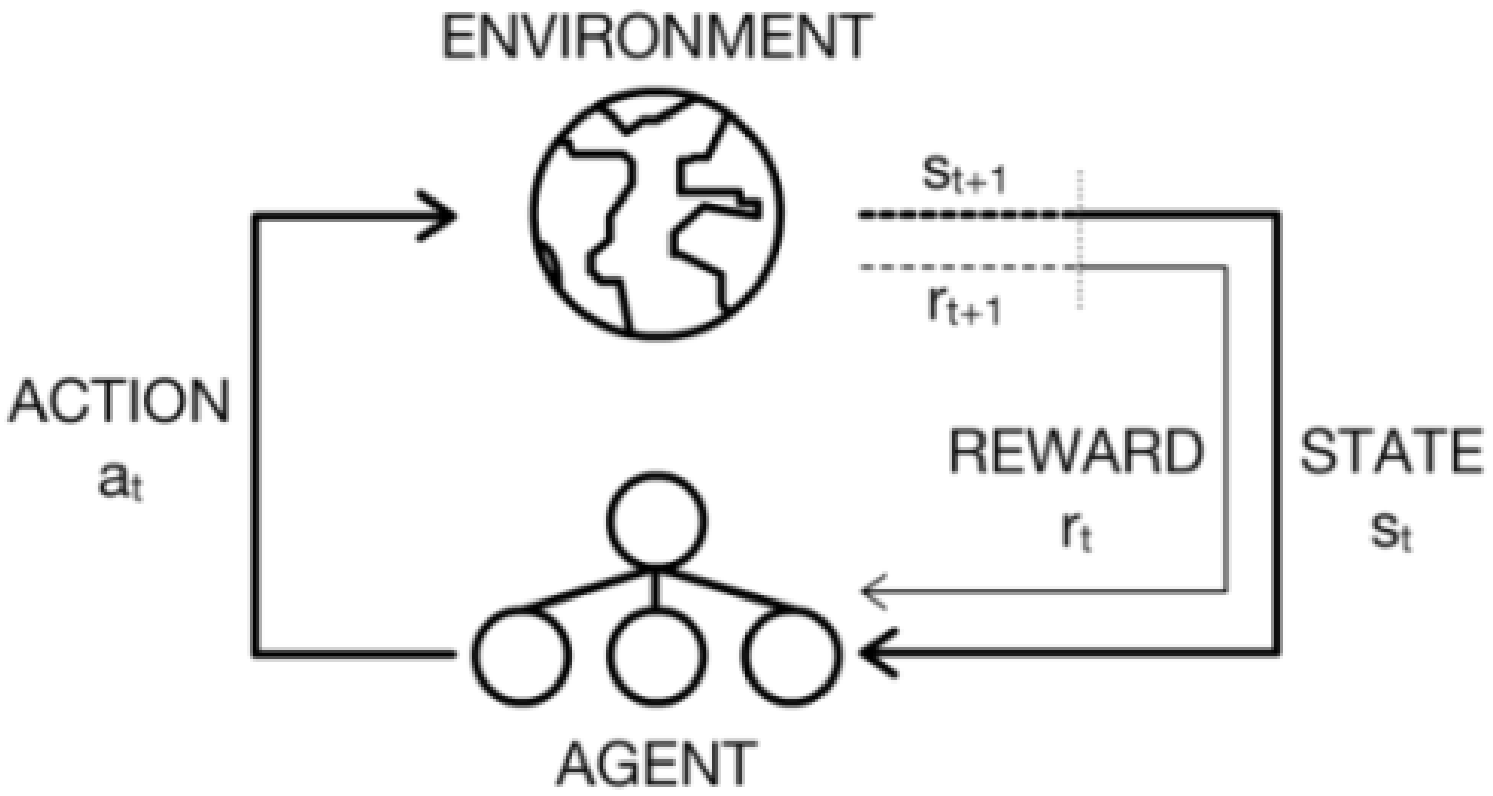

14
Euge
RL

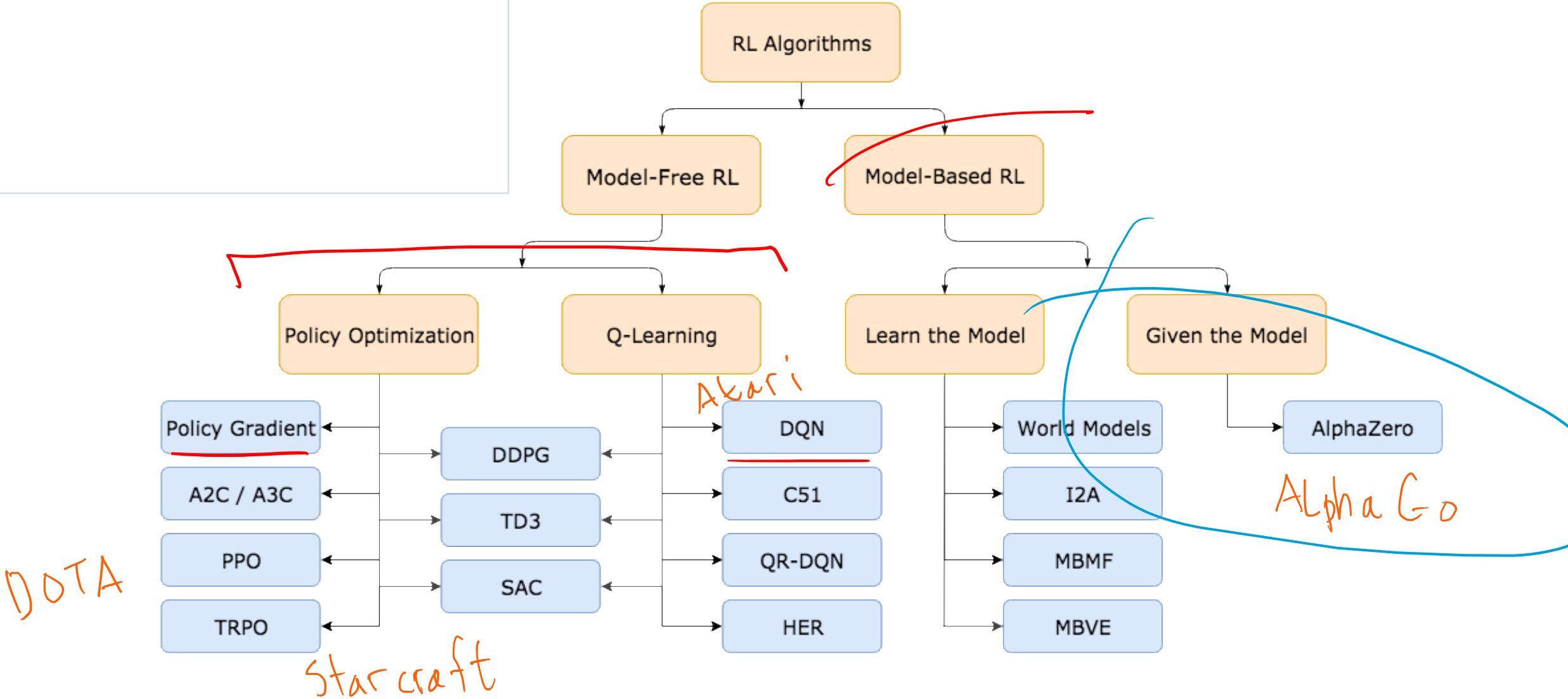
Обучение с подкреплением Reinforcement Learning



