НИС: методы искусственного интеллекта в робототехнике

Александр Панов и Константин Яковлев

НИУ ВШЭ

16 октября 2017

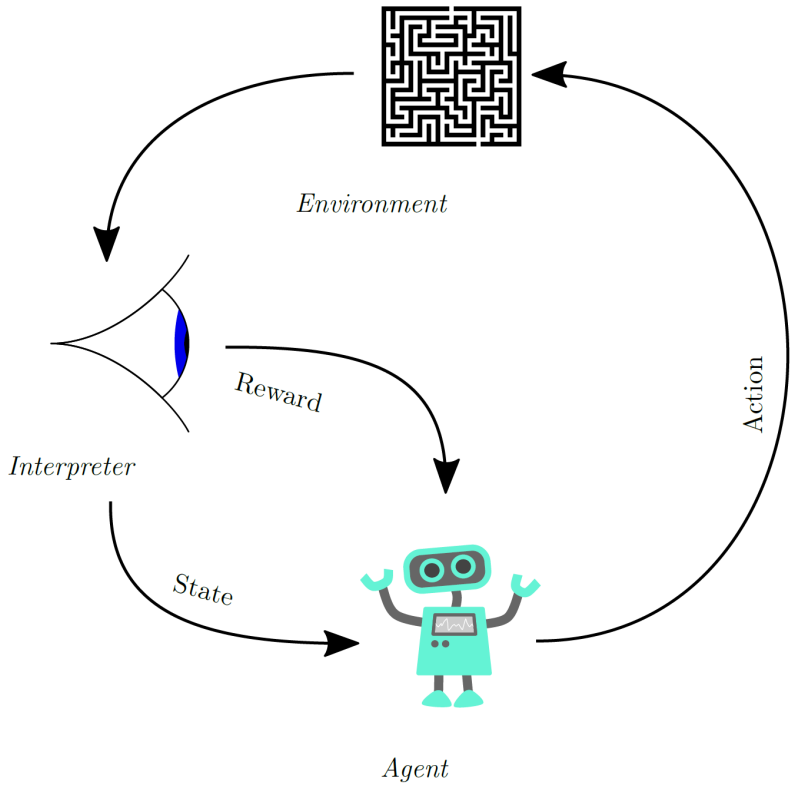
apanov@hse.ru

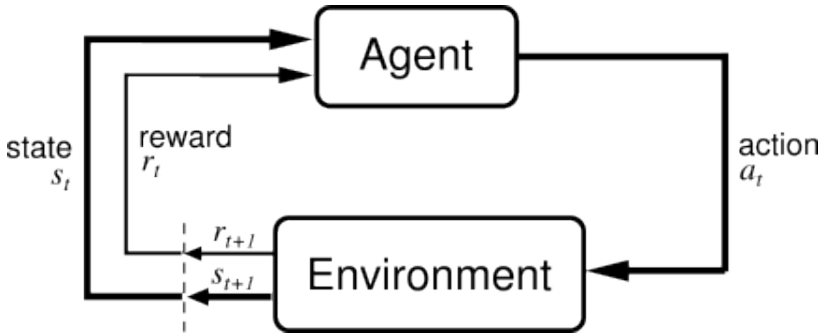
Техническое

1	04.09.2017	Яковлев	Интеллектуальная робототехника (ИИ + роботы + интеллектуальные агенты). Постановочная занятие. Знакомство с предметной областью. Основные определения. Проекты. Направления исследований.
2	25.09.2017	Панов	Архитектуры управления робототехническими системами: основные понятия, принципы и организация. Память и обучение в когнитивных архитектурах. Модели представление знаний и их пополнения.
3	02.10.2017	Яковлев	Многоуровневые интеллектуальные системы управления. Тактический уровень (SLAM, навигация, планирование траектории). Информированный и неинформированный поиск в решении навигационных задач (и не только).
4	16.10.2017	Панов	Алгоритмы обучение: иерархическая временная память. Обучение с подкреплением.
5	13.11.2017	Яковлев	Графовые модели для задач планирования траектории (2D). Алгоритмы семейства A* для решения задач планирования траектории (от основ, к динамике/перепланированию).
6	27.11.2017	Панов	Синтез плана поведения - стратегический уровень. Коллоборативная и групповая робототехника.
7	04.12.2017	Яковлев	Планирование траекторий для группы агентов. Централизованные и децентрализованные подходы.
8	18.12.2017	Панов	Психологически правдоподобные методы в робототехнических системах.

