



# Синаптическая пластичность: обучение, память

*Николай Ильич Базенков, к.т.н.*

*Институт проблем управления им. В.А. Трапезникова РАН*

*Летняя школа РАИИ, 5-18 июля 2021 г.*

# Где хранится опыт?

**Связи**

**Нейроны**

**Активность**

# Содержание

1. Хеббовское обучение
2. STDP
3. Распознавание образов

# Синаптическая пластичность

## «Правило Хебба»

Когда аксон клетки А способен возбудить клетку В и постоянно принимать участие в ее активности, в одной или обеих клетках запускается метаболический процесс, усиливающий эффективность возбуждения клетки В клеткой А

**Fire together, wire together**

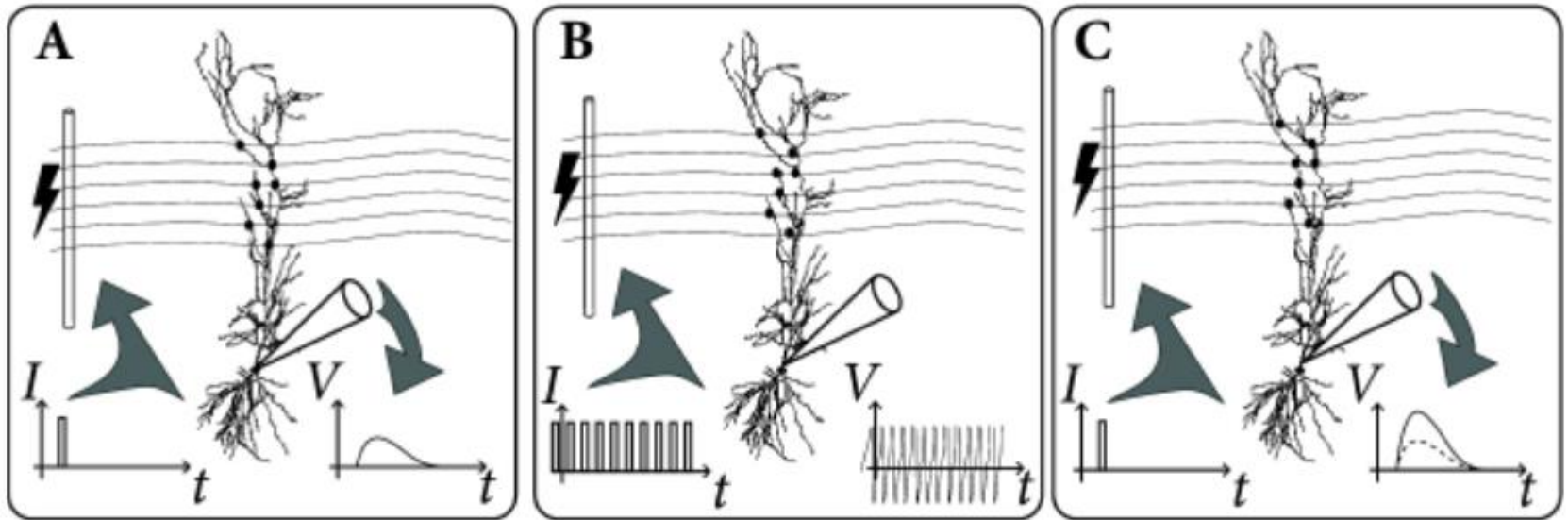
<https://neurondynamics.epfl.ch/online/Ch19.html>



Дональд Хебб  
(1904-1985)