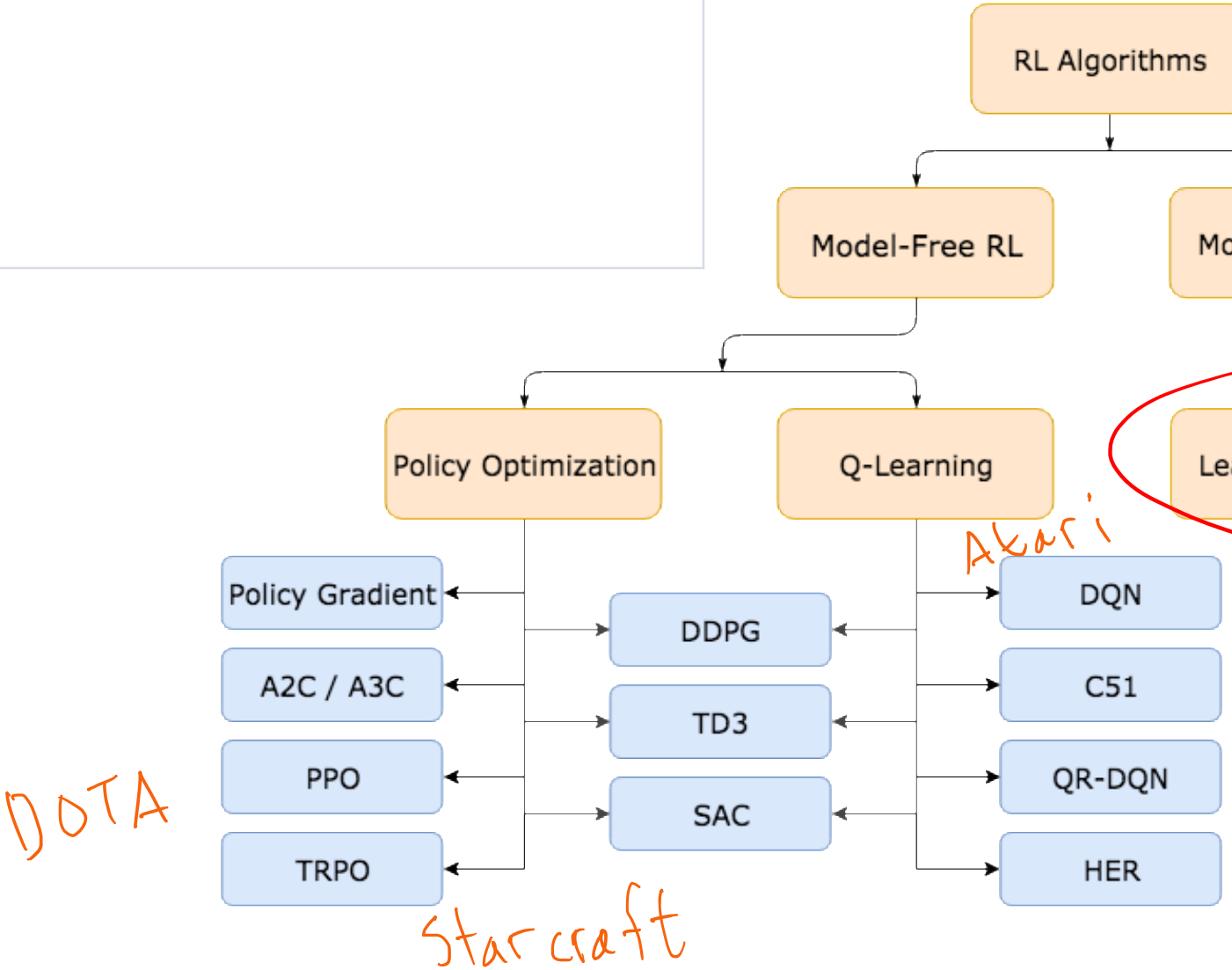
s_t – состояние среды (state) на шаге t
a_t– действие агента (action) на шаге t
r_t - награда (reward) на шаге t

Среда может:
- Быть недетерминированной
- Иметь внутреннее состояние

Source



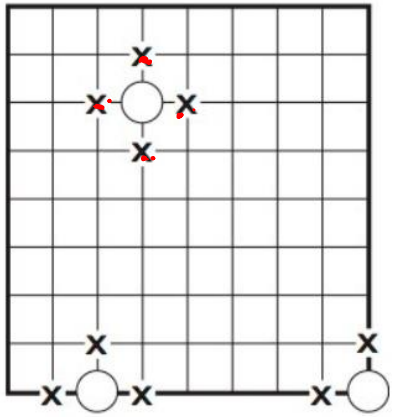
[OpenAI Spinning Up in RL]

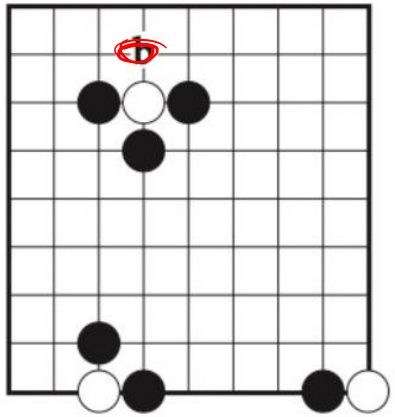


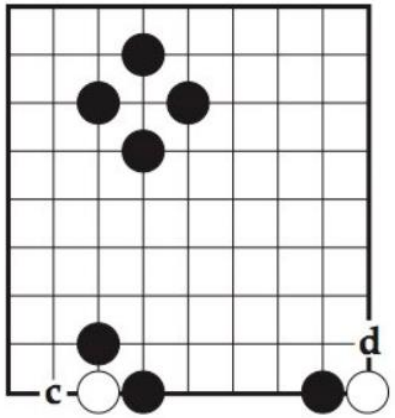
[OpenAI Spinning Up in RL]