Обсуждение, вопросы, презентации и ДЗ - на странице курса на Piazza piazza.com/hse.ru/fall2017/aicognitive004

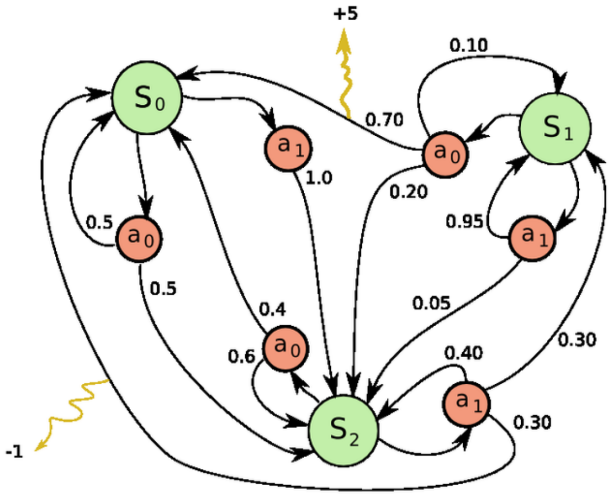
Обучение с подкреплением: постановка задачи





- t - дискретные моменты времени,
- $a_t \in A$ - действие агента в момент времени t,
- E - среда, $s_t \to s_{t+1}$ - состояния среды,
- r_t - вознаграждение, поступающее от среды,
- $R = \sum_t \gamma^t r_t$ - суммарное вознаграждение, $0 < \gamma \leq 1$ - дисконтирующий множитель.

Марковский процесс



Марковский процесс (Markov decision process (MDP)) - кортеж $\langle S, A, P, R \rangle$:

- S - конечное число состояний,
- A - конечное число действий
- $P = \{P_a(s, s') = P(s_{t+1} = s' | s_t = s, a_t = a)\}$ - вероятности переходов,
- $R = \{r_a(s, s')\}$ - вознаграждение.

Стратегия агента

Цель агента - обучиться стратегии π выбора действия в наблюдаемых состояниях среды $\pi : S \to A$, в результате применения которой он получит максимальное суммарное вознаграждение R:

$$\sum_t \gamma^t r_t \to \max_\pi$$

Баланс между исследованием среды и учетом предыдущего опыта (exploration vs explotation) - ε-жадный метод:

- с вероятность $1 - \varepsilon$ выбирается действие на основе предыдущих прецедентов,
- с вероятностью ε- случайное действие из доступных на данный момент

Параметр ε уменьшают с течением времени.

Способы решения

- Если известны $P(s_t, a_t, s_{t+1})$ и $R(s_t, a_t)$, то это задача, основанная на модели, решение *уравнения Беллмана*:

$$V(s) = \max_a Q(s, a),$$

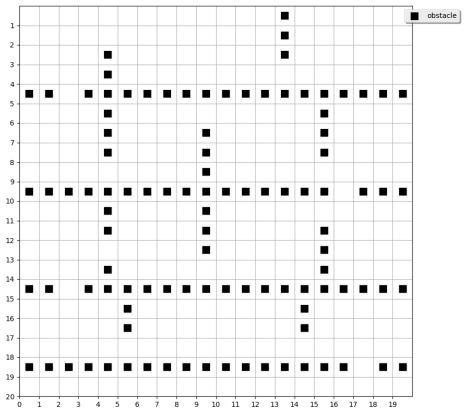
$$Q(s, a) = \sum_{s_{t+1}} P(s_t, a_t, s_{t+1}) \left(R(s_t, a_t) + \gamma V(s_{t+1})\right),$$

где $V(s) = \mathbf{E}[R|s, \pi]$ - функция полезности, а $Q(s, a) = \mathbf{E}[R|s, a, \pi]$ - функция полезности действия.

- Оценка функций полезности $V(s)$ или $Q(s, a)$.