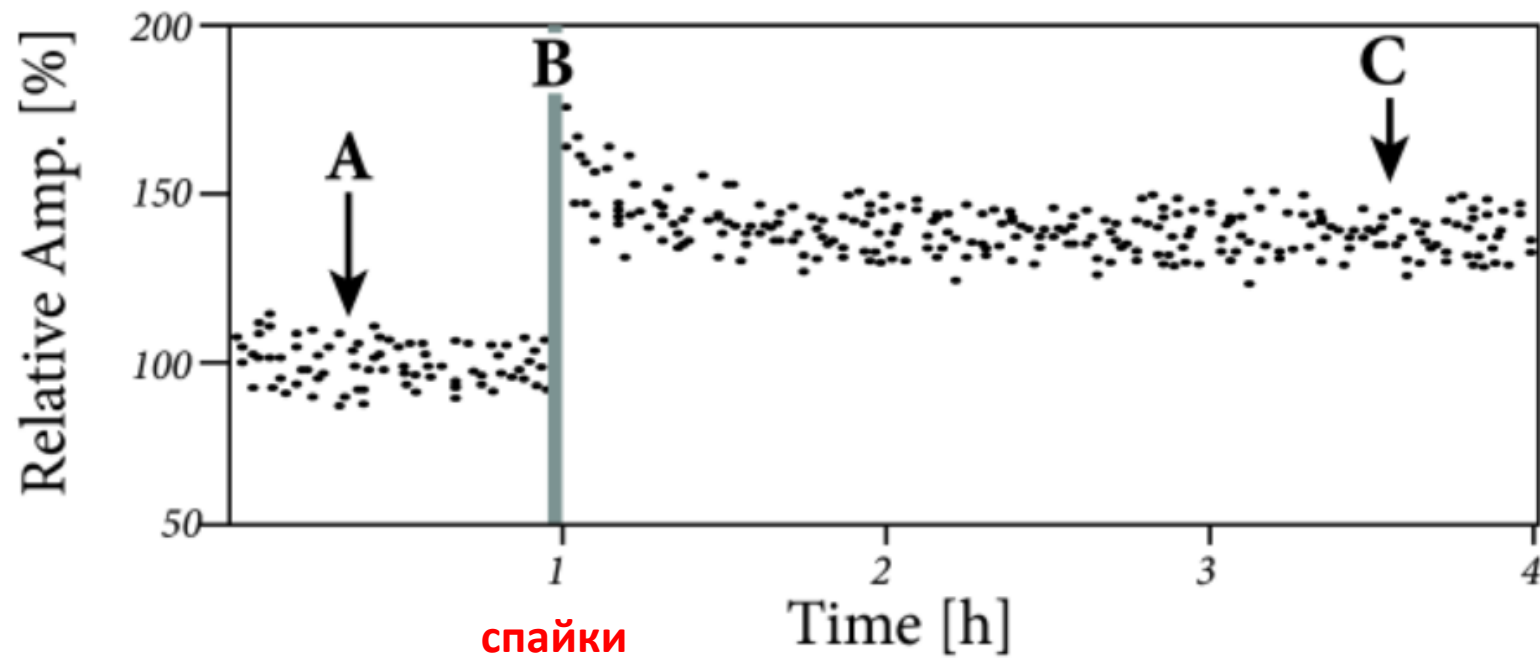
# Долговременная потенция

**Long-term potentiation (LTP)** – долговременное усиление связи между нейронами



спайки

# Долговременная потенция



# Свойства синаптической пластичности

1. Локальность
2. Сочетание активности двух нейронов
3. Без учителя – нейроны не знают про смысл своей активности

## Хеббовское обучение

- Совпадающая активность (корреляция)

## STDP

- Временная последовательность (каузальность)

## Анти-хеббовское обучение

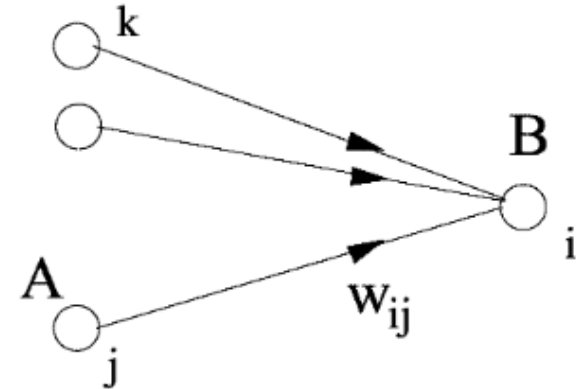
- Ослабление после совместной активности

# Математическая модель

Будем рассматривать частоту спайков

- Пресинаптического нейрона  $j$  –  $\nu_j$
- Постсинаптического нейрона  $i$  –  $\nu_i$

$w_{ij}$  – синаптический вес



$$\frac{d}{dt} w_{ij} = F(w_{ij}; \nu_i, \nu_j)$$

Gerstner, W., & Kistler, W. M. (2002). Mathematical formulations of Hebbian learning. *Biological cybernetics*, 87(5), 404-415.



# Общий вид правил обучения

Разложим  $F(w_{ij}; \nu_i, \nu_j)$  в ряд Тейлора до 2-го члена:

$$\frac{d}{dt} w_{ij} = c_0(w_{ij}) + c_1^{\text{pre}}(w_{ij}) \nu_j + c_1^{\text{post}}(w_{ij}) \nu_i + c_2^{\text{pre}}(w_{ij}) \nu_j^2 + c_2^{\text{post}}(w_{ij}) \nu_i^2 + c_{11}^{\text{corr}}(w_{ij}) \nu_i \nu_j + \mathcal{O}(\nu^3) .$$