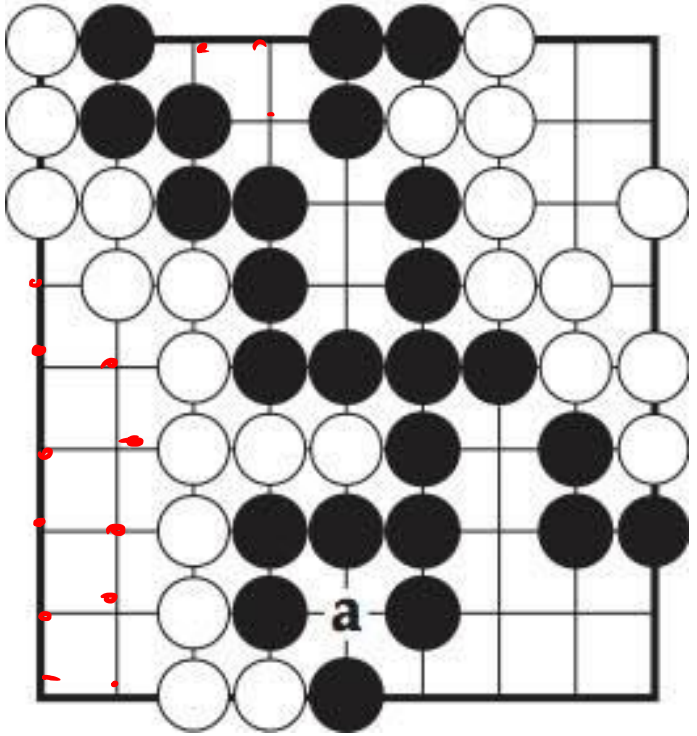

19×19
囲碁上達の秘訣
高等囲棋講義録

Правила

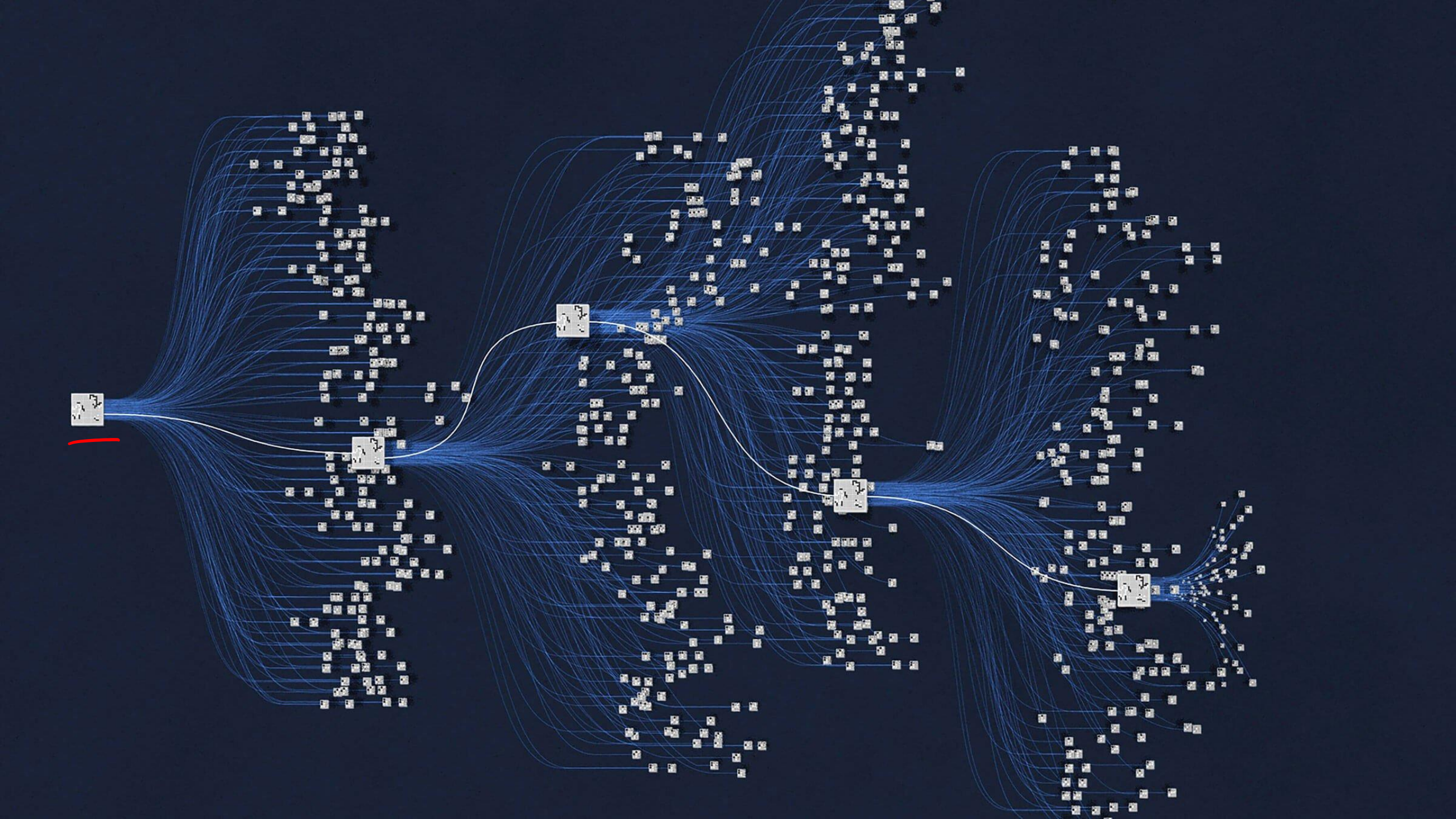


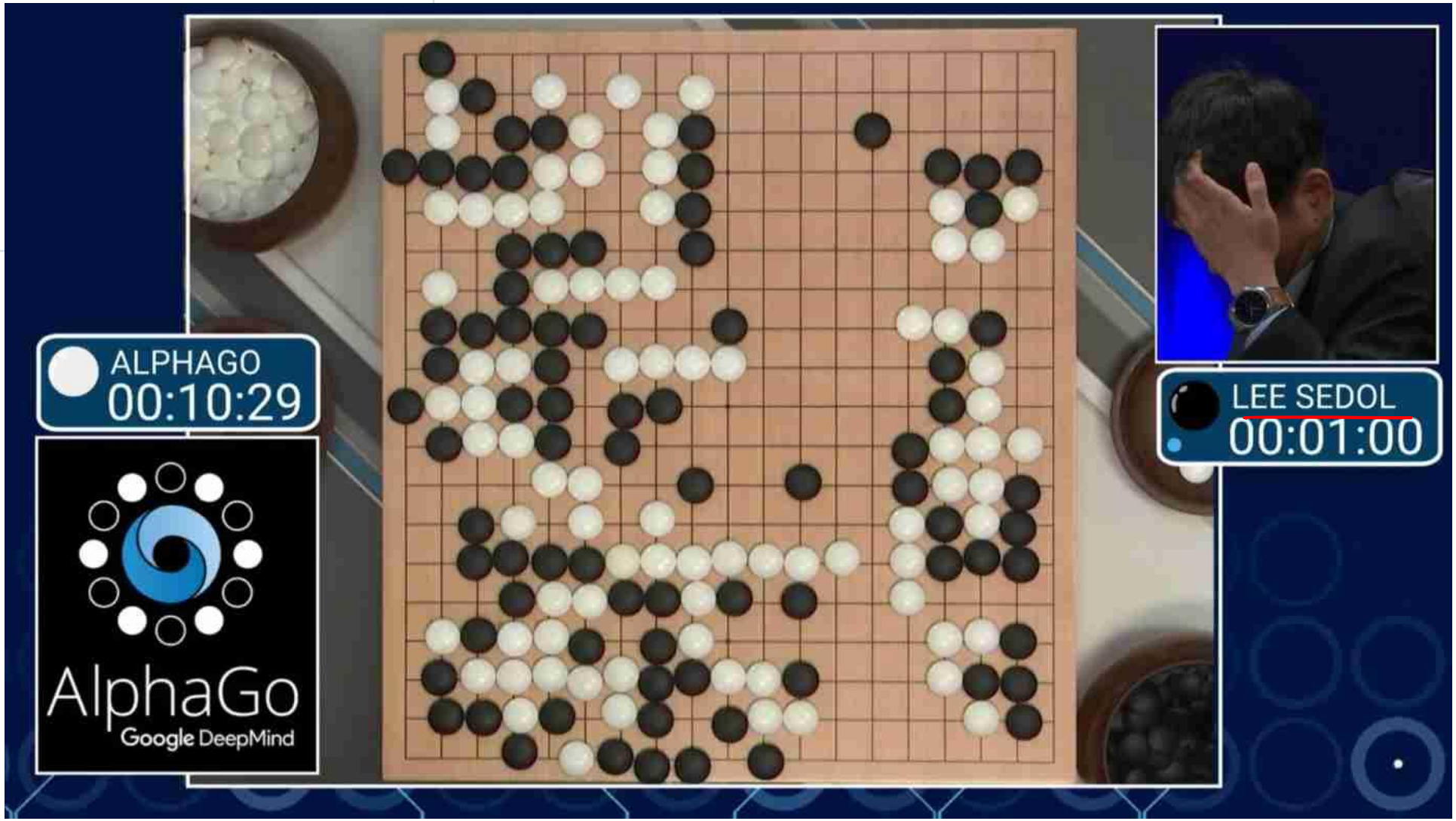






https://www.britgo.org/intro/intro2.html




ALPHAGO
00:10:29
AlphaGo
Google DeepMind
LEE SEDOL
00:01:00

Monte-Carlo Tree Search (MCTS)

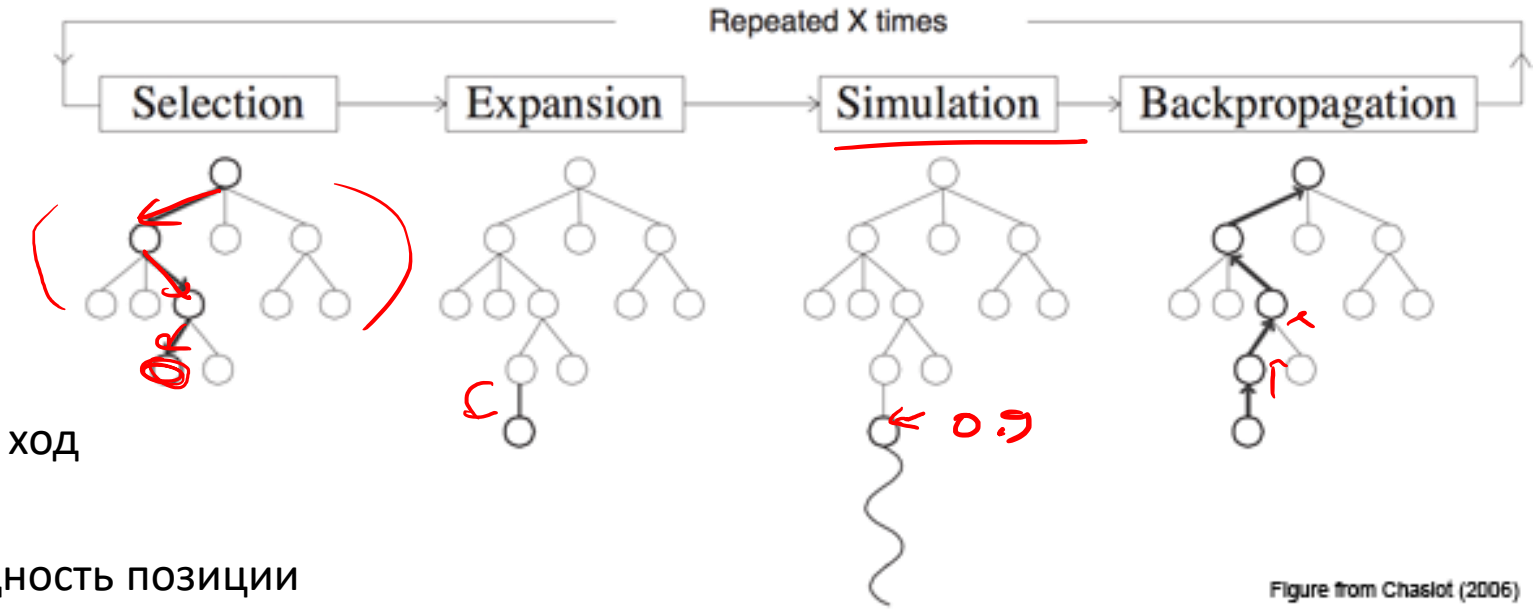
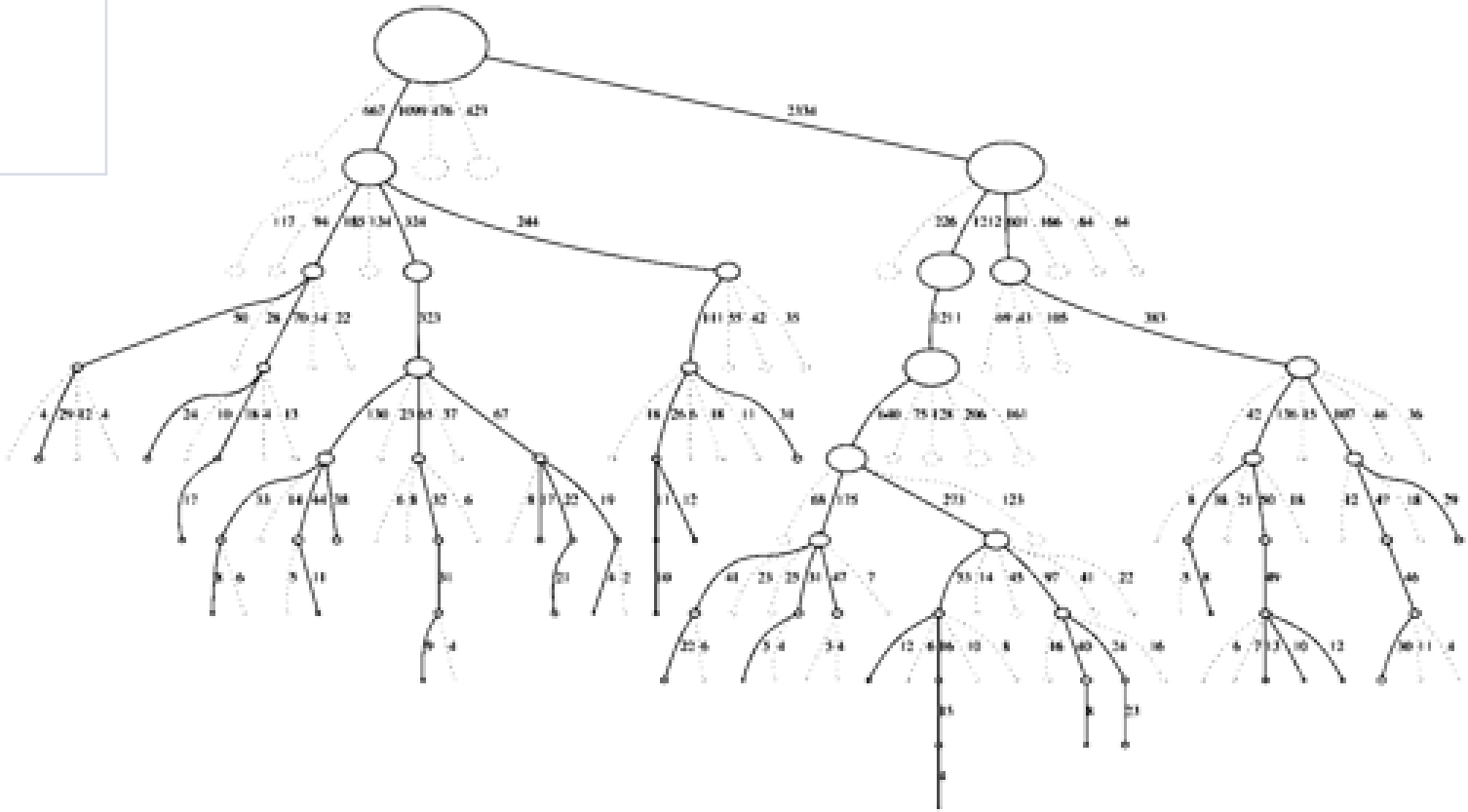