Обучение с подкреплением: правила перемещения

- $E = (M, G)$ - среда, где M - карта местности, $G(p_s, p_f)$ - алгоритм генерации вознаграждения,
- $a_t = p_t \rightarrow p_{t+1}$ - действия агента по перемещению,
- $s_t \in R^{(2d)^2}$ - наблюдения агента (сенсорная информация).



Пусть $Q^*(s_t, a_t) = \max_\pi \mathbf{E}[R|s_t, a_t, \pi]$ - оптимальная функция полезности, тогда с учетом определения R получаем следующее уравнение Беллмана:

$$Q^*(s, a) = \mathbf{E}_{s_t \sim E}\left[r_t + \gamma \max_{a_t} Q^*(s_t, a_t) \,|s, a \right]$$

Обучение с подкреплением: аппроксимация

Для решения итерационными методами уравнения Беллмана используют различные аппроксимации функции $Q^*(s, a)$: $Q(s, a; \theta) \approx Q^*(s, a)$.

В процессе обучения происходит настройка параметров θ в результате минимизации функции потерь $L(\theta)$:

$$L_i(\theta_i) = \mathbf{E}_{s,a\sim\rho(\cdot)}\left[\left(y_i - Q(s, a; \theta_i)\right)^2\right],$$

$$y_i = \mathbf{E}_{s_t\sim E}\left[r_t + \gamma \max_{a_t} Q(s_t, a_t; \theta_{i-1})|s, a\right]$$

$$\nabla_{\theta_i} L_i(\theta_i) = \mathbf{E}_{s,a\sim\rho(\cdot);s_t\sim E}\left[\left(r_t + \gamma \max_{a_t} Q(s_t, a_t; \theta_{i-1}) - \right.\right.$$
$$\left.\left. - Q(s, a; \theta_i)\right) \nabla_{\theta_i} Q(s, a; \theta_i)\right].$$

Обучение с подкреплением: переигровки

- Эпизод - это набор действий агента и реакций среды на перемещения от начального положения до конечно, либо до достижения максимального количества действий N_a,
- $e_t = (s_t, a_t, r_t, s_{t+1})$ - прецедент сохраняется в память агента D,
- обучение идет по некоторой случайной выборке e из памяти

$$D = \left\{ e_1, \boxed{e_2}, \ldots, \boxed{e_i}, e_{i+1}, \ldots e_j, \boxed{e_{j+1}}, \ldots \right\}$$

- одно действие можно использовать несколько раз $\rightarrow$ расширяем выборку, устраняем корреляции соседних состояний.

Генерация вознаграждения

Для расчета функции вознаграждения использовали следующий алгоритм:

$$G(s,g,t) = \begin{cases} \alpha_{opt} r_t^{opt} + \alpha_{rat} r_t^{rat} + \alpha_{euq} r_t^{euq}, & p_t \leftarrow 0, \\ r^{obs}, & p_t \leftarrow 1, \\ r^{tar}, & p_t = g, \end{cases}$$