Комбинируя разные члены разложения, можно получать разные правила обучения

Коэффициенты могут зависеть от  $w_{ij}$

# Линейное правило

$$\frac{d}{dt} w_{ij} = c_{11}^{\text{corr}} \nu_i \nu_j .$$

$c_{11}^{\text{corr}} > 0$  – Хеббовское обучение

$c_{11}^{\text{corr}} < 0$  – Антихеббовское обучение

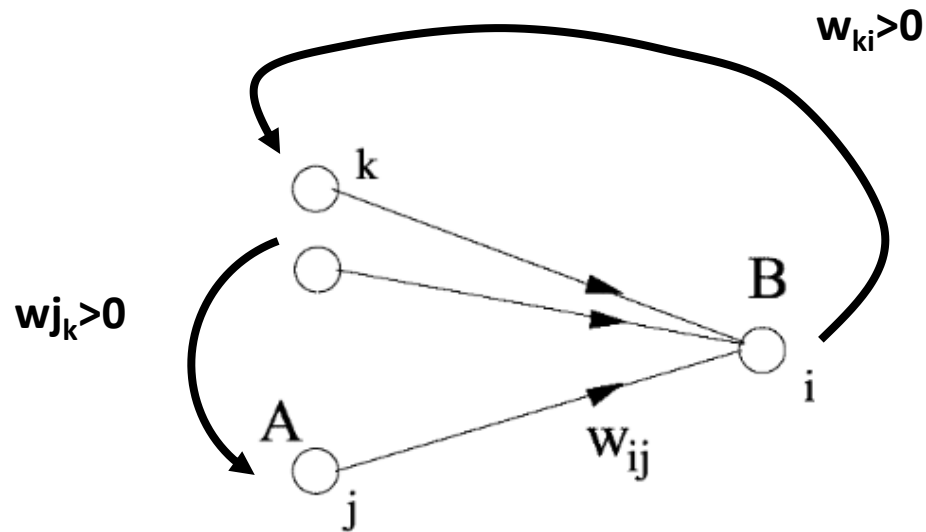
Хотя Хебб подразумевал причинно-следственную связь между нейронами, сейчас хеббовским обычно называют усиление связи, если есть корреляция между активностью нейронов

# Линейное правило

Что будет происходить в сети с линейным правилом?

Частота  $v_i(t)$ :

$$v_i(t) = \sum_k w_{ik} v_k(t)$$



# Стабилизация весов

**Насыщение** – есть максимальный вес  $w^{\max}$ :

**“Hard” bound:**

$$c_{11}^{\text{corr}} = \text{const}, \quad 0 < w_{ij} < w^{\max}$$
$$c_{11}^{\text{corr}} = 0, \quad w_{ij} = w^{\max}$$

**“Soft” bound**

$$c_{11}^{\text{corr}}(w_{ij}) = \gamma_2 (w^{\max} - w_{ij})^\beta$$

$$\gamma_2, \beta > 0$$

**Забывание**

$$\frac{d}{dt} w_{ij} = \gamma_2 (1 - w_{ij}) \nu_i \nu_j - \gamma_0 w_{ij}$$

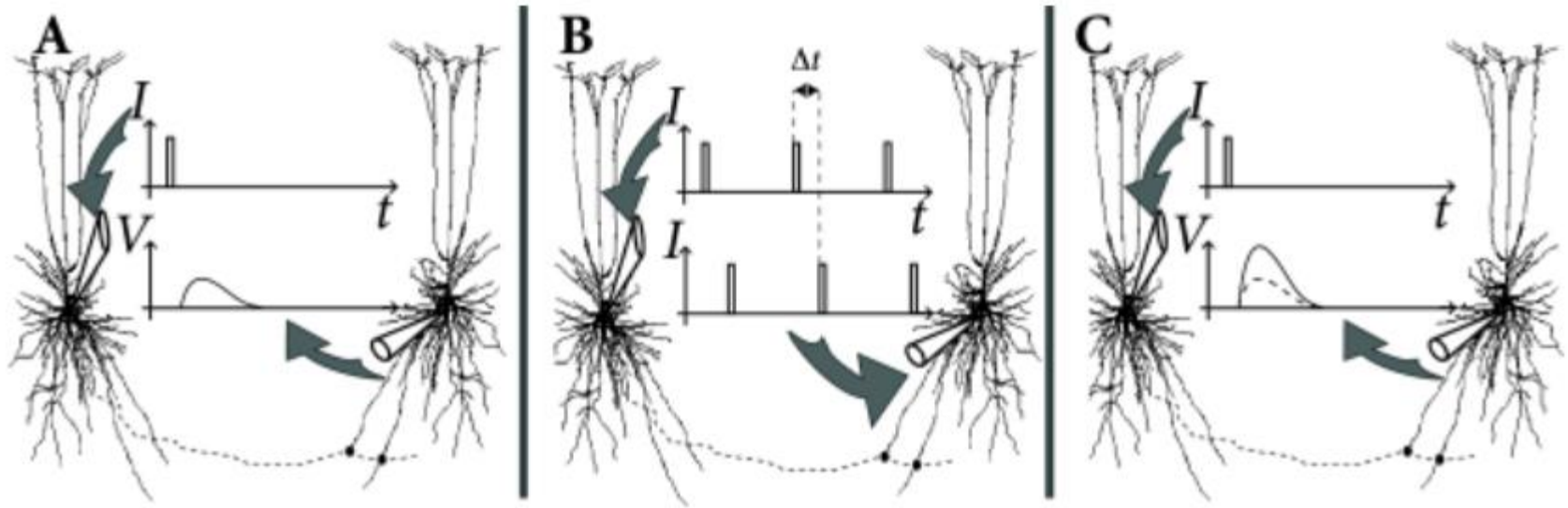
При отсутствии возбуждения веса возвращаются к 0

# Хеббовское обучение

1. **Локальные правила.** Изменение весов связано только с ближайшими нейронами
2. **Кооперация.** Для изменения нужна совместная активность нейронов
3. **Подавление.** Веса должны убывать при отсутствии активности
4. **Ограниченность.** Веса не должны расти неограниченно
5. **Конкуренция.** Усиление одних весов должно сопровождаться ослаблением других
6. **Долговременная устойчивость.** В процессе обучения новая информация не должна стирать прошлый опыт

