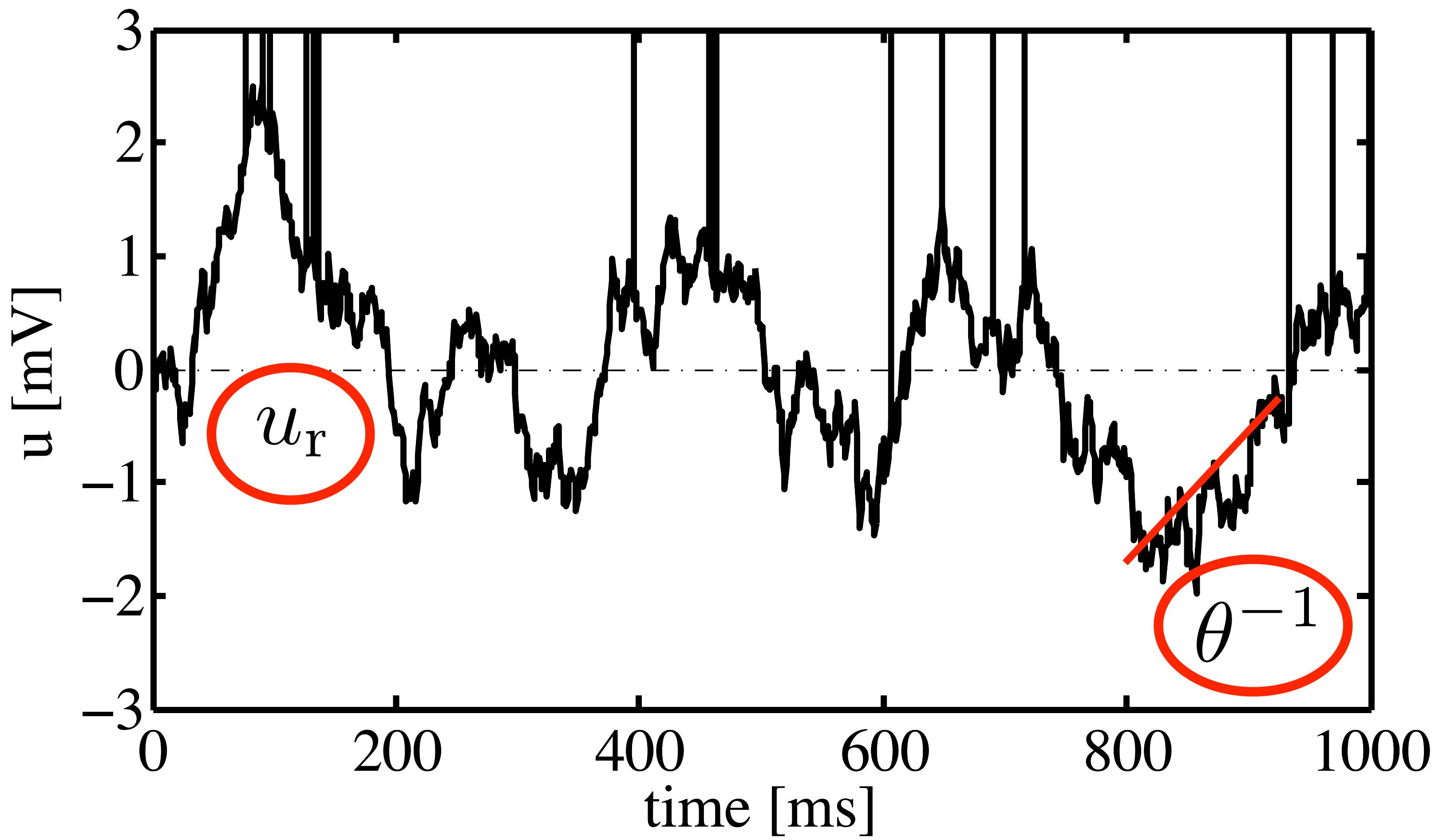
Ornstein-Uhlenbeck process: $$u_t = u_{t-1} - \theta(u_{t-1} - u_r)\Delta t + W_t\sqrt{\Delta t}$$

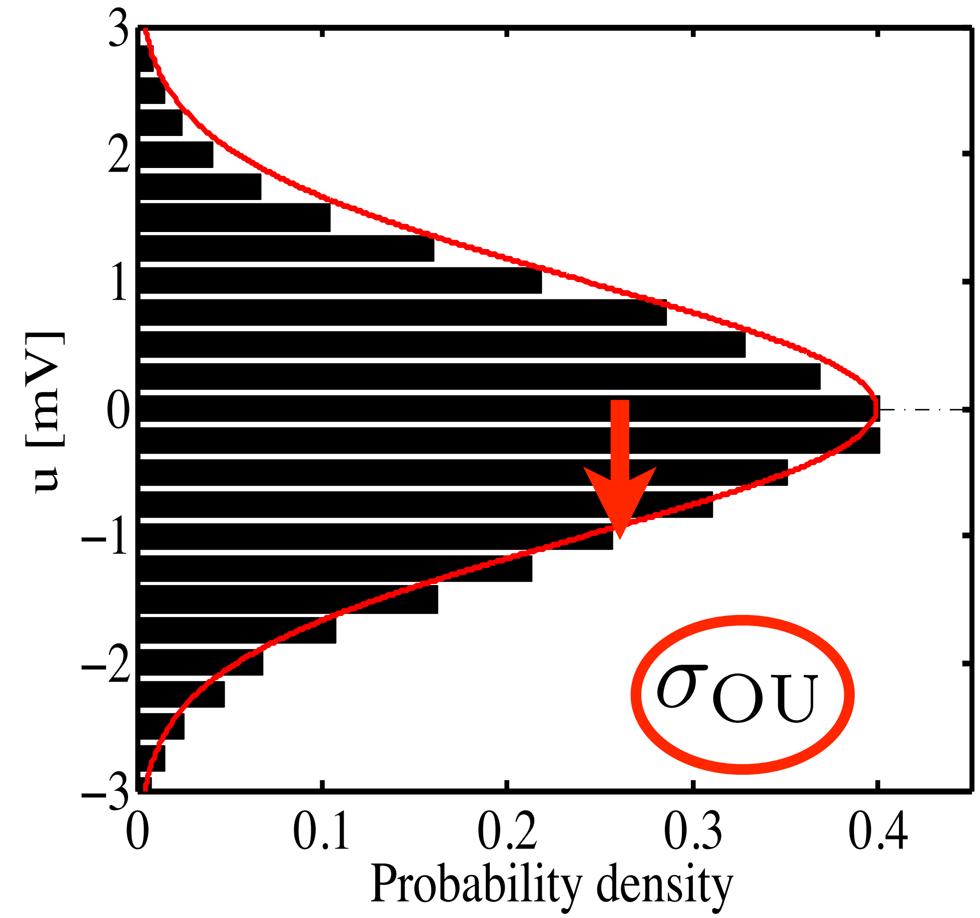
prob. spike: $$p(s_t = 1|u_t) = g(u_t)\Delta t$$

$$g(u) = g_0 \exp(\beta u)$$



MODEL OF PRESYNAPTIC DYNAMICS





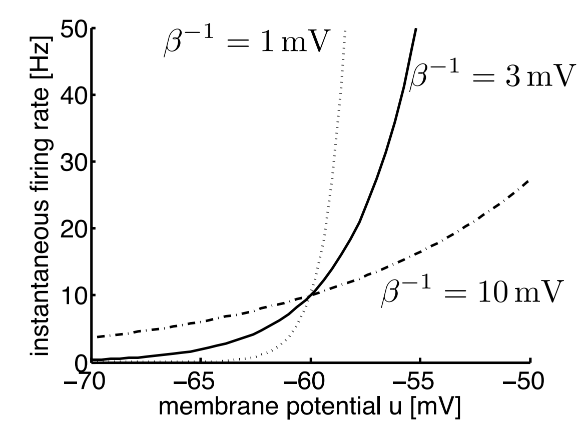
Ornstein-Uhlenbeck process:

$$u_t = u_{t-1} - \theta(u_{t-1} - u_r)\Delta t + W_t\sqrt{\Delta t}$$

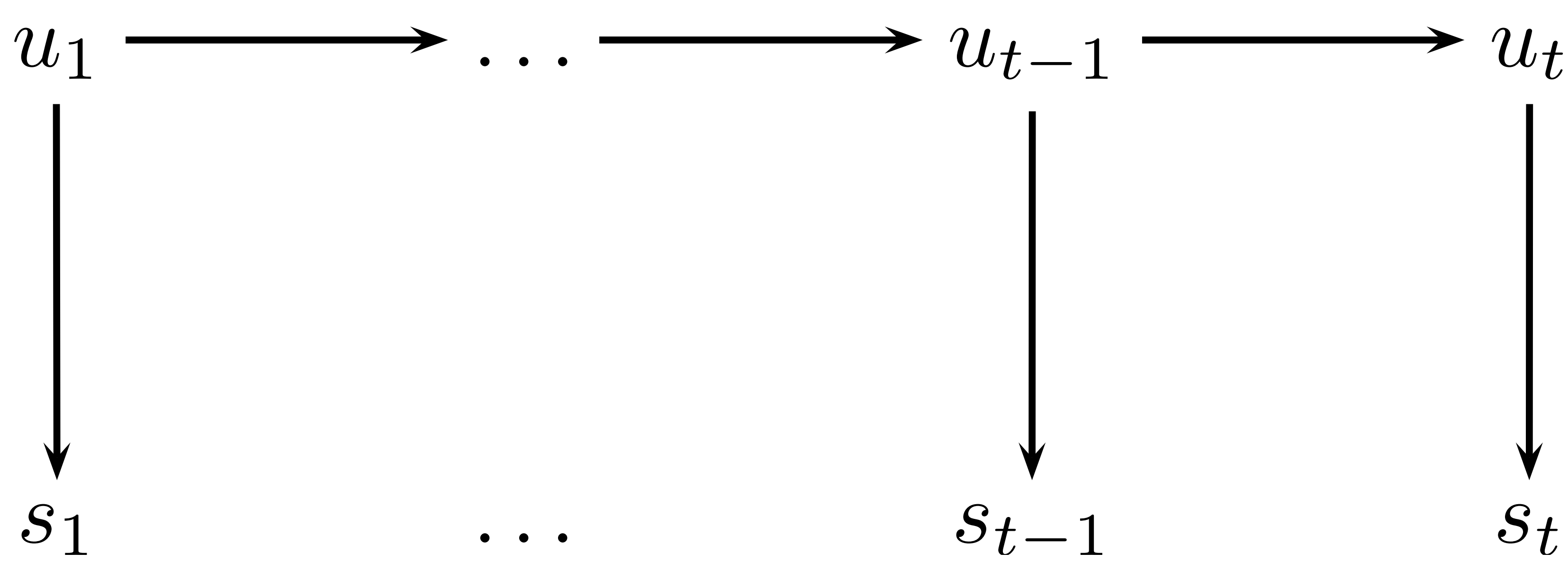
prob. spike:

$$p(s_t = 1 | u_t) = g(u_t)\Delta t$$

$$g(u) = g_0 \exp(\beta u)$$



MODEL OF PRESYNAPTIC DYNAMICS

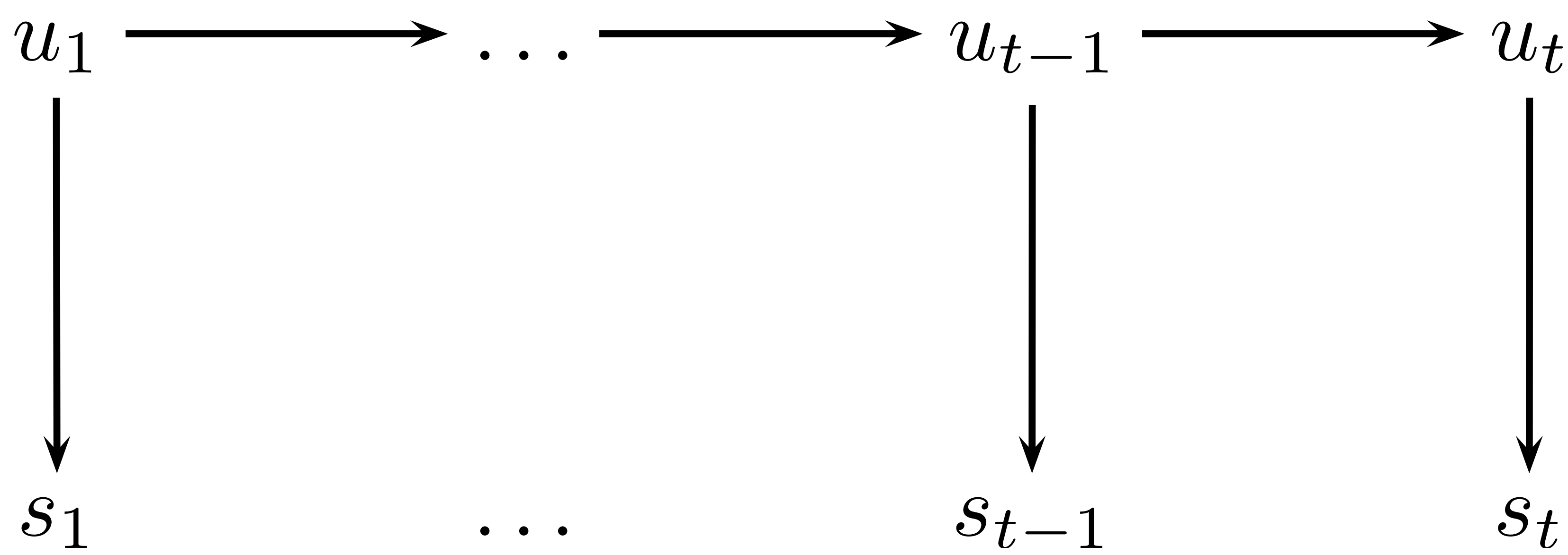


Ornstein-Uhlenbeck process: $$u_t = u_{t-1} - \theta(u_{t-1} - u_{\mathrm{r}})\Delta t + W_t\sqrt{\Delta t}$$

prob. spike: $$p(s_t = 1|u_t) = g(u_t)\Delta t$$

$$g(u) = g_0 \exp(\beta u)$$

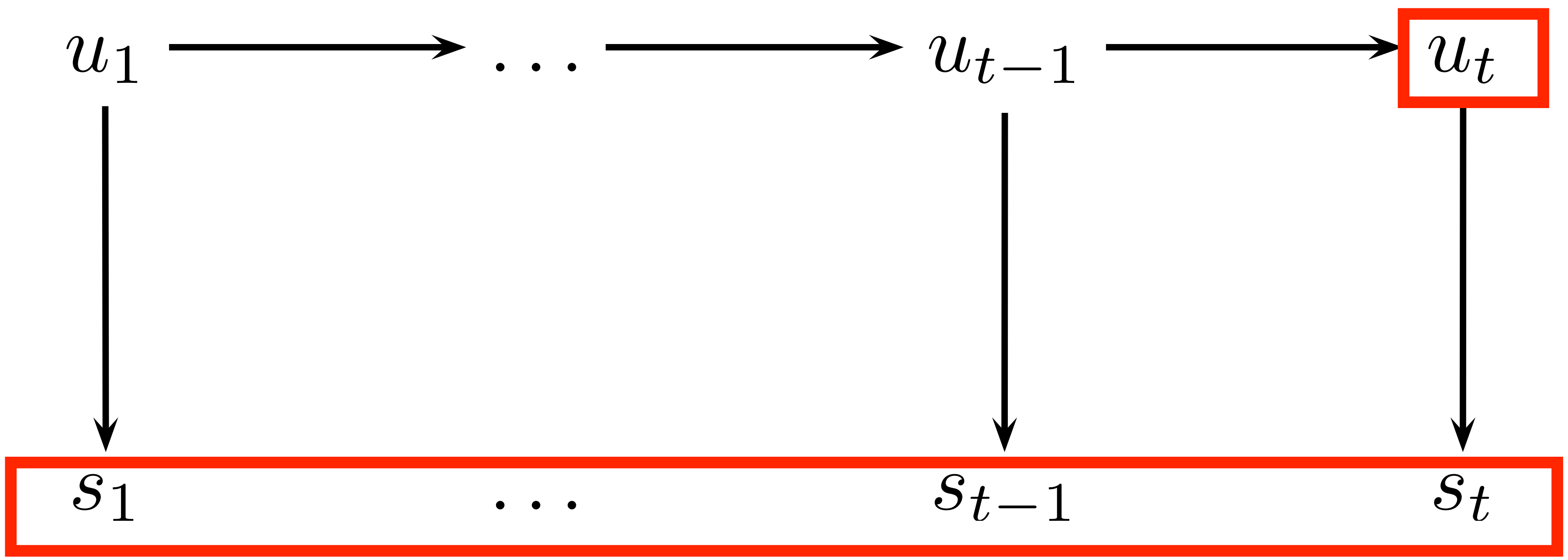
INFERRING THE MEMBRANE POTENTIAL



posterior distribution:

$$p(u_t|s_{1\dots t}) \propto p(s_t|u_t) \int p(u_t|u_{t-1})\, p(u_{t-1}|\, s_{1\dots t-1})\, du_{t-1}$$

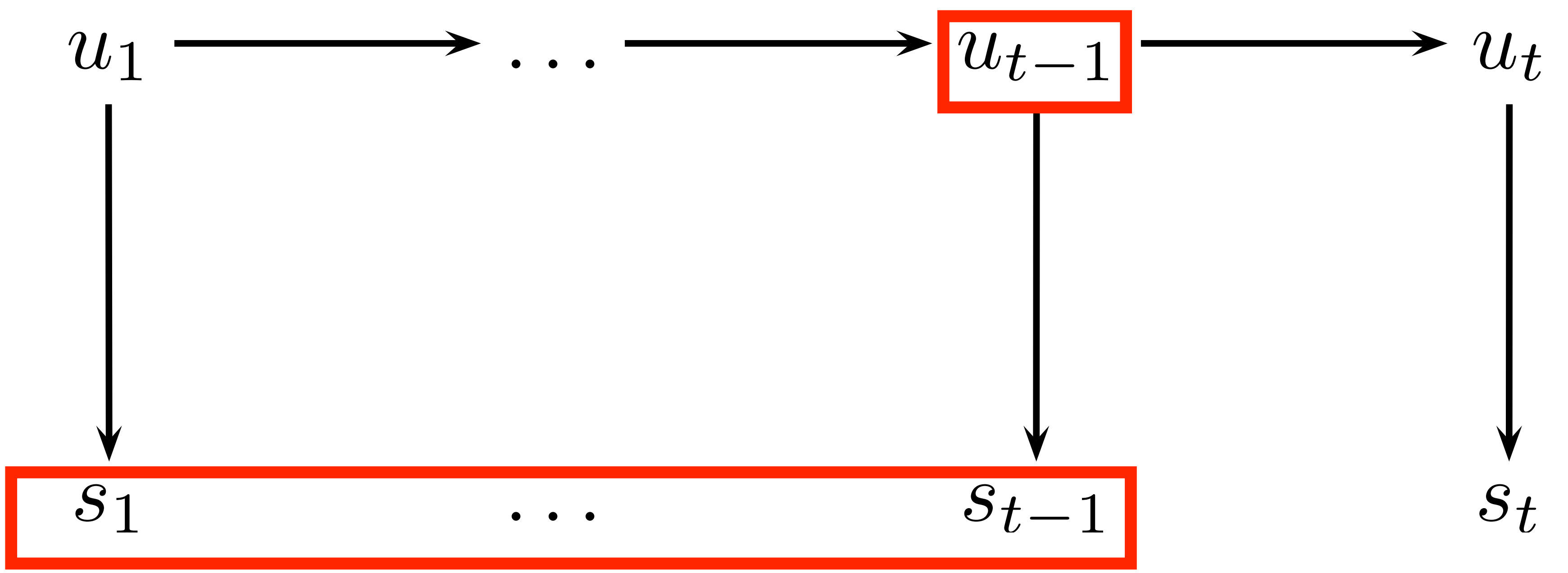
INFERRING THE MEMBRANE POTENTIAL



posterior distribution:

$$p(u_t|s_{1\dots t}) \propto p(s_t|u_t) \int p(u_t|u_{t-1})\, p(u_{t-1}|\, s_{1\dots t-1})\, du_{t-1}$$