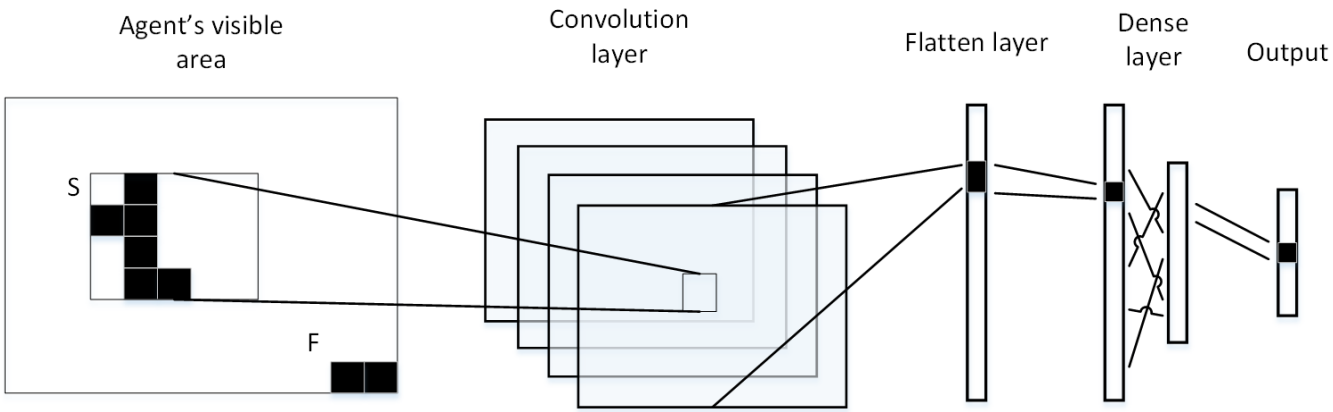
где

- $\sum \alpha_i = 1$ - нормировка,
- $r_t^{opt} = l_t - l_{t-1}$ - изменение оптимального расстояния,
- $r_t^{rat} = e^{-l_t/l_0}$ - штраф за отклонение от цели,
- $r_t^{euq} = |p_t - g| - |p_{t-1} - g|$ - регуляризатор для спрямления пути.

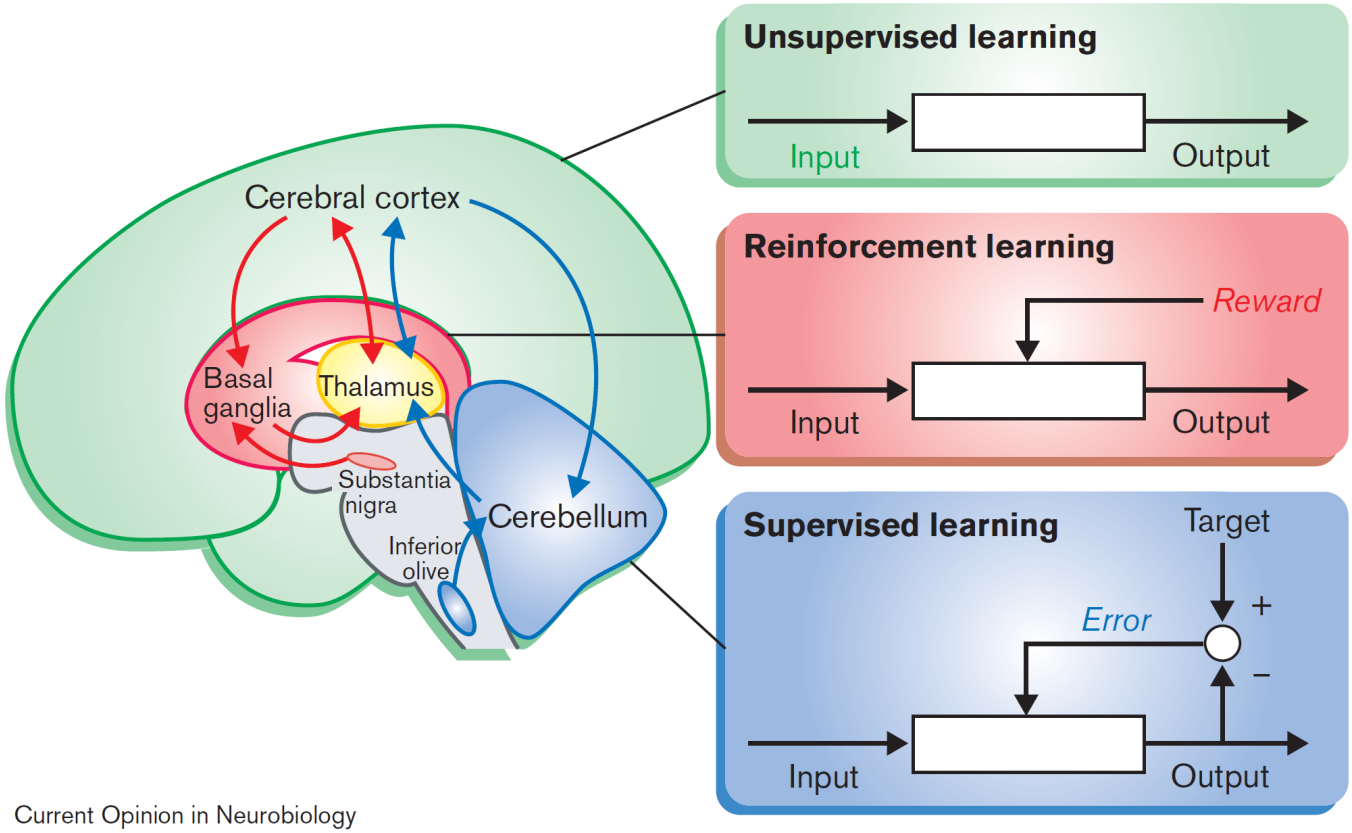
Нейросетевые архитектуры для аппроксимации Q