Figure from Chaslot (2006)

$P(board)$ – оценивает какой ход сделать следующим

$V(board)$ – оценивает выгодность позиции

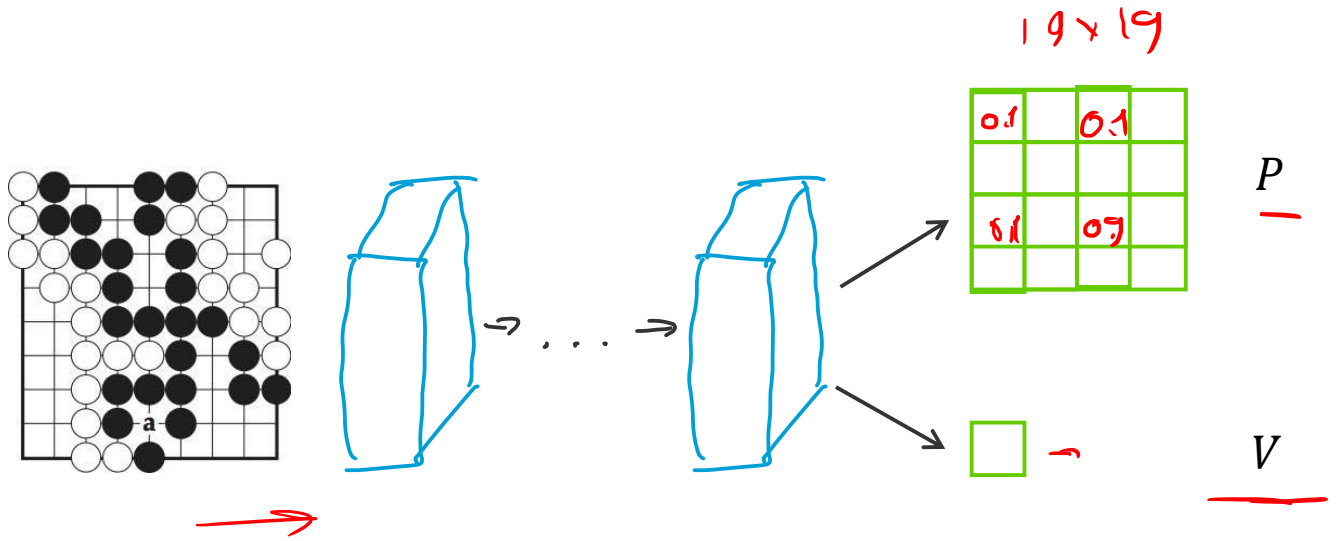
http://www.cameronius.com/research/mcts/about/index.html

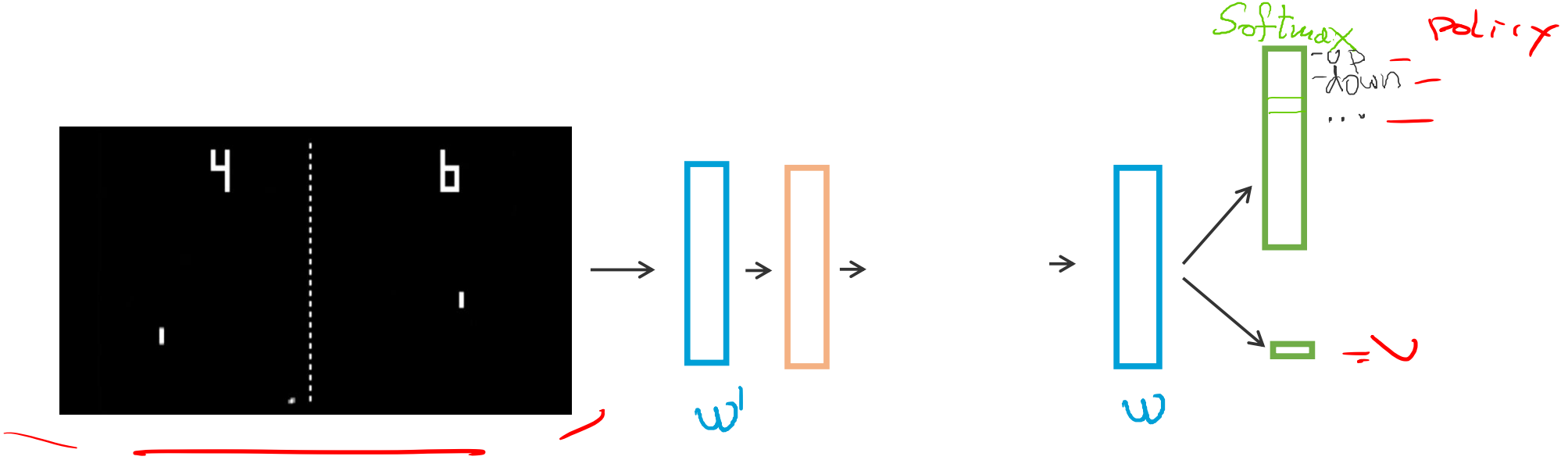


Эффективность метода принципиально зависит от эффективности P и V

http://www.cameronius.com/research/mcts/about/index.html

MCTS с нейросетями





MCTS в AlphaGoZero

1600

π – финальная фунция выбора хода

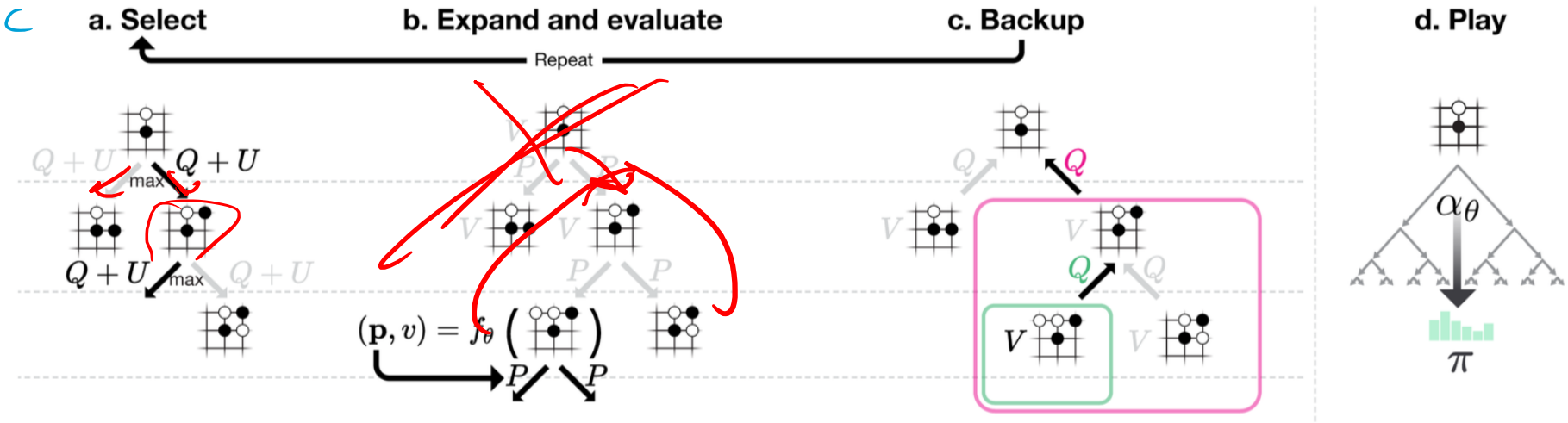
static

У каждого узла:
V – value, который выдала сеть для позиции на доске

У каждого ребра:
P – что выдала policy для хода и позиции на доске
N – количество проходов по ребру