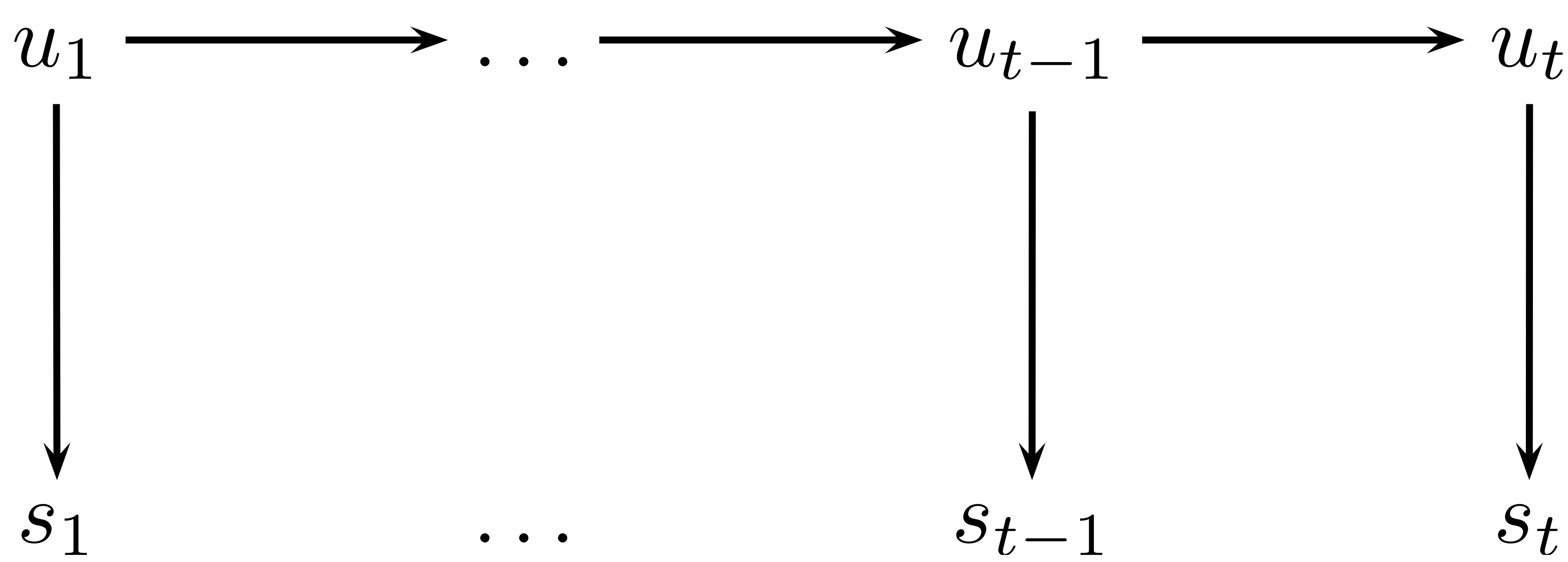
INFERRING THE MEMBRANE POTENTIAL



posterior distribution:

$$p(u_t|s_{1\ldots t}) \propto p(s_t|u_t) \int p(u_t|u_{t-1})\, \underline{p(u_{t-1}|\, s_{1\ldots t-1})}\, du_{t-1}$$

INFERRING THE MEMBRANE POTENTIAL



posterior distribution:

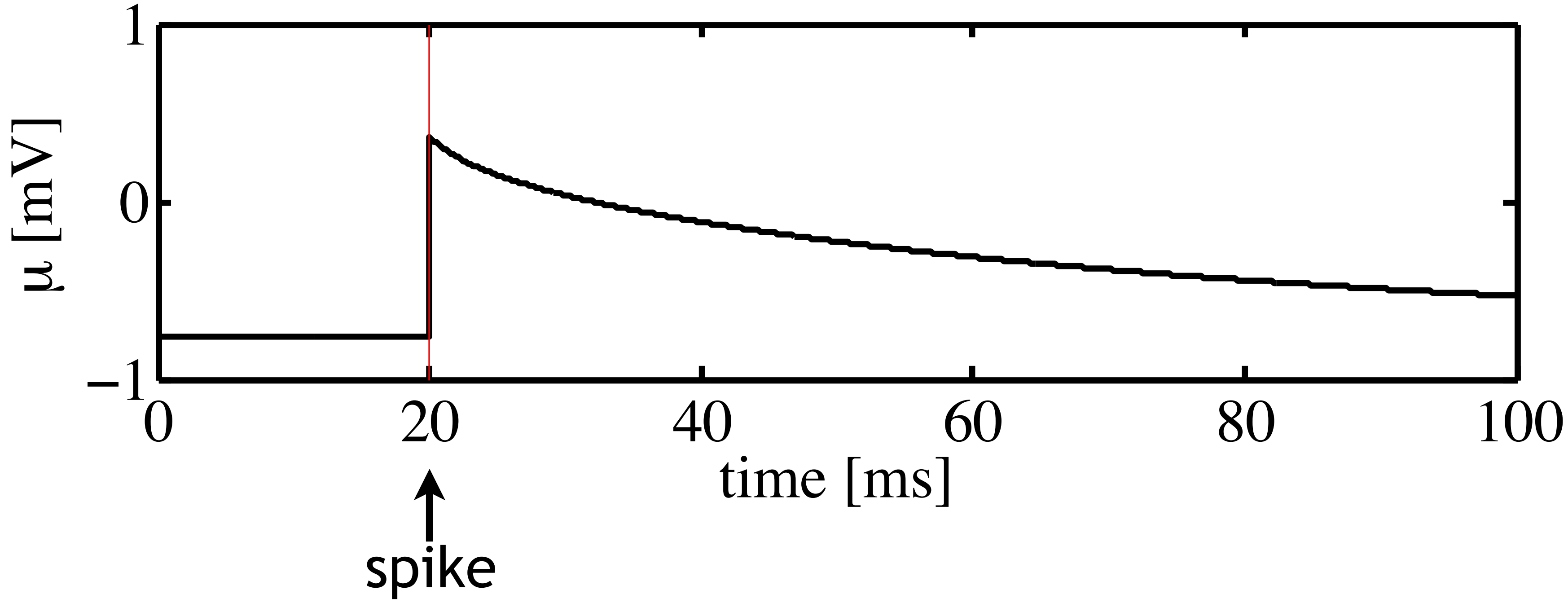
$$\underbrace{p(u_t|s_{1\ldots t})}_{\mathcal{N}(u_t|\mu_t,\sigma_t^2)} \propto p(s_t|u_t) \int p(u_t|u_{t-1})\, \underbrace{p(u_{t-1}|\, s_{1\ldots t-1})}_{\mathcal{N}(u_{t-1}|\mu_{t-1},\sigma_{t-1}^2)}\, du_{t-1}$$