# Пластичность, зависящая от времени спайков

**Spike Timing Dependent Plasticity (STDP)** – изменение веса зависит от разницы времени появления пре- и пост-синаптических спайков



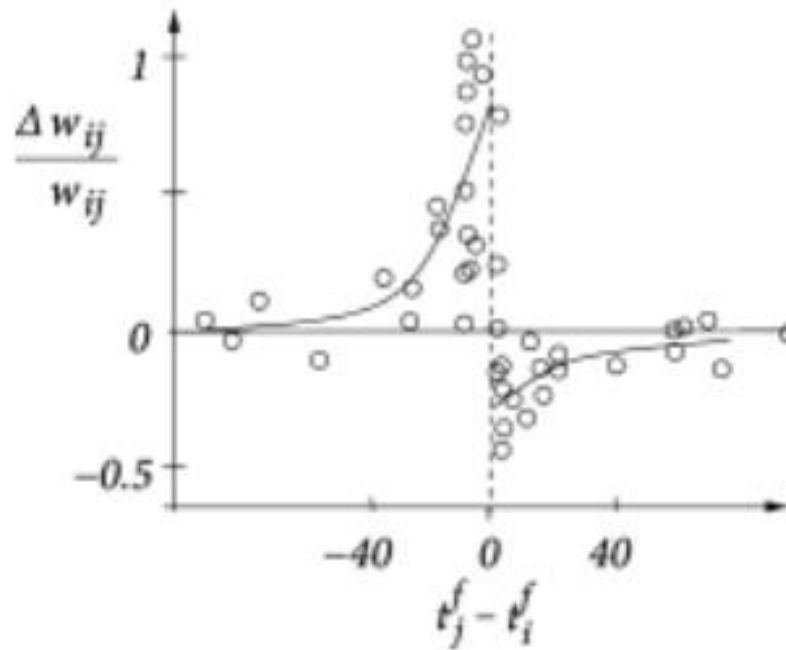
# STDP



$j$  - пресинаптический

$i$  - постсинаптический

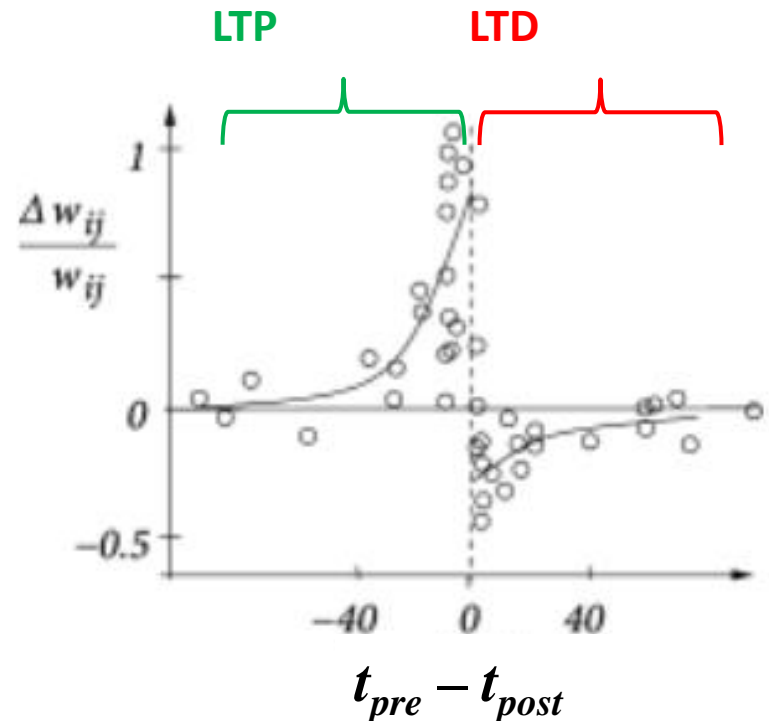
$w_{ij}$  – вес связи от  $j$  к  $i$



# Математическая модель STDP

1. Вес растет, если  $t_{\text{pre}} < t_{\text{post}}$
2. Вес уменьшается, если  $t_{\text{pre}} > t_{\text{post}}$
3. Изменение веса максимально, если

$$t_{\text{pre}} = t_{\text{post}} - \varepsilon,$$
$$\varepsilon \rightarrow 0$$



**LTP – Long-term potentiation**

$$\Delta w_+ = A_+(w) \cdot \exp(-|\Delta t| / \tau_+) \quad \text{at } t_{\text{post}} \quad \text{for } t_{\text{pre}} < t_{\text{post}}$$

**LTD – Long-term depression**

$$\Delta w_- = A_-(w) \cdot \exp(-|\Delta t| / \tau_-) \quad \text{at } t_{\text{pre}} \quad \text{for } t_{\text{pre}} > t_{\text{post}}$$

$$|\Delta t| = |t_{\text{post}} - t_{\text{pre}}|$$



# Серия спайков

Добавим две новые переменные для каждого синапса  $j$ :