Можно использовать разные варианты сетей:

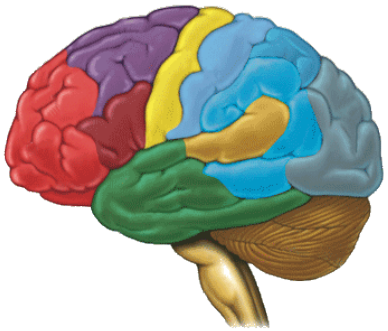
1. Ag_1 - «мелкая» полносвязная нейронная сеть,
2. Ag_2 - сверточная сеть средней глубины с полносвязными выходным слоем,
3. Ag_3 - глубокая сеть, состоящая из блоков Inception.

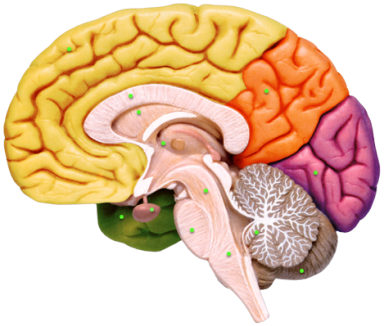


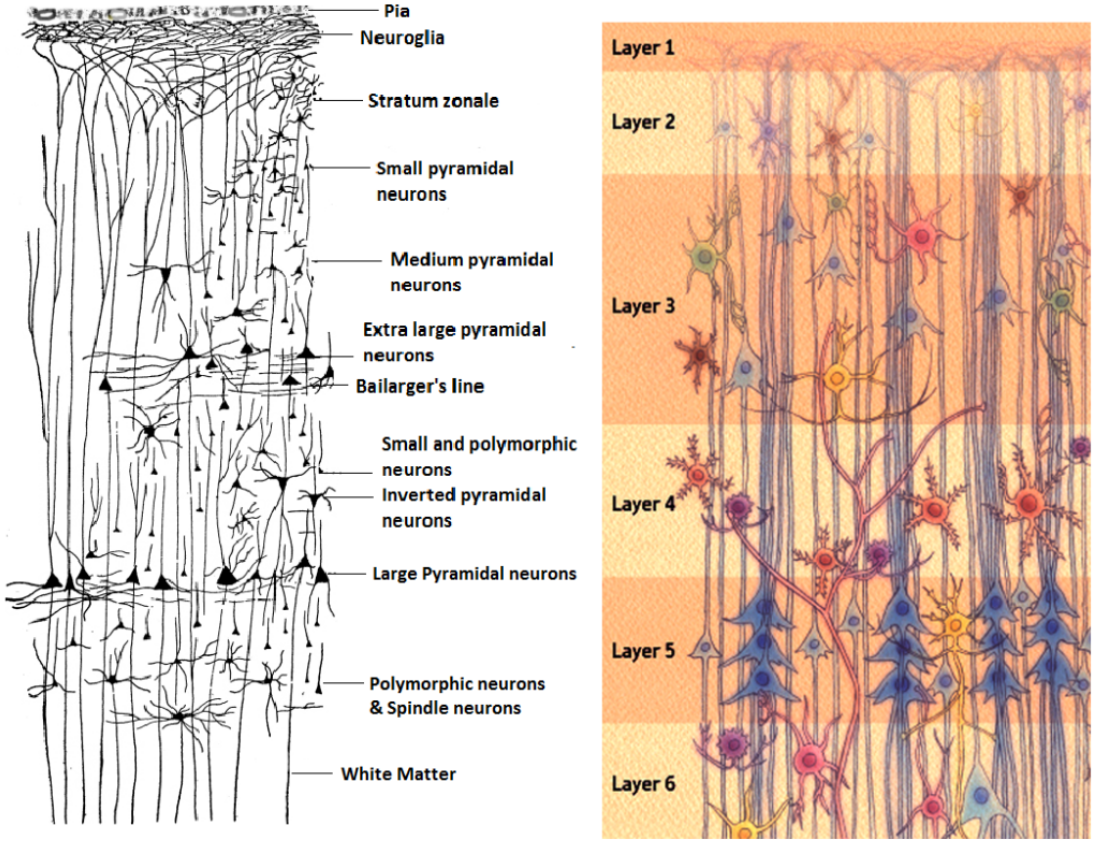
Модели обучения в мозге



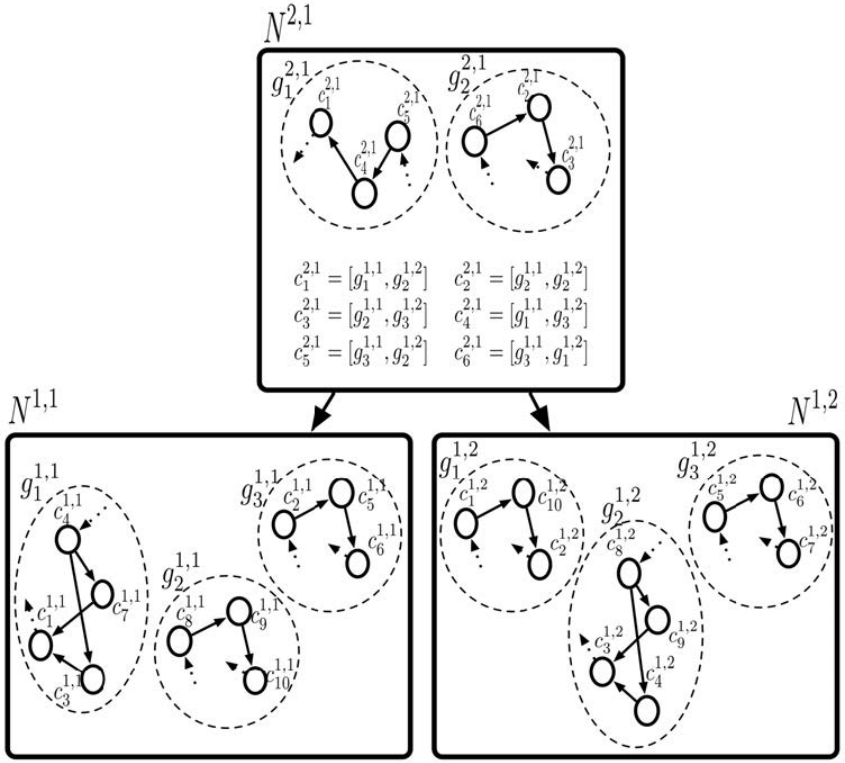