DYNAMICS OF THE OPTIMAL ESTIMATOR

presyn. spike train

mean posterior firing rate

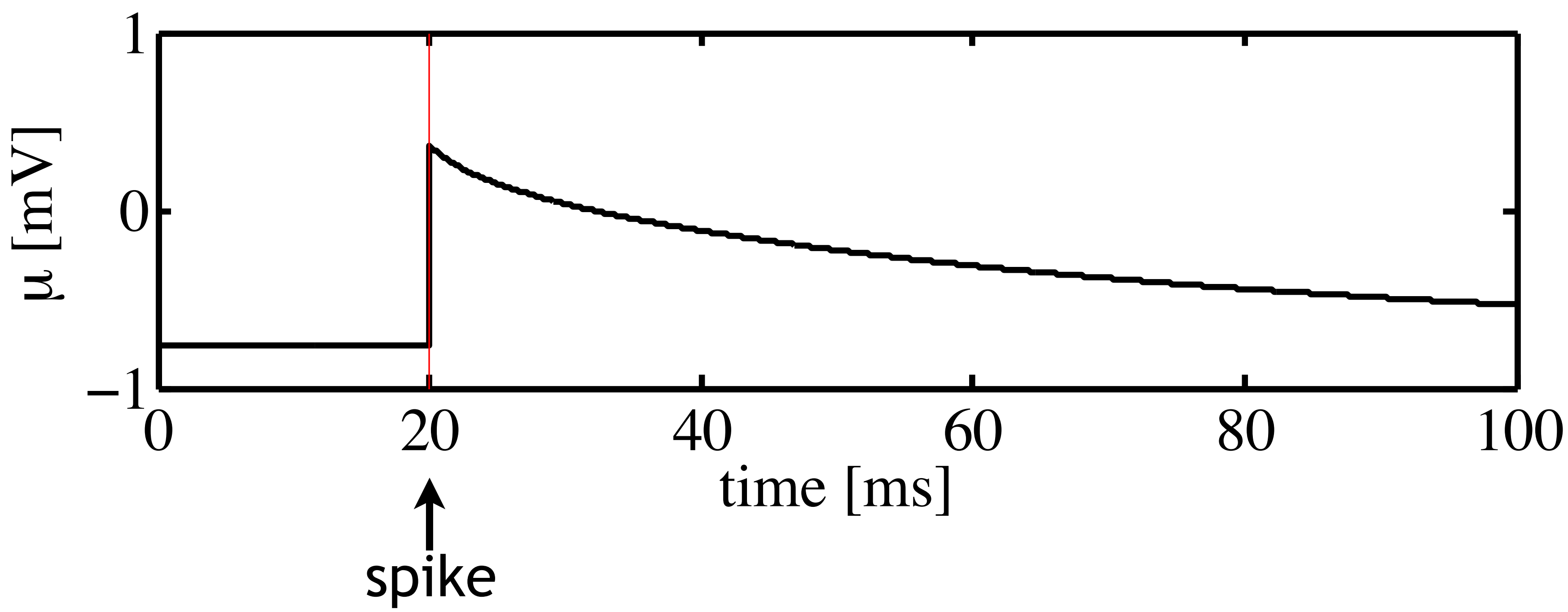
$$\dot{\mu} = -\theta(\mu - u_{\mathrm{r}}) + \beta\sigma^2(S(t) - \gamma(t)) \qquad \gamma(t) = \langle g(u) \rangle_{u|\mu(t),\sigma^2(t)}$$



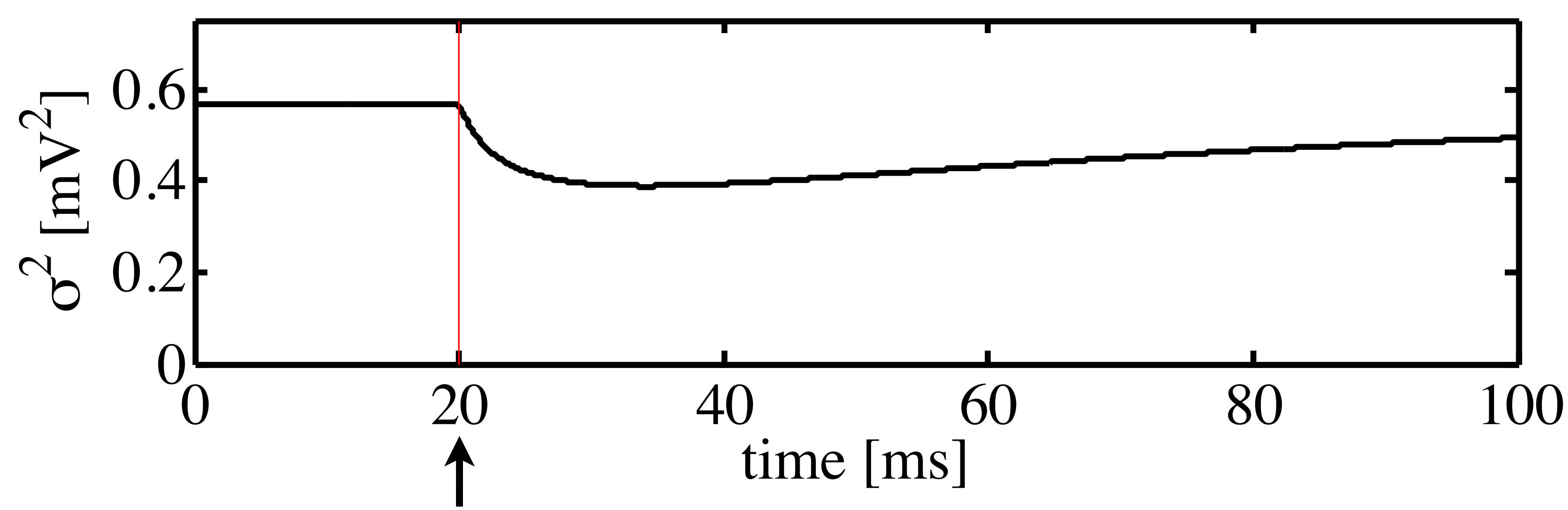
DYNAMICS OF THE OPTIMAL ESTIMATOR

presyn. spike train — mean posterior firing rate

$$\dot{\mu} = -\theta(\mu - u_{\mathrm{r}}) + \beta\sigma^2(S(t) - \gamma(t)) \qquad \gamma(t) = \langle g(u) \rangle_{u|\mu(t),\sigma^2(t)}$$



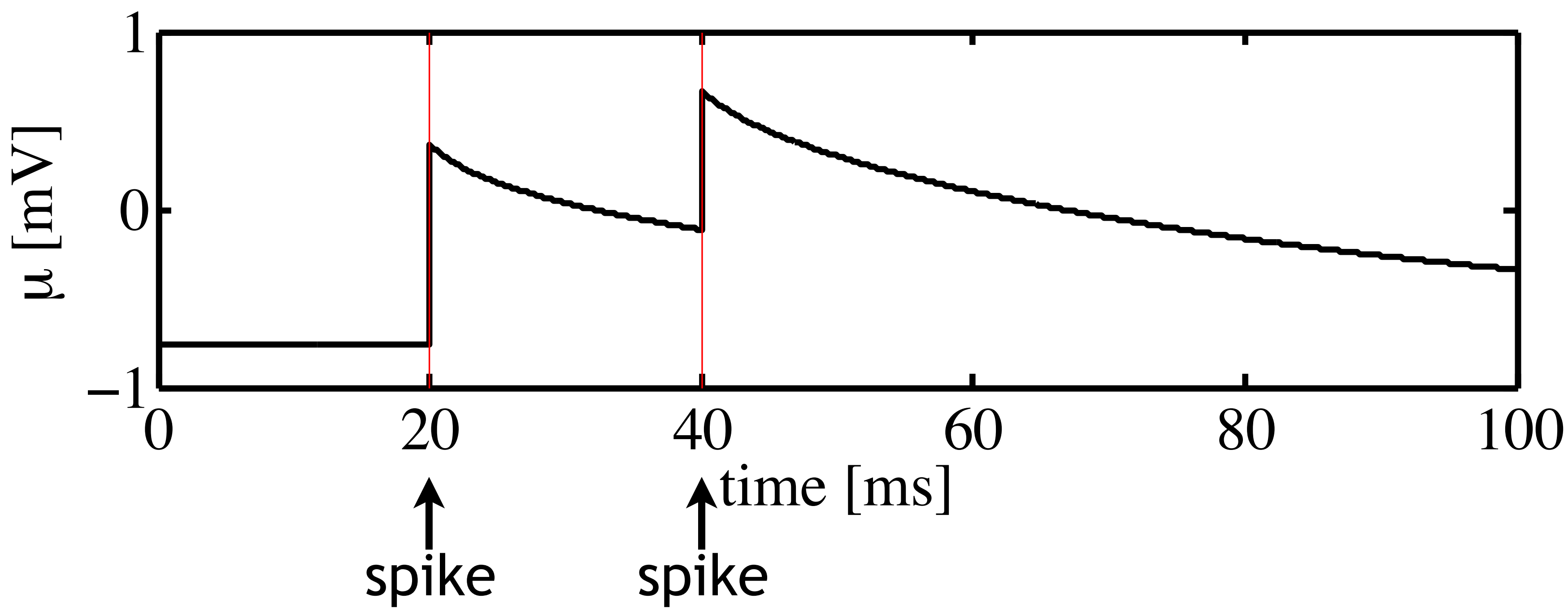
$$\dot{\sigma}^2 = -2\theta(\sigma^2 - \sigma^2_{\mathrm{OU}}) - \gamma(t)\beta^2\sigma^4$$



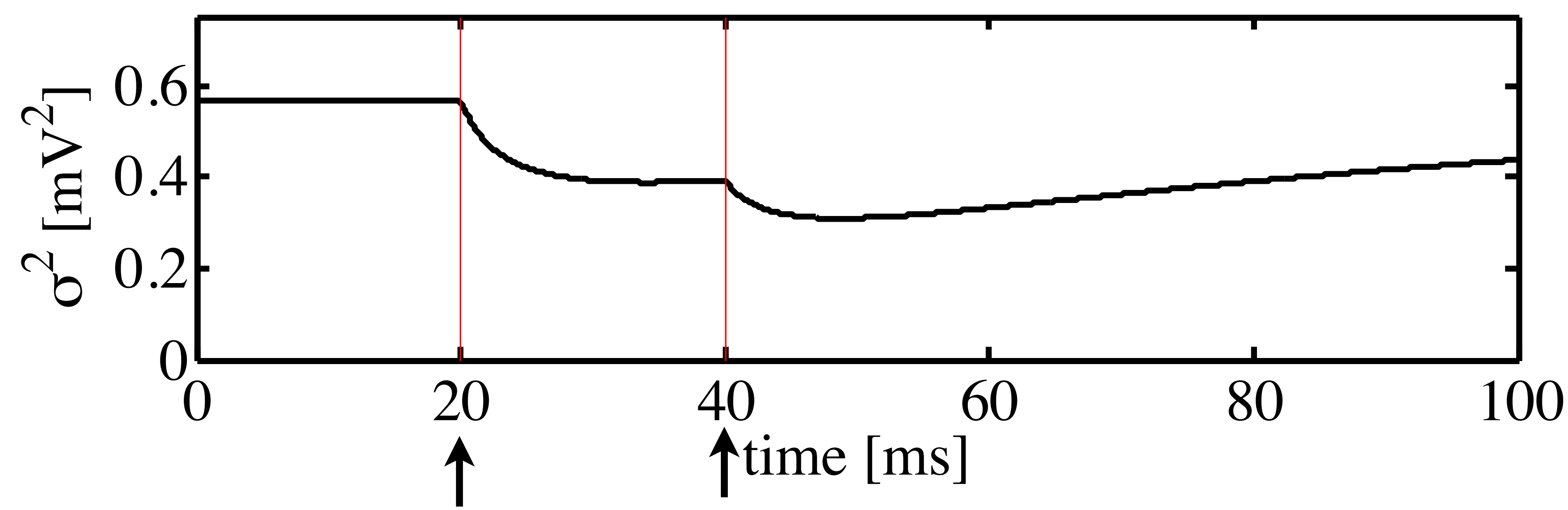
DYNAMICS OF THE OPTIMAL ESTIMATOR

presyn. spike train → $S(t)$; mean posterior firing rate → $\gamma(t)$

$$\dot{\mu} = -\theta(\mu - u_\mathrm{r}) + \beta\sigma^2(S(t) - \gamma(t)) \qquad \gamma(t) = \langle g(u) \rangle_{u|\mu(t),\sigma^2(t)}$$