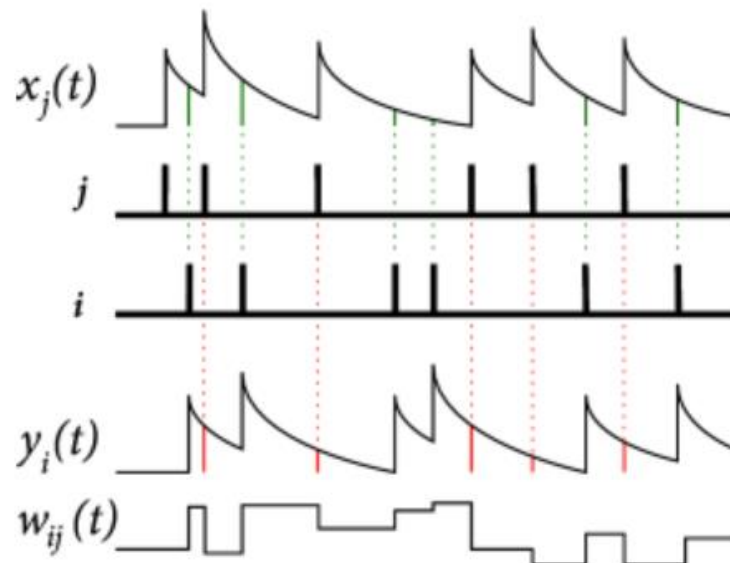
$x_j$  – суммирует пресинаптические спайки  $t_j^f$

$y_i$  – суммирует постсинаптические спайки  $t_i^f$

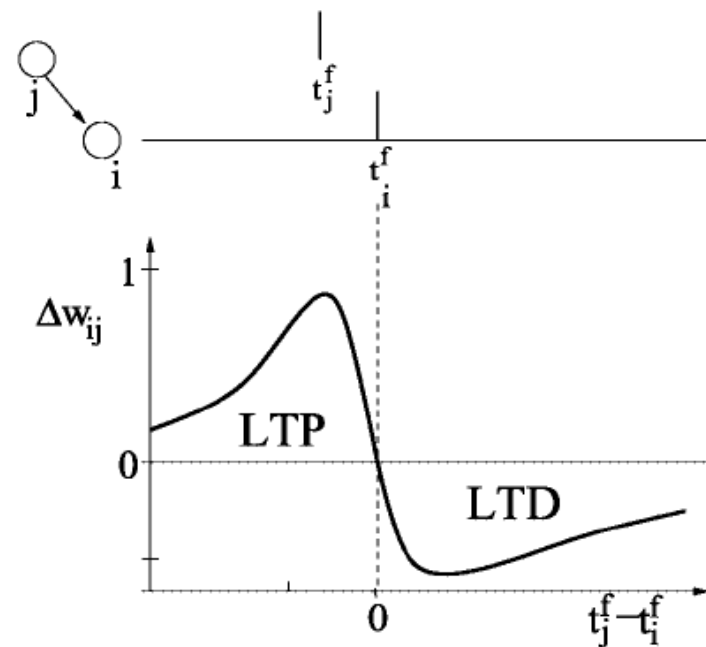
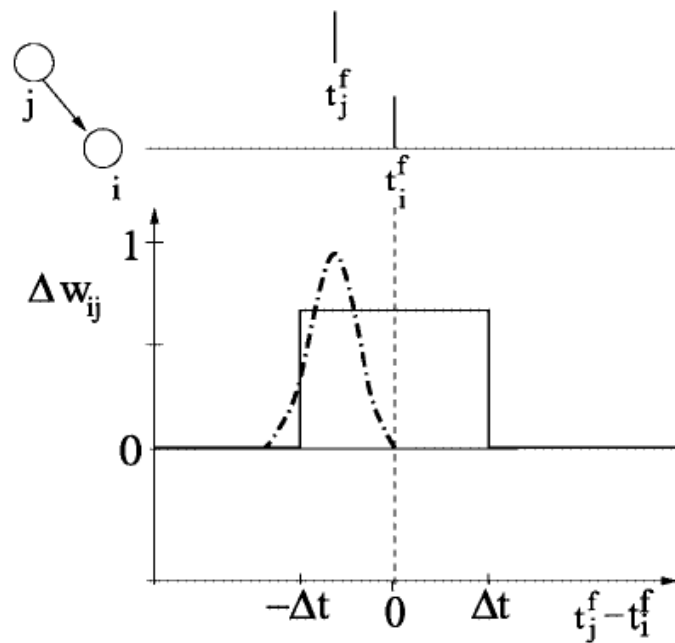
$$\frac{d}{dt} x_j = -\frac{x_j}{\tau_+} + \sum_f \delta(t - t_j^f)$$

$$\frac{d}{dt} y_i = -\frac{y_i}{\tau_-} + \sum_f \delta(t - t_i^f)$$

$$\frac{d}{dt} w_{ij}(t) = A_-(w_{ij}) y_i(t) \sum_f \delta(t - t_j^f) + A_+(w_{ij}) x_j(t) \sum_f \delta(t - t_i^f)$$