Q - оценка выигрыша хода

dynamic



$$Q = \frac{1}{N} \sum_{descendants} V$$

$$U \approx \frac{P}{1+N}$$

$P_{out}, V_{out} = NN(board)$

Новый узел:
$V = V_{out}$

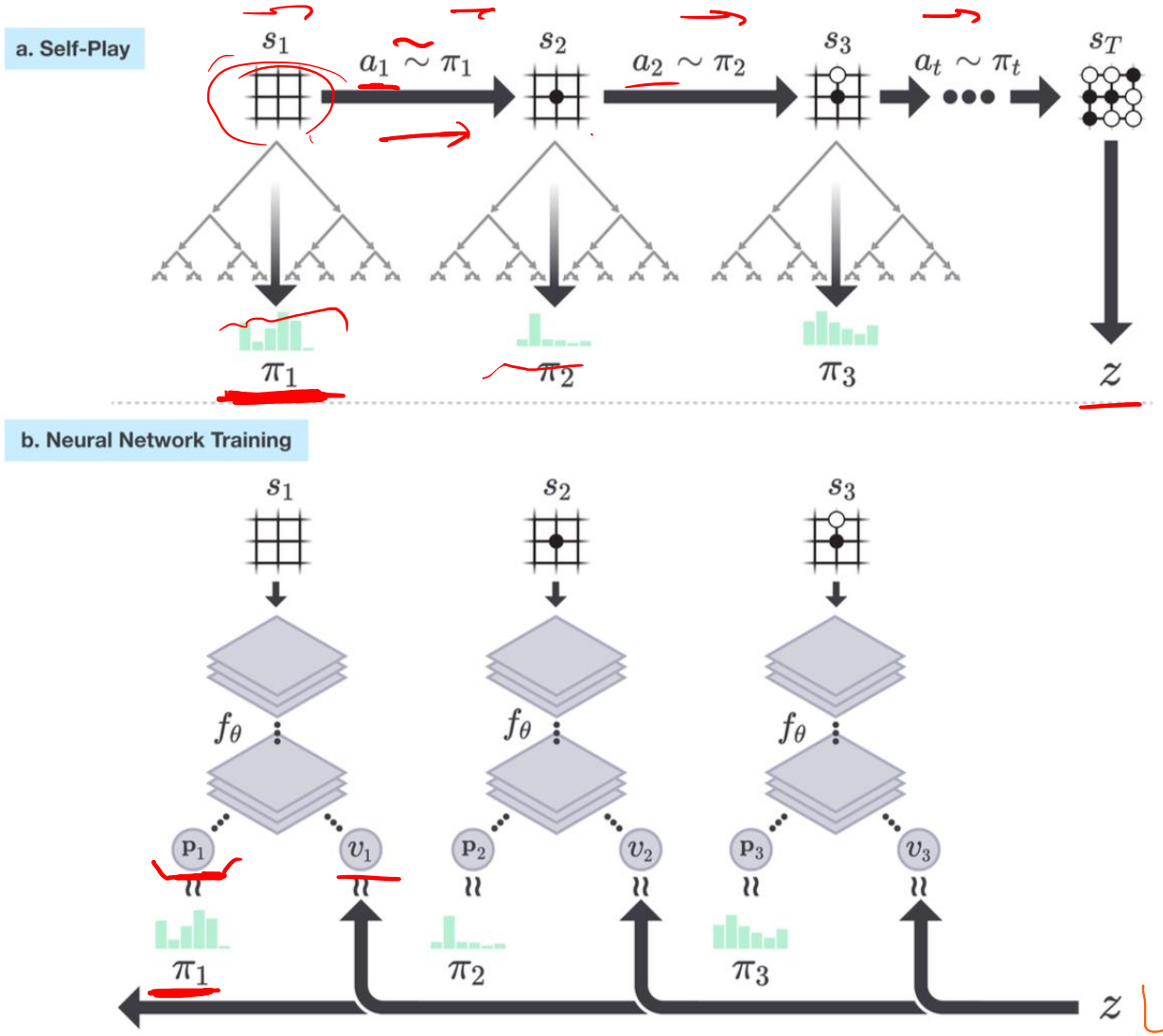
Новые ребра:
$P = P_{out}$
$N = 0$
$Q = 0$

Для всех ребер на пути:
$N_{new} = N + 1$

$$Q_{new} = \frac{1}{N_{new}} \sum_{descendants} V$$

$\pi \sim N$

[Mastering the Game of Go without Human Knowledge'17]



π_i- итоговые вероятности хода для клеток на ходе i после MCTS

z – результат игры, +1 или -1

MCTS генерирует таргет для выхода P
Результат игры генерирует таргет выхода V

MCTS возможен только если есть модель среды!

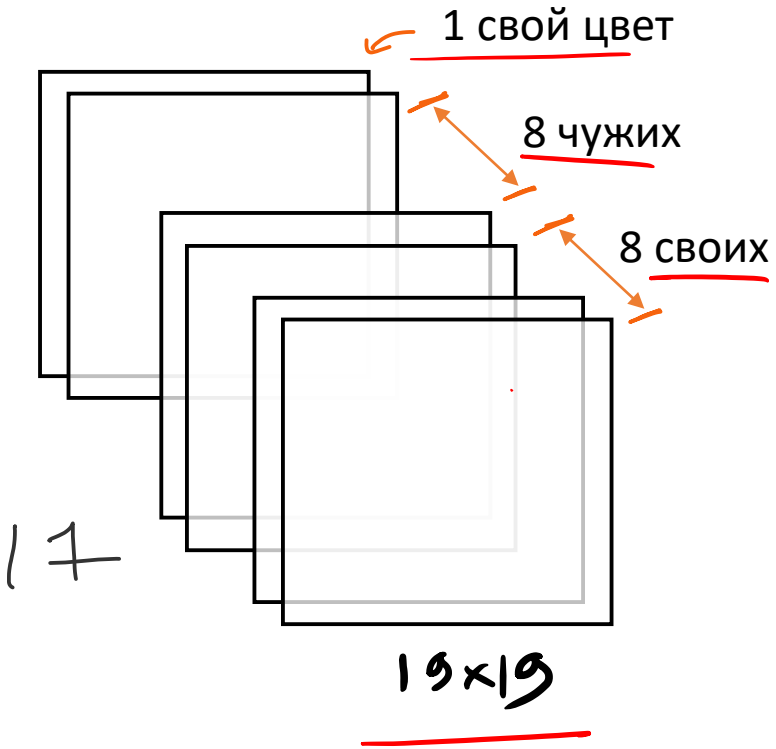
$$(\mathbf{p}, v) = f_\theta(s), \qquad l = (z - v)^2 - \boldsymbol{\pi}^\top \log \mathbf{p} + c||\theta||^2$$