$$\dot{\sigma}^2 = -2\theta(\sigma^2 - \sigma^2_\mathrm{OU}) - \gamma(t)\beta^2\sigma^4$$



RELATION TO SHORT-TERM PLASTICITY

"biophysical" model of dynamic synapse

postsynaptic membrane potential

$$\frac{dv}{dt} = \frac{v_0 - v}{\tau_{\mathrm{m}}} + J\,x\,S(t)$$

synaptic resource

$$\frac{dx}{dt} = \frac{1 - x}{\tau_{\mathrm{D}}} - Y\,x\,S(t)$$

Tsodyks et al., 1998

RELATION TO SHORT-TERM PLASTICITY

“biophysical” model of dynamic synapse

postsynaptic membrane potential

$$\frac{dv}{dt} = \frac{v_0 - v}{\tau_{\mathrm{m}}} + J\,x\,S(t)$$

synaptic resource

$$\frac{dx}{dt} = \frac{1 - x}{\tau_{\mathrm{D}}} - Y\,x\,S(t)$$

Tsodyks et al., 1998

optimal estimator
(in the limit)

mean estimate

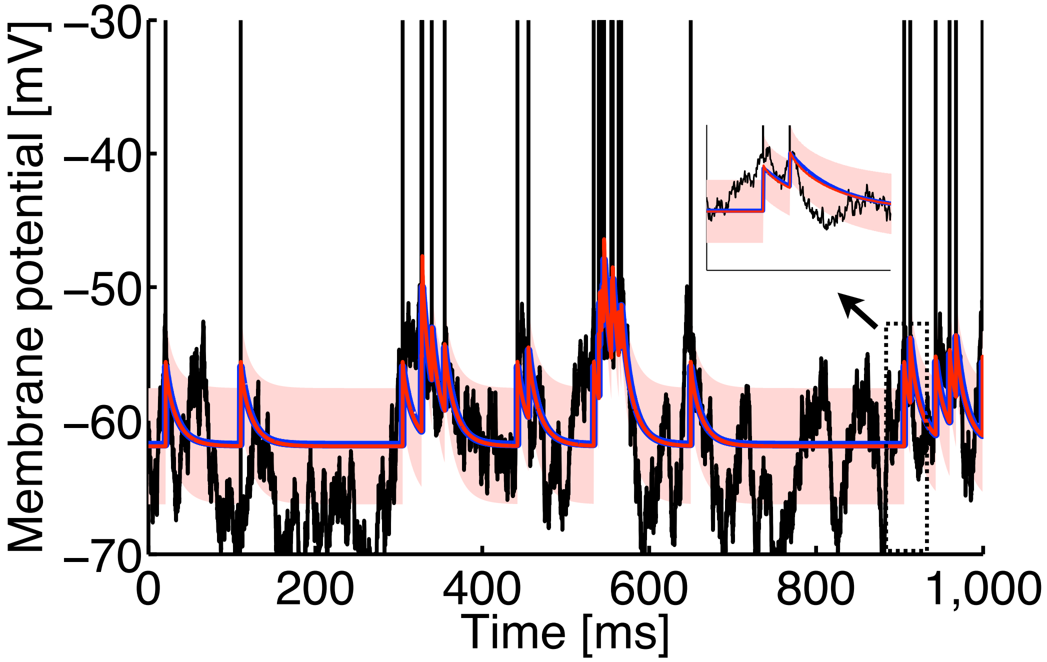
$$\frac{d\hat{u}}{dt} = \frac{\hat{u}_0 - \hat{u}}{\tau_{\mathrm{m}}} + J\,\sigma_{\mathrm{u}}^2\,S(t)$$

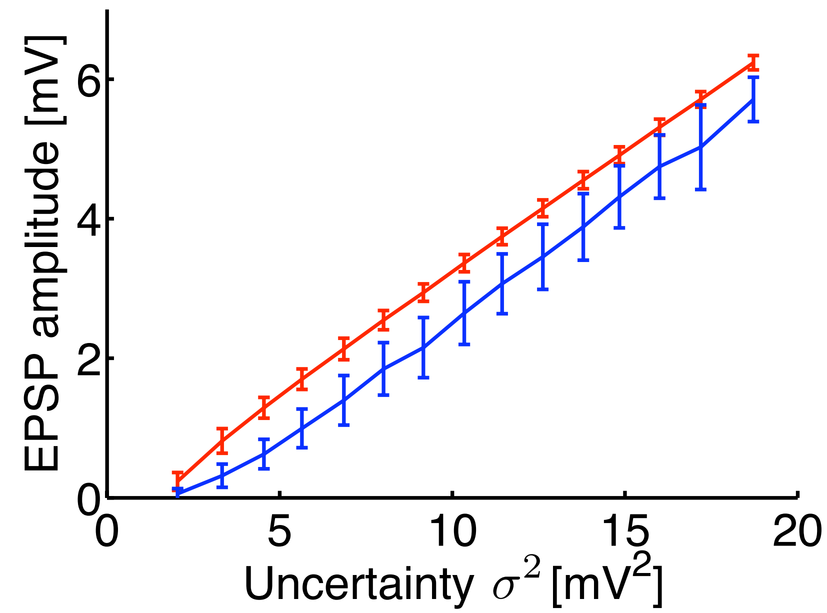
estimator uncertainty

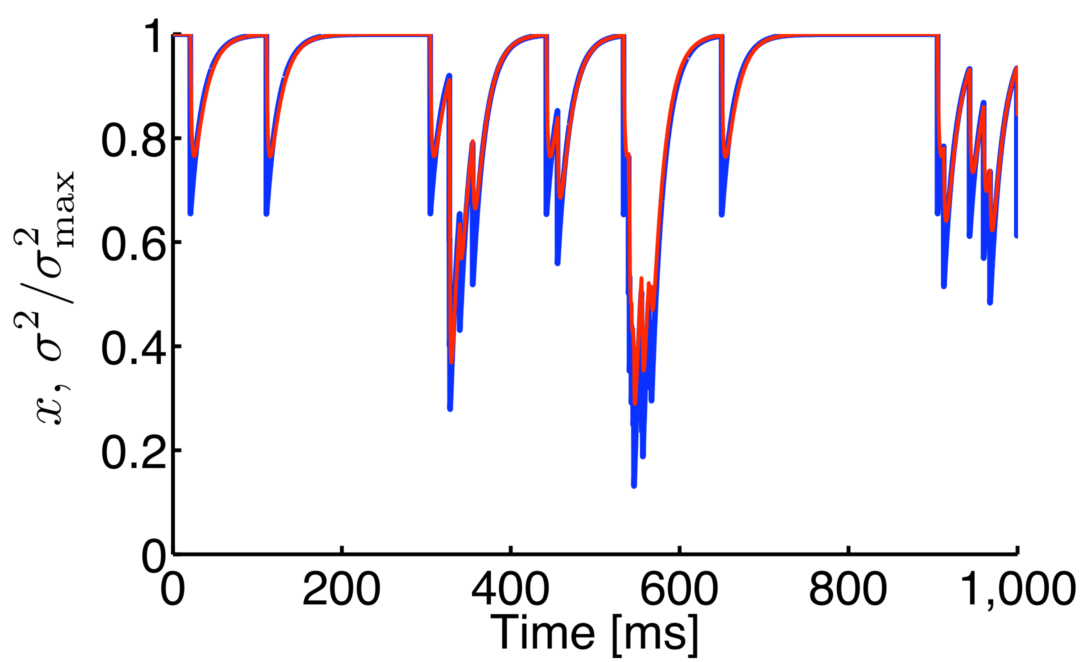
$$\frac{d\sigma_{\mathrm{u}}^2}{dt} = \frac{1 - \sigma_{\mathrm{u}}^2}{\tau_{\mathrm{D}}} - Y\,\sigma_{\mathrm{u}}^2\,S(t)$$