# Окно обучения



# Распознавание образов



Peter Udo Diehl

 FOLLOW

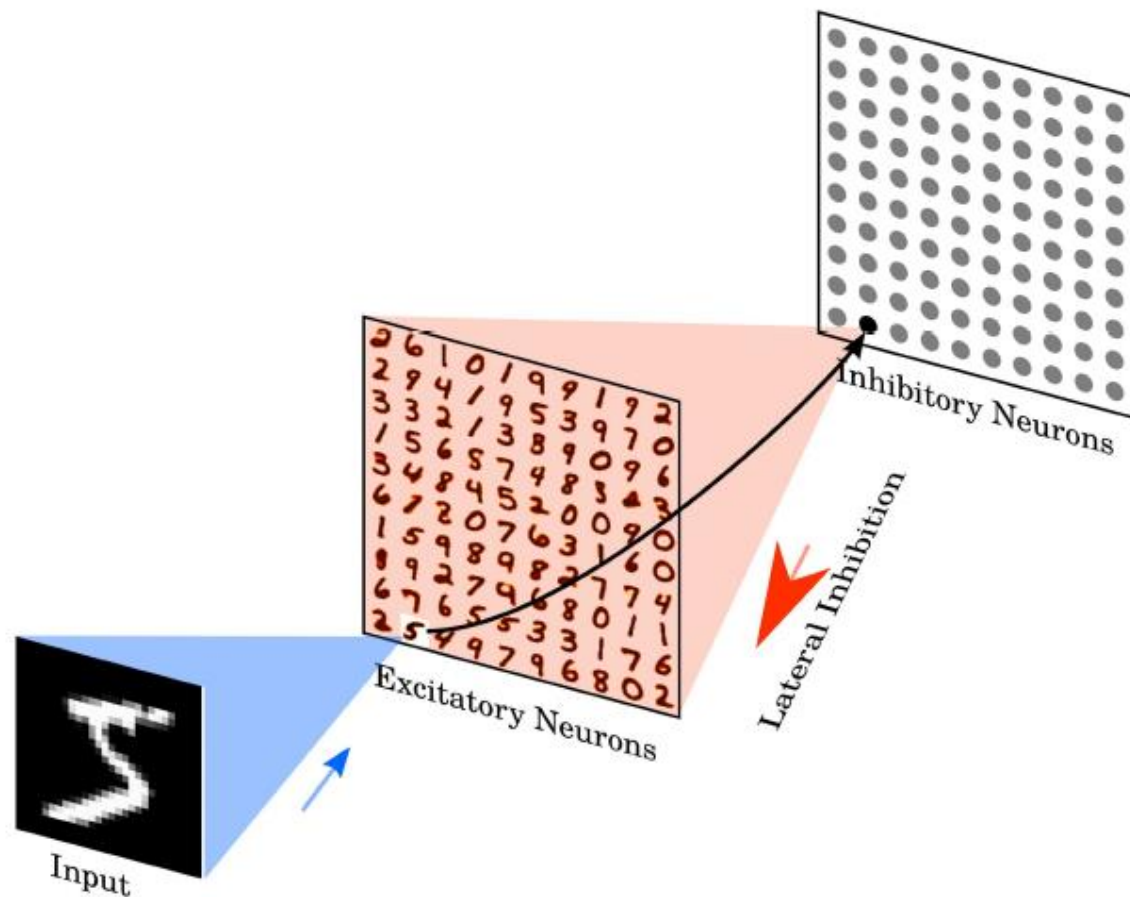
PhD, ETH Zürich

Verified email at audatic.ai - [Homepage](#)

[Deep Learning](#) [Artificial Intelligence](#) [Computational Neuroscience](#) [Audiology](#)  
[Neuromorphic Computing](#)

TITLE	CITED BY	YEAR
<a href="#">Unsupervised Learning of Digit Recognition Using Spike-Timing-Dependent Plasticity</a> PU Diehl, M Cook Frontiers in Computational Neuroscience	710	2015
<a href="#">Fast-Classifying, High-Accuracy Spiking Deep Networks Through Weight and Threshold Balancing</a> PU Diehl, D Neil, J Binas, M Cook, SC Liu, M Pfeiffer IEEE International Joint Conference on Neural Networks (IJCNN)	541	2015
<a href="#">Conversion of Artificial Recurrent Neural Networks to Spiking Neural Networks for Low</a> PU Diehl, G Zarrella, A Cassidy, BU Pedroni, E Neftci arXiv	<del>424</del> *	2016
<a href="#">Conversion of Artificial Recurrent Neural Networks to Spiking Neural Networks for Low-power Neuromorphic Hardware</a> PU Diehl, G Zarrella, A Cassidy, BU Pedroni, E Neftci arXiv	121	2016

# Распознавание образов в сети с STDP



Diehl, P. U., & Cook, M. (2015). Unsupervised learning of digit recognition using spike-timing-dependent plasticity. *Frontiers in computational neuroscience*, 9, 99.