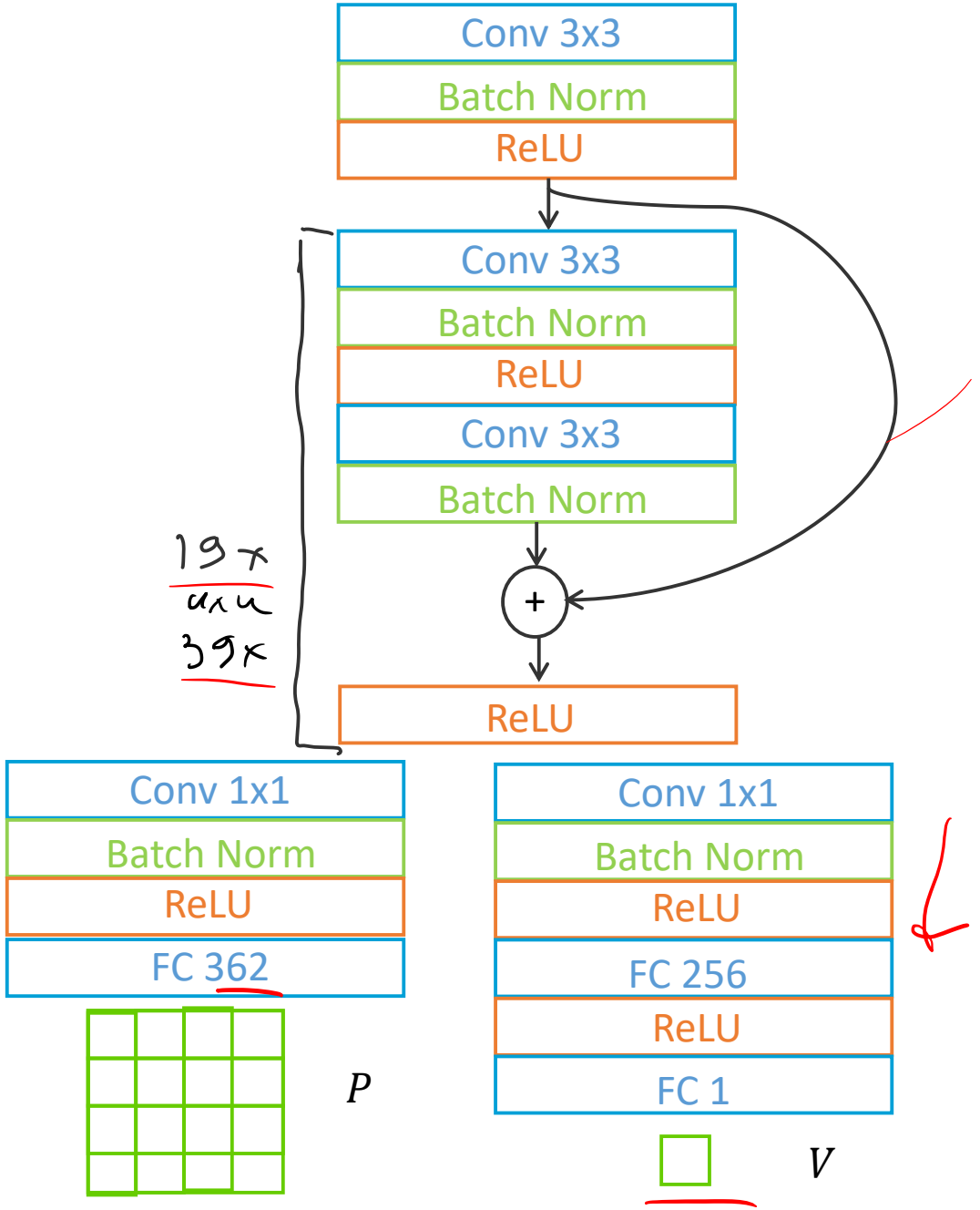
L2 reg

L^2 loss на V

CE loss на P

[Mastering the Game of Go without Human Knowledge'17]

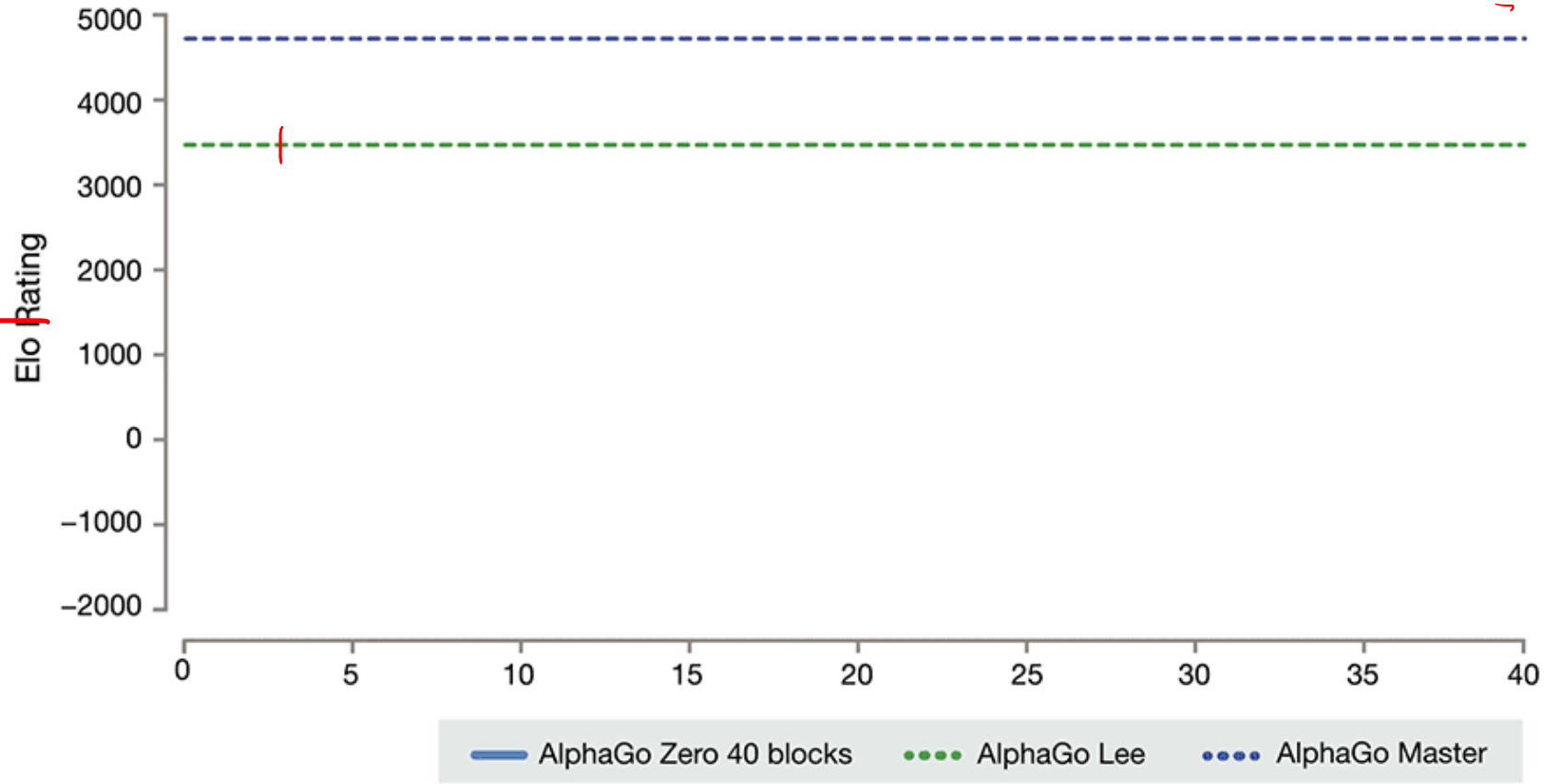




[Mastering the Game of Go without Human Knowledge'17]

AlphaGoZero

- 64 GPUs для тренировки
- 4 TPUs для inference
- Стоимость тренировки: $25M



https://deepmind.com/blog/alphago-zero-learning-scratch/

AlphaZero, шахматы

Chess	
Feature	Planes
P1 piece	6
P2 piece	6
Repetitions	2
Colour	1
Total move count	1
P1 castling	2
P2 castling	2
No-progress count	1
Total	119

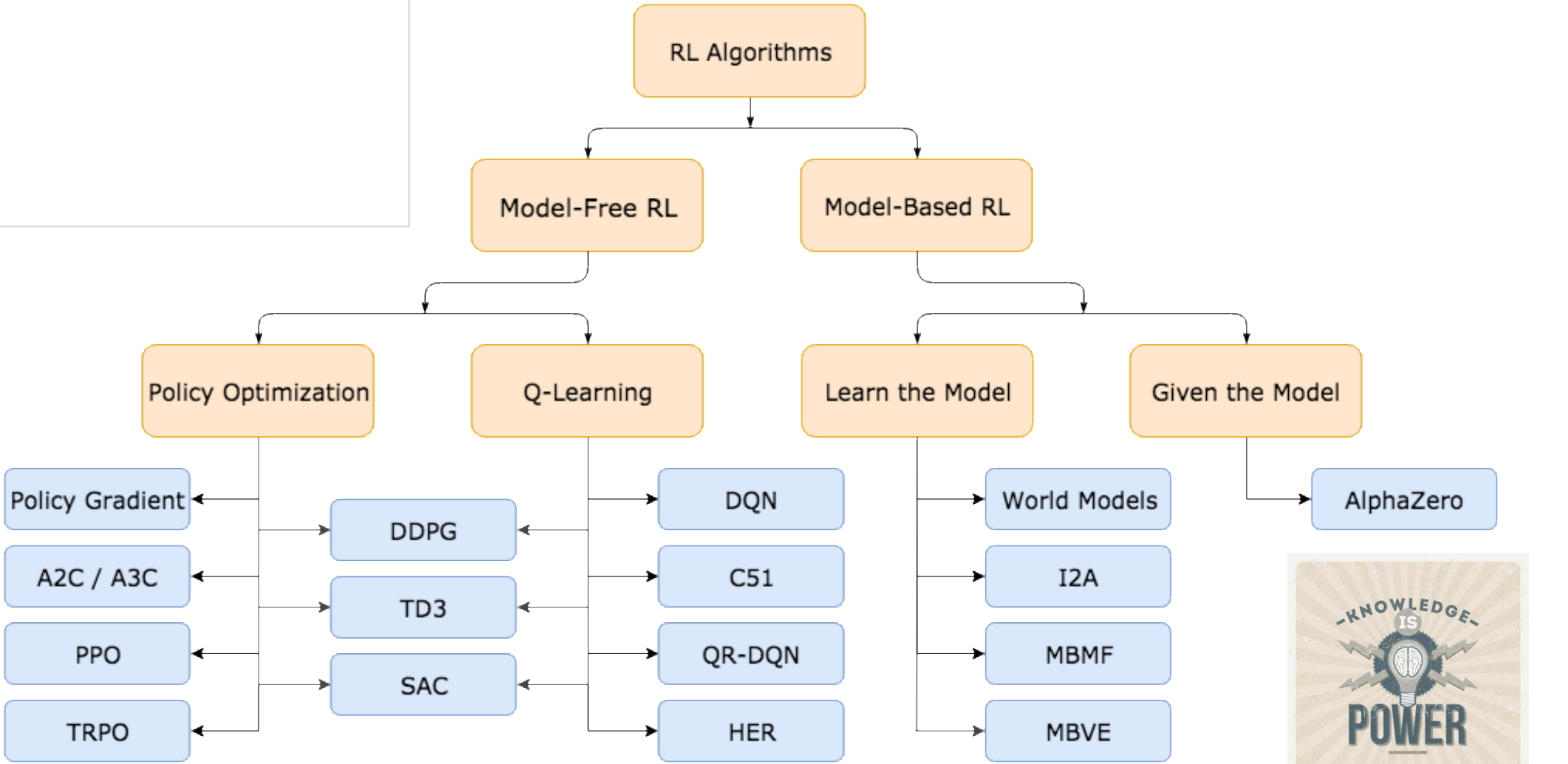
Chess	
Feature	Planes
Queen moves	56
Knight moves	8
Underpromotions	9
Total	73

[Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm'17]

AlphaZero, шахматы
Результаты

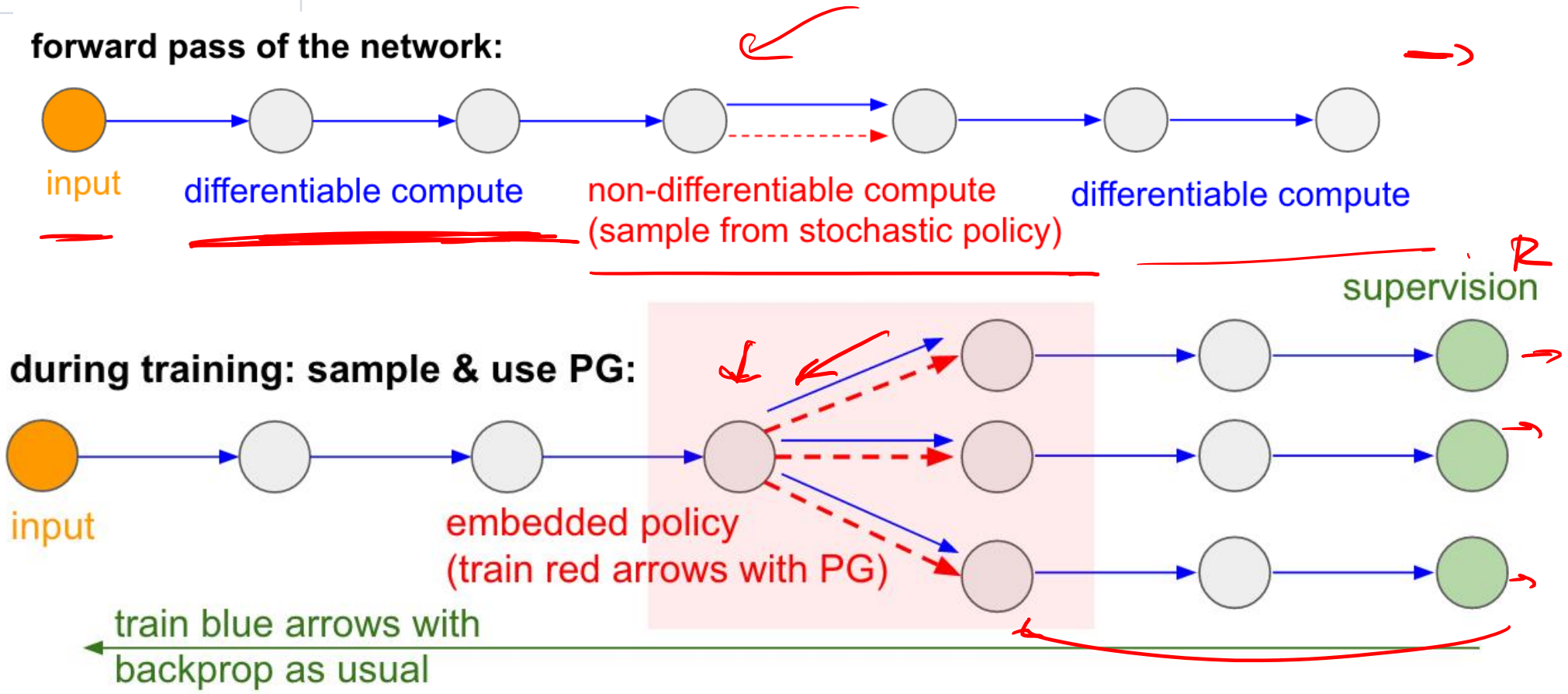
Leela Zero

Game	White	Black	Win	Draw	Loss
Chess	*AlphaZero*	*Stockfish*	25	25	0
	Stockfish	*AlphaZero*	3	47	0


RL Algorithms
Model-Free RL
Model-Based RL
Policy Optimization
Q-Learning
Learn the Model
Given the Model
Policy Gradient
A2C / A3C
PPO
TRPO
DDPG
TD3
SAC
DQN
C51
QR-DQN
HER
World Models
I2A
MBMF
MBVE
AlphaZero
KNOWLEDGE IS POWER


IT'S ALMOST OVER
STAY STRONG
memegenerator.net

RL с другой стороны: недифференцируемые блоки



http://karpathy.github.io/2016/05/31/rl/

Например, раскраска графов

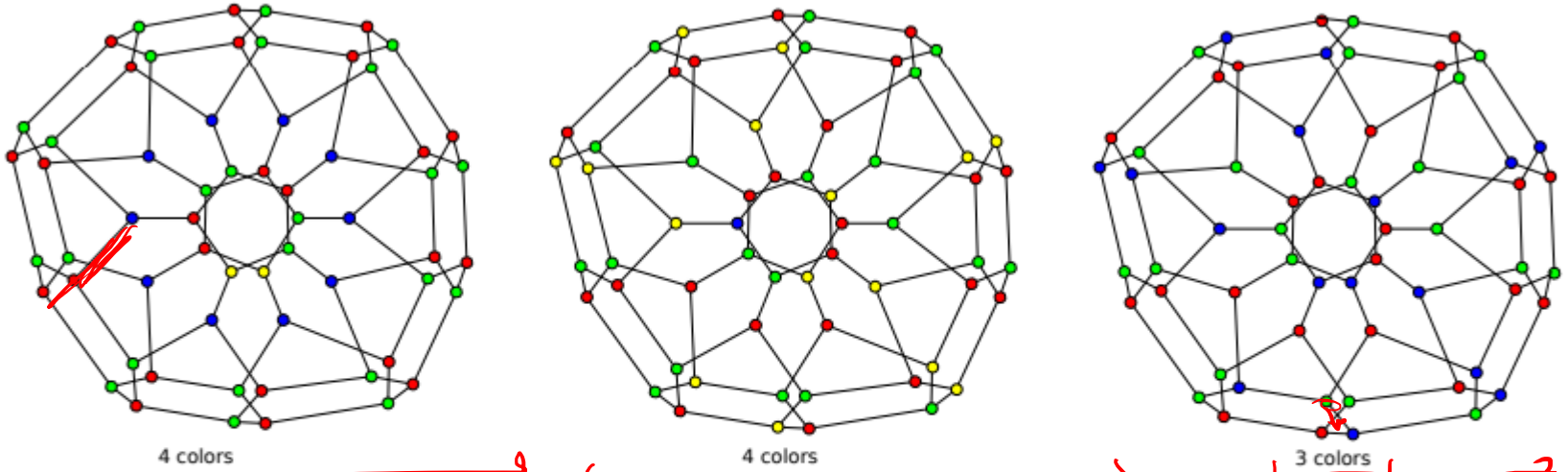


Figure 2. Graph coloring or, more precisely, vertex coloring is a way of assigning colors to the vertices of a graph such that no two adjacent vertices are of the same color. For example, the left and center graphs are colored with greedy heuristics. The right graph is colored optimally. Graph coloring has found many practical applications in diverse domains. The popular game of Sudoku can be