Pfister et al., NIPS 2009
Pfister et al., Nat Neurosci 2010

DEPRESSING SYNAPSES AS NEAR-OPTIMAL ESTIMATORS







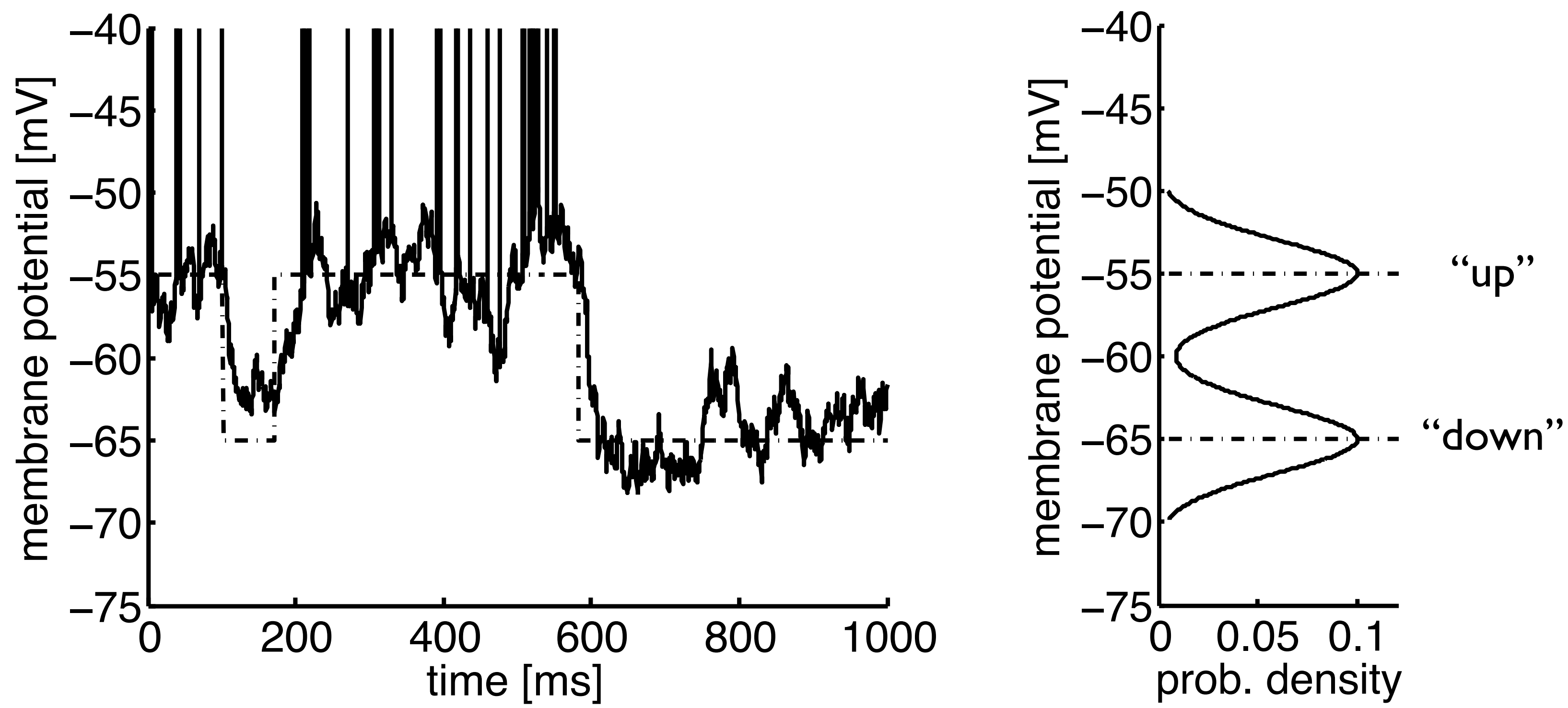
— presynaptic cell

— optimal estimator

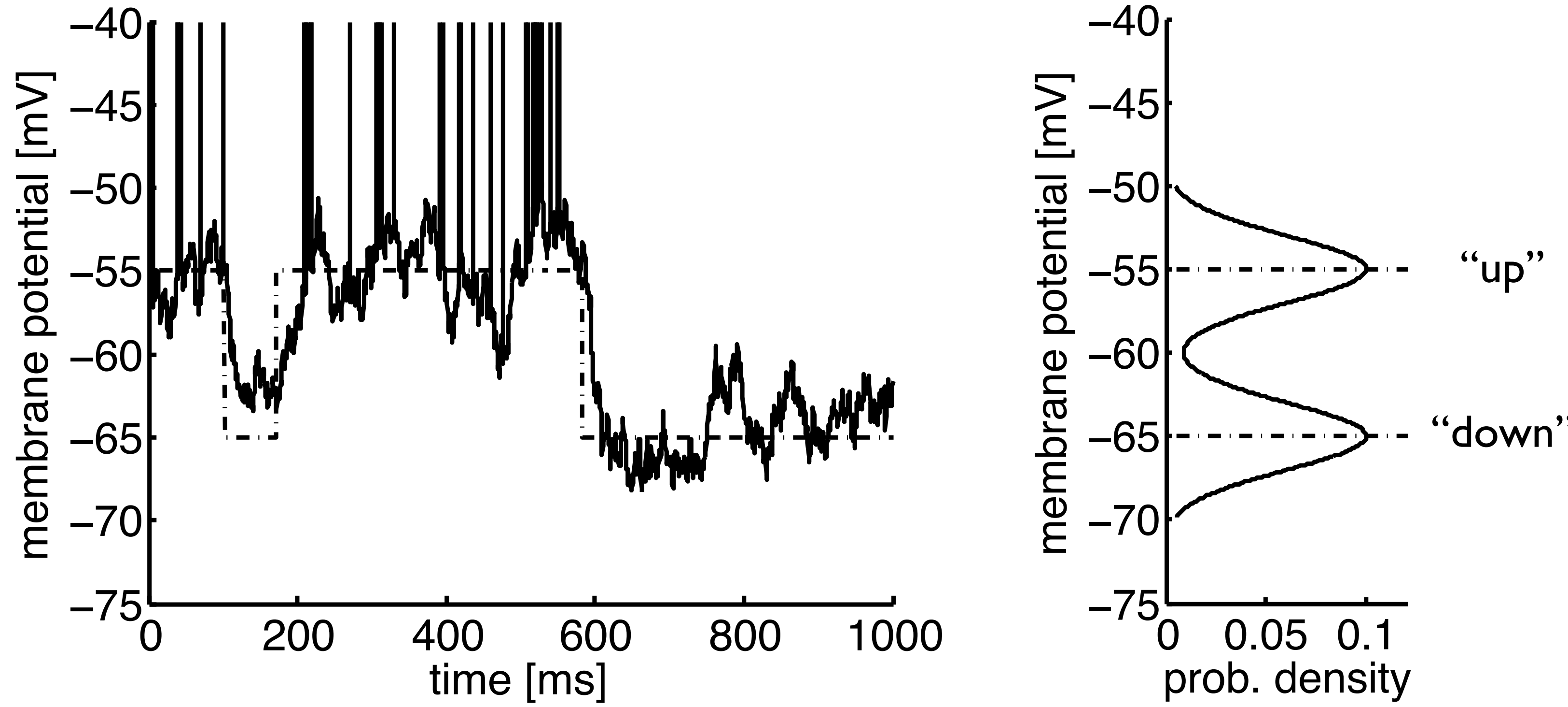
— synapse with STD

Pfister et al, Nat Neurosci 2010

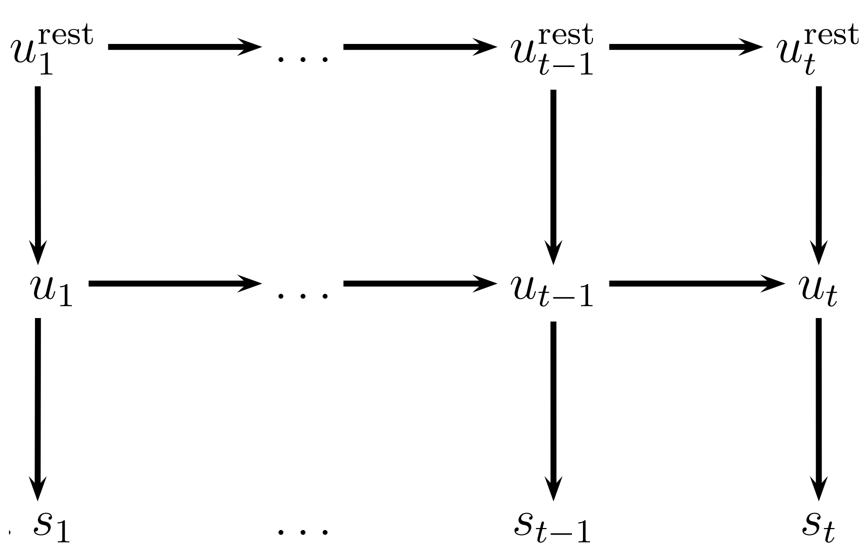
AN EXTENSION: "UP" AND "DOWN"



AN EXTENSION: “UP” AND “DOWN”



switching OU process



$$\mathrm{P}\left(u_t^{\text{rest}} | u_{t-1}^{\text{rest}}\right) = \begin{cases} 1-\eta^{-}\Delta t & \text{if } u_t^{\text{rest}} = u^{+} \text{ and } u_{t-1}^{\text{rest}} = u^{+} \\ \eta^{-}\Delta t & \text{if } u_t^{\text{rest}} = u^{-} \text{ and } u_{t-1}^{\text{rest}} = u^{+} \\ 1-\eta^{+}\Delta t & \text{if } u_t^{\text{rest}} = u^{-} \text{ and } u_{t-1}^{\text{rest}} = u^{-} \\ \eta^{+}\Delta t & \text{if } u_t^{\text{rest}} = u^{+} \text{ and } u_{t-1}^{\text{rest}} = u^{-} \end{cases}$$