<https://github.com/peter-u-diehl/stdp-mnist>

# Архитектура сети

## Входной слой

28 x 28 нейронов

1 нейрон = 1 пиксель

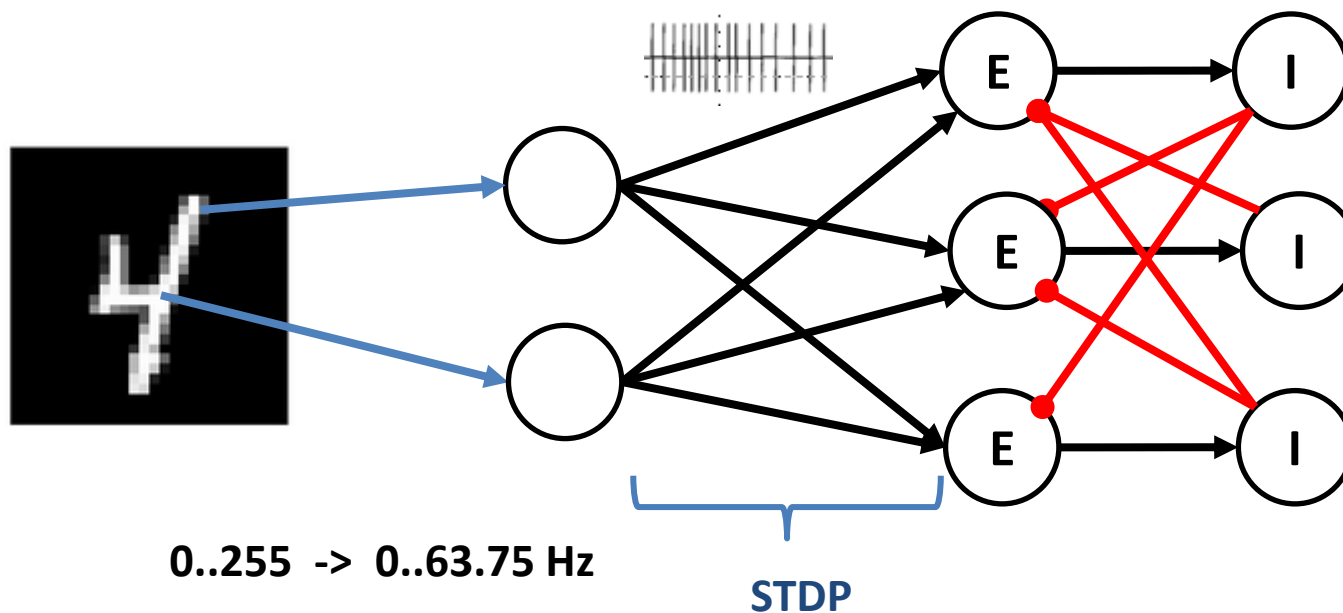
Преобразует яркость в спайки

## Слой обработки

$2n = 400 \times 2$  нейронов

1 возбуждающий связан с

1 тормозящим



# Модель нейрона

$$\tau \frac{dV}{dt} = (E_{rest} - V) + g_e(E_{exc} - V) + g_i(E_{inh} - V)$$

$V$  – мембранный потенциал

$E_{rest}$  – потенциал покоя

$g_e, g_i$  – проводимости возбуждающих и тормозящих синапсов

$E_{exc}, E_{inh}$  – равновесный потенциал синапсов

$\tau$  – постоянная времени,  $\tau_{exc} = 100$  мс,  $\tau_{inh} = 10-20$  мс

$V_{thres}, V_{reset}$  – порог и перезагрузка после спайка

Рефрактерный период 1-10 мс

Адаптация порога для возбуждающих нейронов:

$V_{thres} += \theta$

# Изменение весов

## Динамика проводимости

$$\tau_{g_e} \frac{dg_e}{dt} = -g_e$$

После поступления спайка на вход проводимость увеличивается:

$$g_e = g_e + w,$$

$w$  возрастает с поступлением новых спайков:

$$\Delta w = \eta(x_{pre} - x_{tar})(w_{max} - w)^\mu$$

$x_{pre}$  – presynaptic trace, суммирует пресинаптические спайки

$x_{tar}$  – обеспечивает убывание синаптических весов

# Процедура обучения

## 1. Предъявление образа - 350 мс

Если на выходе (E) < 5 спайков, то макс. частота  $\pm 32$  Гц и к шагу 1

## 3. Симуляция динамики сети = обучение

## 4. Пауза 150 мс

Все переменные, кроме порогов  $v_{thres}$ , возвращаются к равновесным значениям