Нейронный субстрат





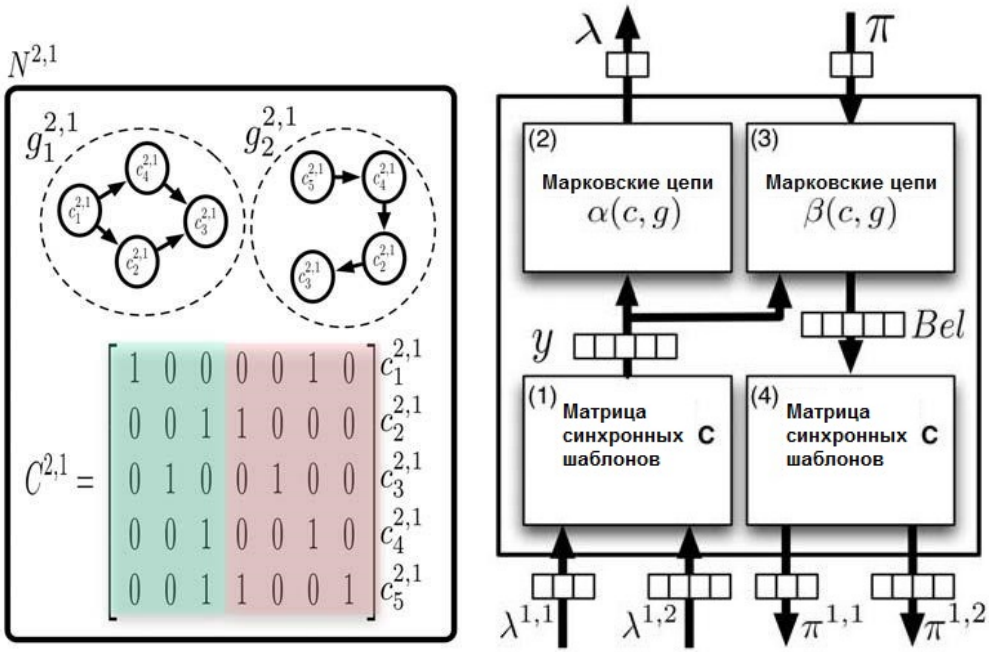


Иерархическая временная память

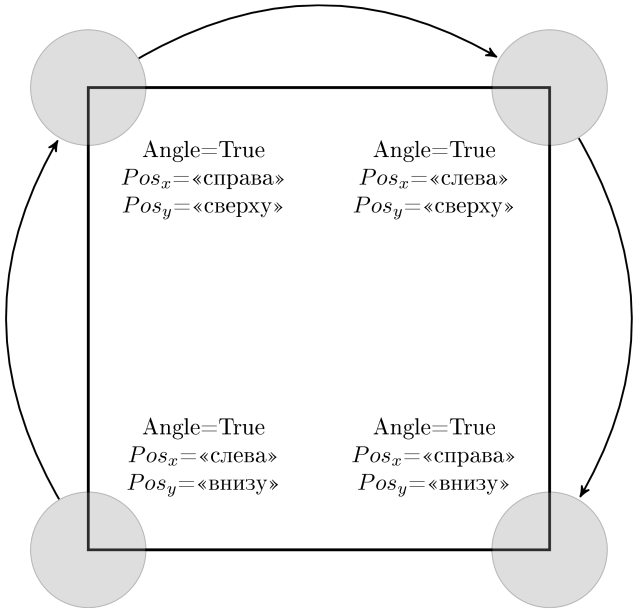


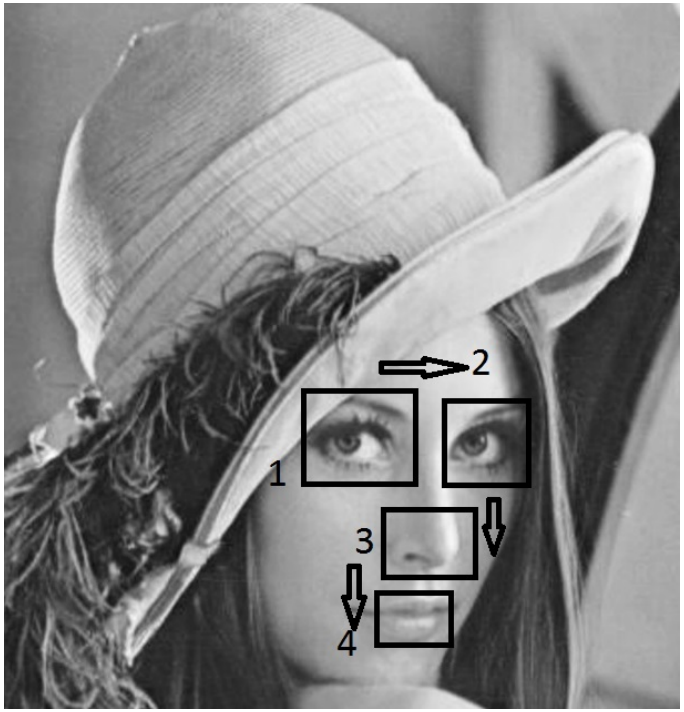
- $N^{i,j}$ - узлы сети,
- $g^{i,j}$ - временные группы,
- $c_k^{i,j}$ - паттерн (синхронные шаблоны).

Иерархическая временная память



ИВП: пример





$$z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Модель процесса обучения

К основным принципам работы механизма обучения относятся:

- использование иерархии вычислительных узлов с восходящими и нисходящими связями,
- использование Хэббовских правил обучения,
- разделение пространственного и временного группировщиков,
- подавление второстепенной активации для формирования разреженного представления.

Нейронная организация