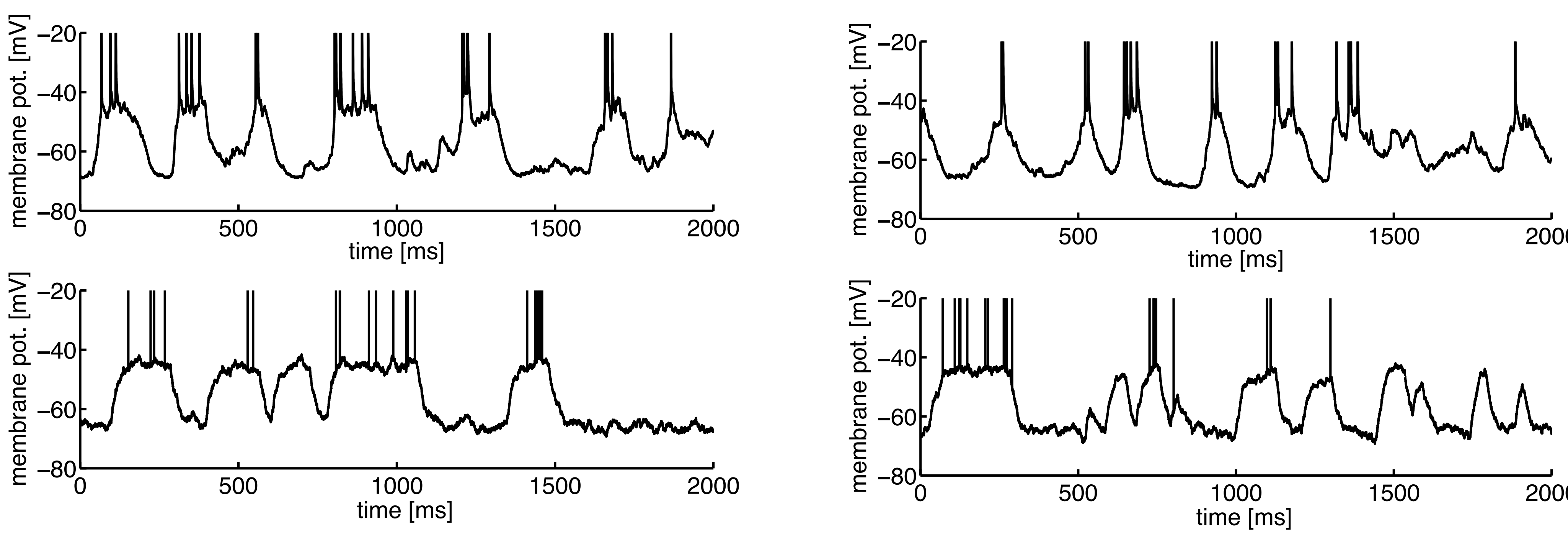
$$\mathrm{P}\left(u_t | u_{t-1}, u_t^{\text{rest}}\right) = \mathcal{N}\left[u_t;\, u_{t-1} + \frac{1}{\tau}\left(u_t^{\text{rest}} - u_{t-1}\right)\Delta t,\, \sigma_{\mathrm{W}}^2 \Delta t\right]$$