## 5. Завершение

Если еще есть образы, то к шагу 1

Иначе зафиксировать пороги  $v_{thres}$  и к шагу 6

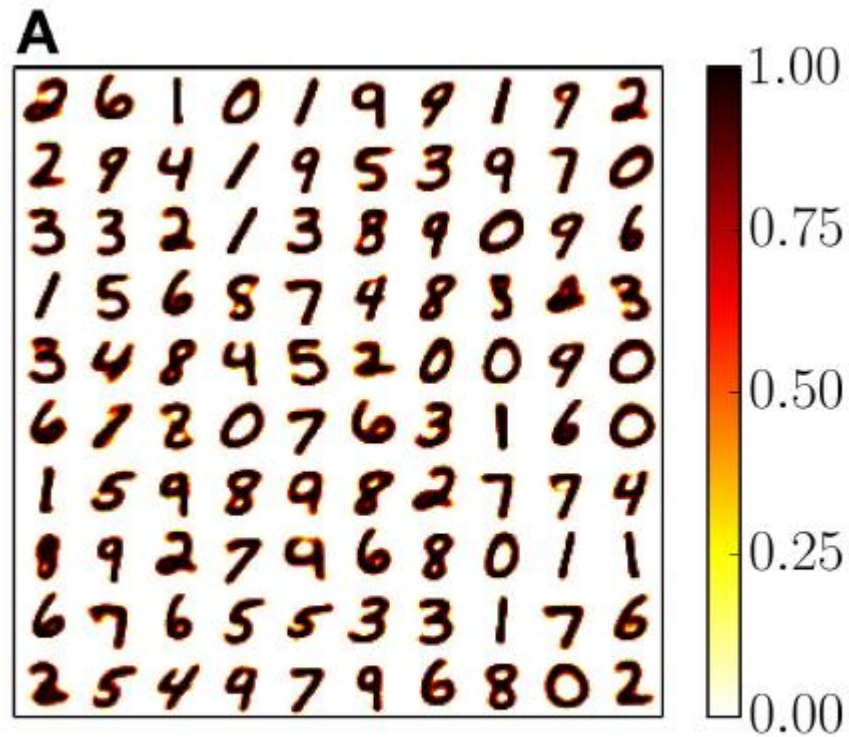
## 6. Назначение классов

Последовательно предъявляются все образы. Каждому нейрону назначается класс в соответствии с его максимальным откликом



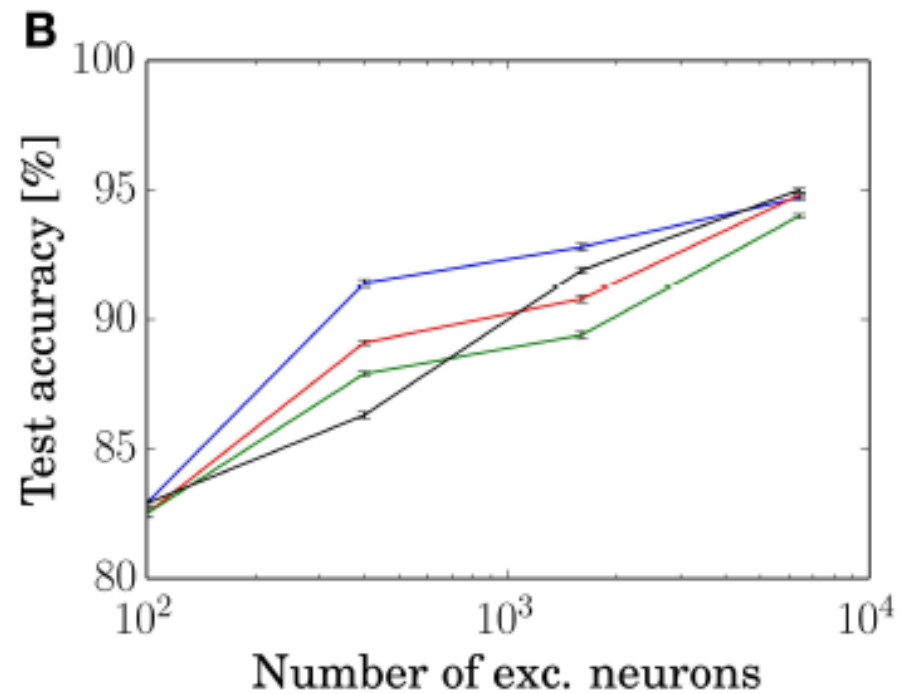
# Результаты

Каждый из 100 нейронов выходного слоя реагирует на «свой» образ:



2D receptive fields

# Результаты



# Сравнение

**TABLE 1 | Classification accuracy of spiking neural networks on MNIST test set.**

Architecture	Preprocessing	Training-type	(Un-)supervised	Learning-rule	Performance
Dendritic neurons (Hussain et al., 2014)	Thresholding	Rate-based	Supervised	Morphology learning	90.3%
Spiking RBM (Merolla et al., 2011)	None	Rate-based	Supervised	Contrastive divergence, linear classifier	89.0%
Spiking RBM (O'Connor et al., 2013)	Enhanced training set to 120,000 examples	Rate-based	Supervised	Contrastive divergence	94.1%
Spiking convolutional neural network (Diehl et al., 2015)	None	Rate-based	Supervised	Backpropagation	99.1%
Spiking RBM (Neftci et al., 2013)	Thresholding	Rate-based	Supervised	Contrastive divergence	92.6%
Spiking RBM (Neftci et al., 2013)	Thresholding	Spike-based	Supervised	Contrastive divergence	91.9%
Spiking convolutional neural network (Zhao et al., 2014)	Scaling, orientation detection, thresholding	Spike-based	Supervised	Tempotron rule	91.3%
Two layer network (Brader et al., 2007)	Edge-detection	Spike-based	Supervised	STDP with calcium variable	96.5%
Multi-layer hierarchical network (Beyeler et al., 2013)	Orientation-detection	Spike-based	Supervised	STDP with calcium variable	91.6%
Two layer network (Querlioz et al., 2013)	None	Spike-based	Unsupervised	Rectangular STDP	93.5%
Two layer network (this paper)	None	Spike-based	Unsupervised	Exponential STDP	95.0%