THE SWITCHING OU PROCESS

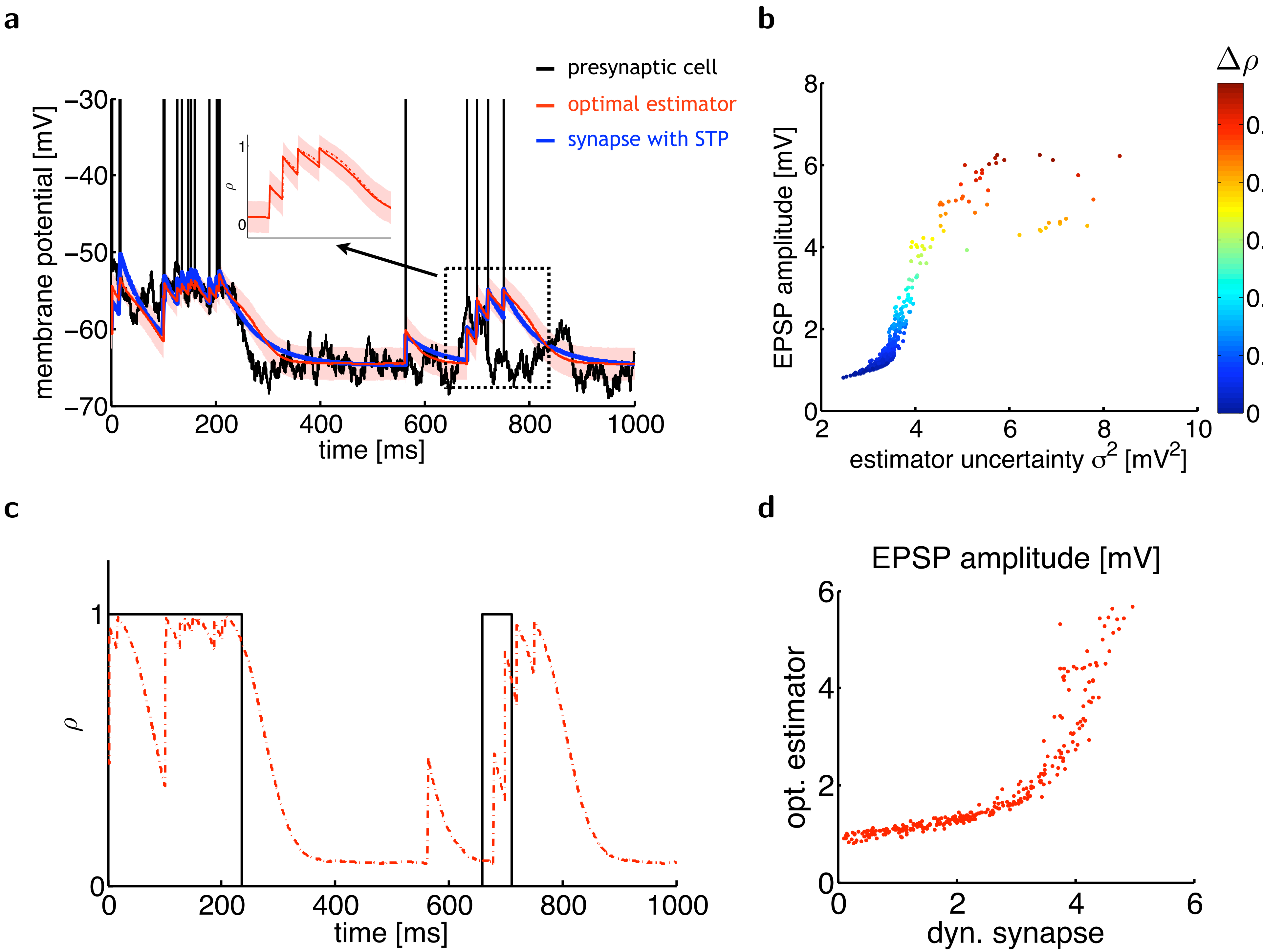
experiment

Gentet et al, 2010

model



SYNAPSES WITH STP AS OPTIMAL ESTIMATORS



Pfister et al, Nat Neurosci, 2010

MATCHING STP PARAMETERS TO NATURAL MEMBRANE POTENTIAL STATISTICS

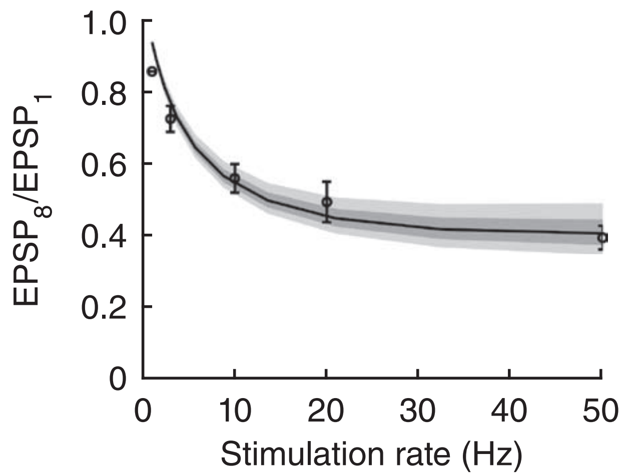
climbing fibers

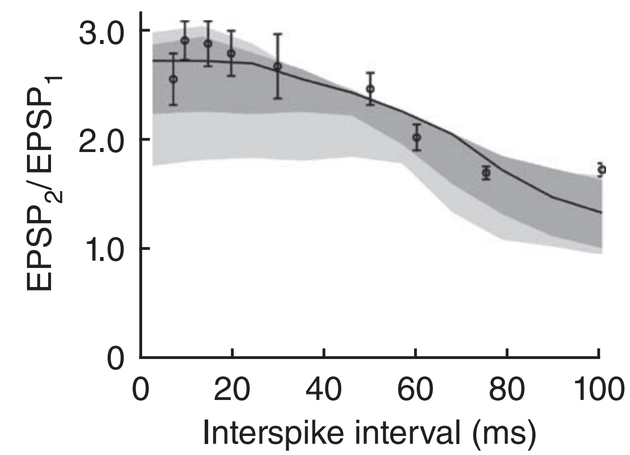
inf. olive ⟶ Purkinje cell

Schäffer collaterals

CA3 pyr ⟶ CA1 pyr

model fit





experiments
Dittman et al, 2000
Dobrunz et al, 1997

— model
Pfister et al, 2010

MATCHING STP PARAMETERS TO NATURAL MEMBRANE POTENTIAL STATISTICS

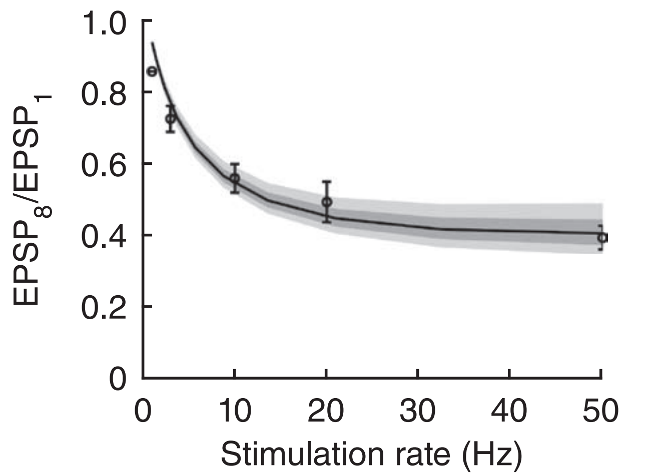
climbing fibers

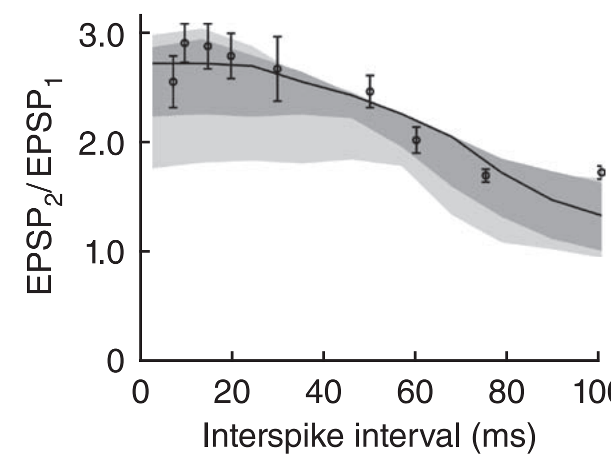
inf. olive ⟶ Purkinje cell

Schäffer collaterals

CA3 pyr ⟶ CA1 pyr

model fit



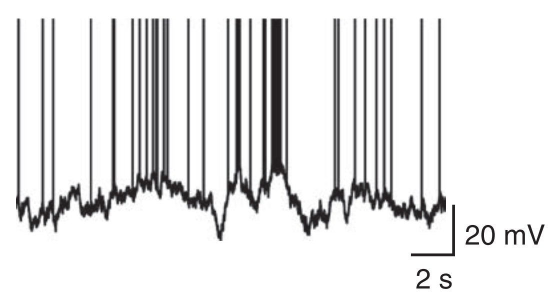


experiments
Dittman et al, 2000
Dobrunz et al, 1997

— model
Pfister et al, 2010

model ***prediction***

Pfister et al, Nat Neurosci 2010



MATCHING STP PARAMETERS TO NATURAL MEMBRANE POTENTIAL STATISTICS

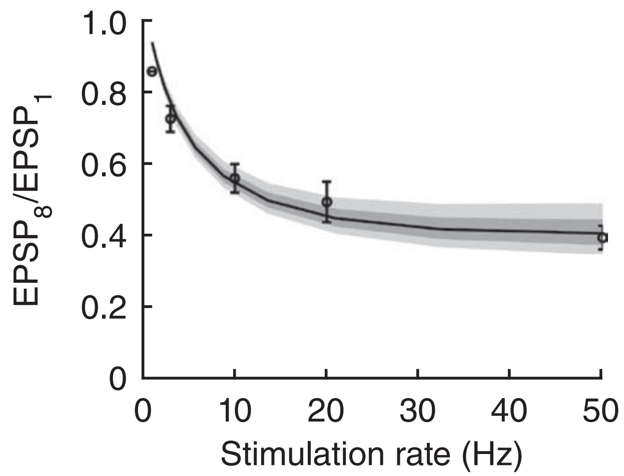
climbing fibers

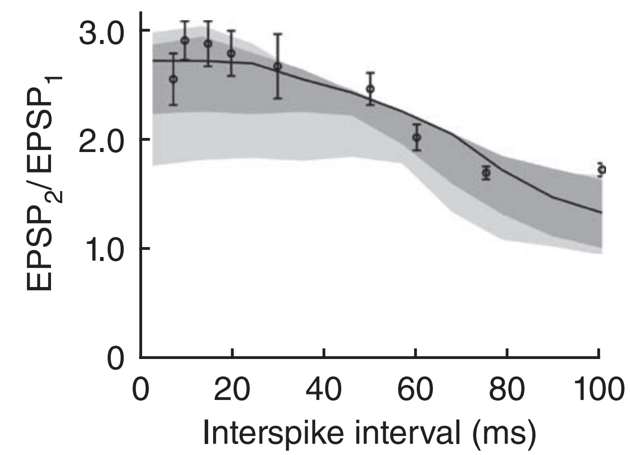
inf. olive → Purkinje cell

Schäffer collaterals

CA3 pyr → CA1 pyr

model fit



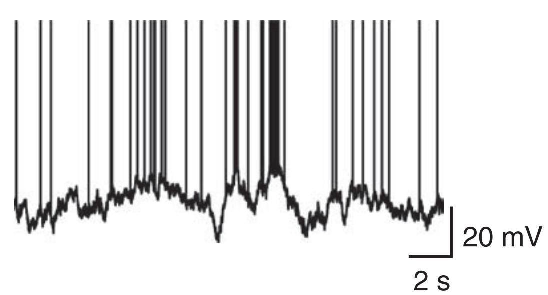


experiments
Dittman et al, 2000
Dobrunz et al, 1997

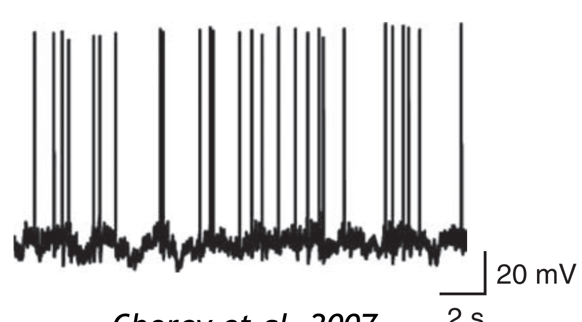
model
Pfister et al, 2010

model *prediction*

Pfister et al, Nat Neurosci 2010



experiment

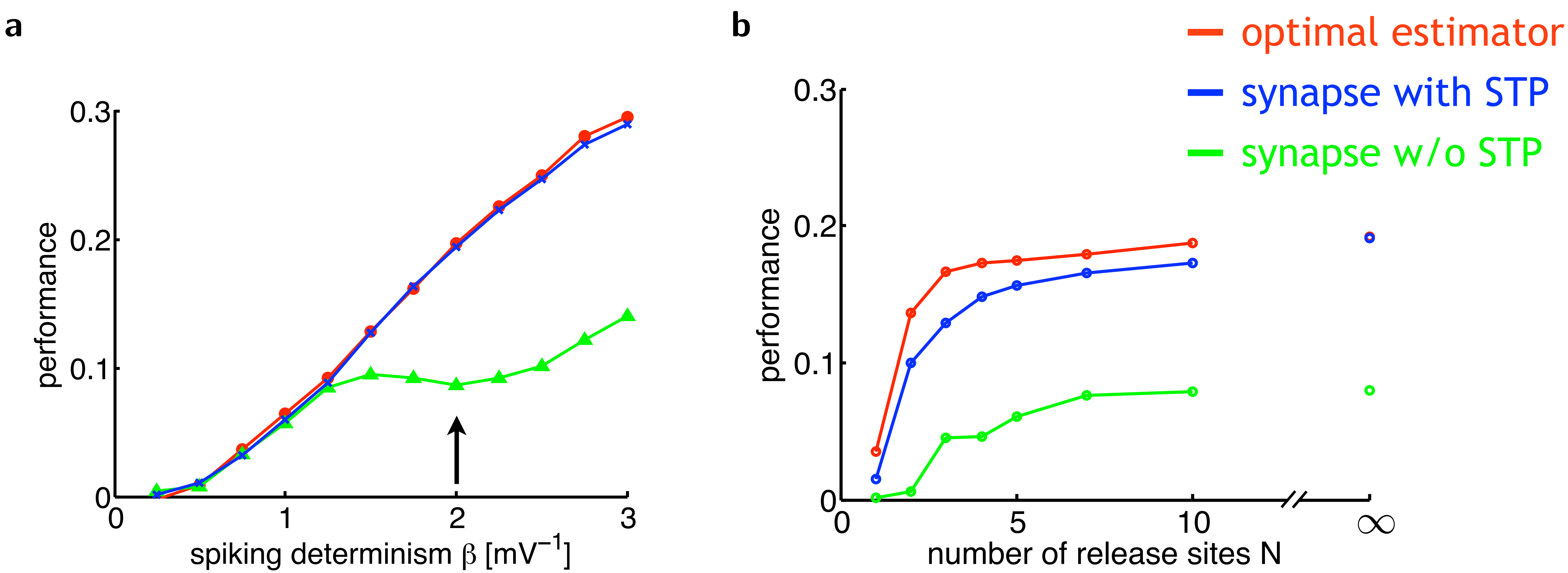


Chorev et al, 2007

Harvey et al, 2009

THE ADVANTAGE OF STP

deterministic synapse | **stochastic synapse**



Pfister et al, Nat Neurosci 2010

$$\text{performance} = 1 - \frac{1}{\sigma_{\mathrm{OU}}} \underbrace{\left[\frac{1}{T}\int_0^T \left(\hat{u}(t) - u(t)\right)^2 \, dt\right]^{\frac{1}{2}}}_{\text{RMSE}}$$

ACKNOWLEDGEMENTS

Cristina Savin

Jean-Pascal Pfister

COLLABORATORS

theory

Peter Dayan
Gatsby, UCL

***in vitro* experiments**

Ole Paulsen   Jeehyun Kwag
U Oxford

FUNDING

wellcome trust
Investigator