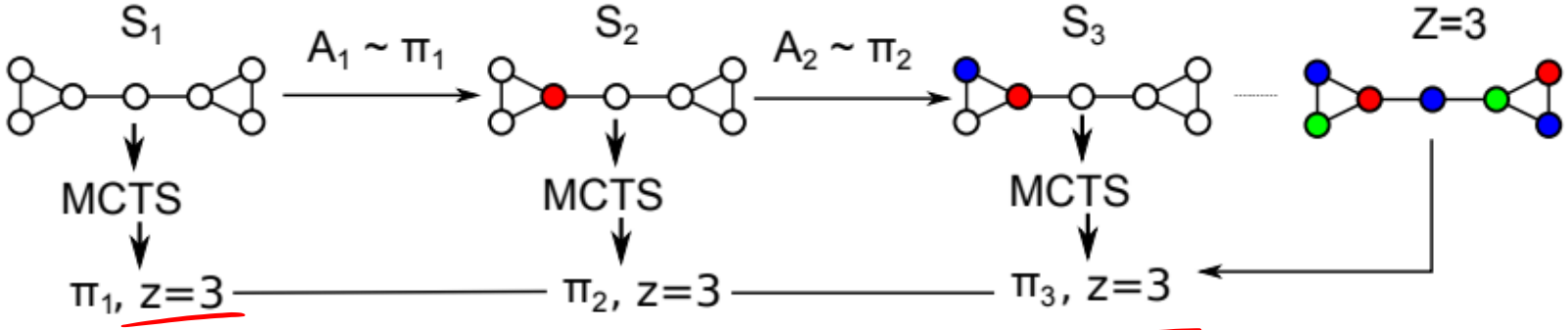
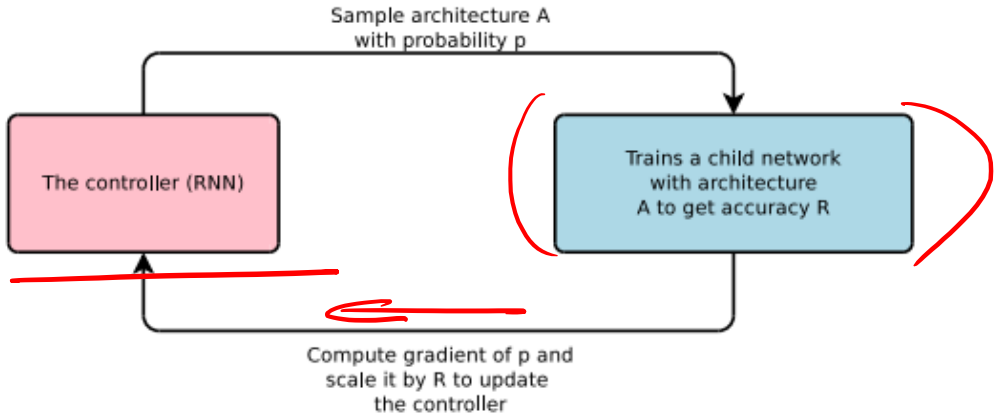


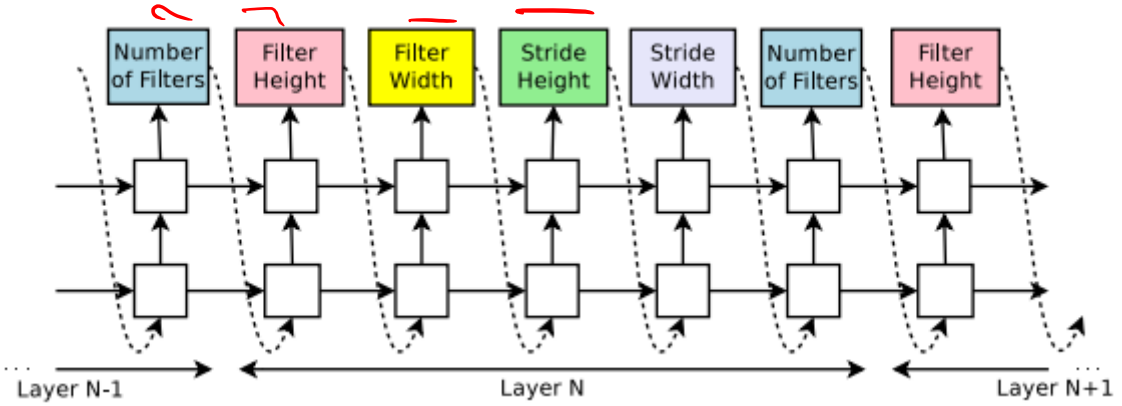
Figure 3. The reinforcement learning algorithm. At each state, a MCTS computes probabilities π_i for the current move, and the next action A_i is selected. When the graph is colored, the final score z is stored as a label for all previous moves.

[Coloring Big Graphs with AlphaGoZero'19]

NASNet

Я такая мета-мета...





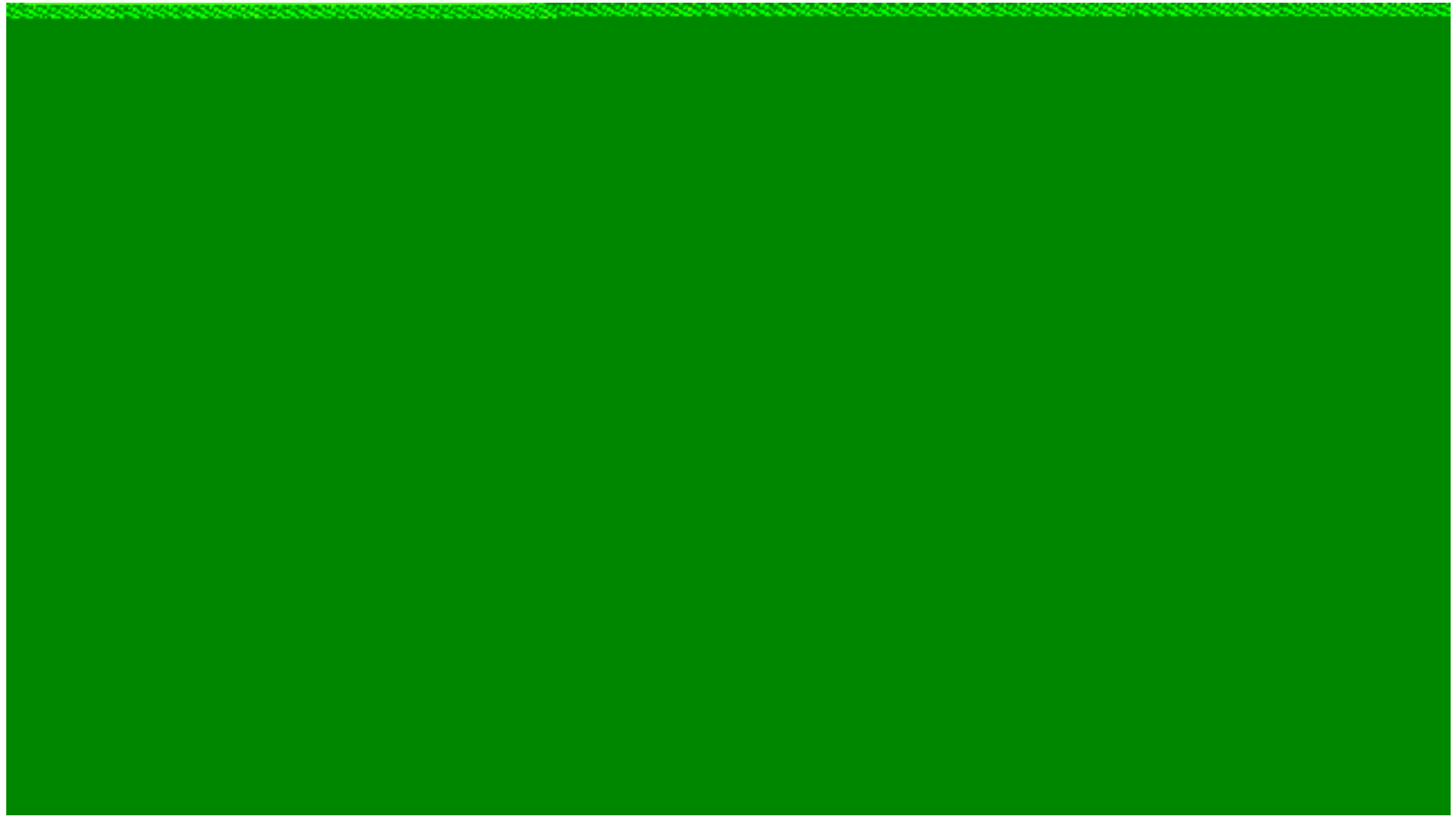
DenseNet ($L = 40, k = 12$) Huang et al. (2016a)	40	1.0M	5.24
DenseNet($L = 100, k = 12$) Huang et al. (2016a)	100	7.0M	4.10
DenseNet ($L = 100, k = 24$) Huang et al. (2016a)	100	27.2M	3.74
DenseNet-BC ($L = 100, k = 40$) Huang et al. (2016b)	190	25.6M	3.46
Neural Architecture Search v1 no stride or pooling	15	4.2M	5.50
Neural Architecture Search v2 predicting strides	20	2.5M	6.01
Neural Architecture Search v3 max pooling	39	7.1M	4.47
Neural Architecture Search v3 max pooling + more filters	39	37.4M	3.65

Table 1: Performance of Neural Architecture Search and other state-of-the-art models on CIFAR-10.

[Neural Architecture Search with Reinforcement Learning'17]


COOL STORY, BRO

PR1 robot, 2008



Youtube

Не то чтобы мы не пробовали…

Personal Robotics: Cloth Grasp Point Detection based on Multiple-View Geometric Cues with Application to Robotic Towel Folding'10

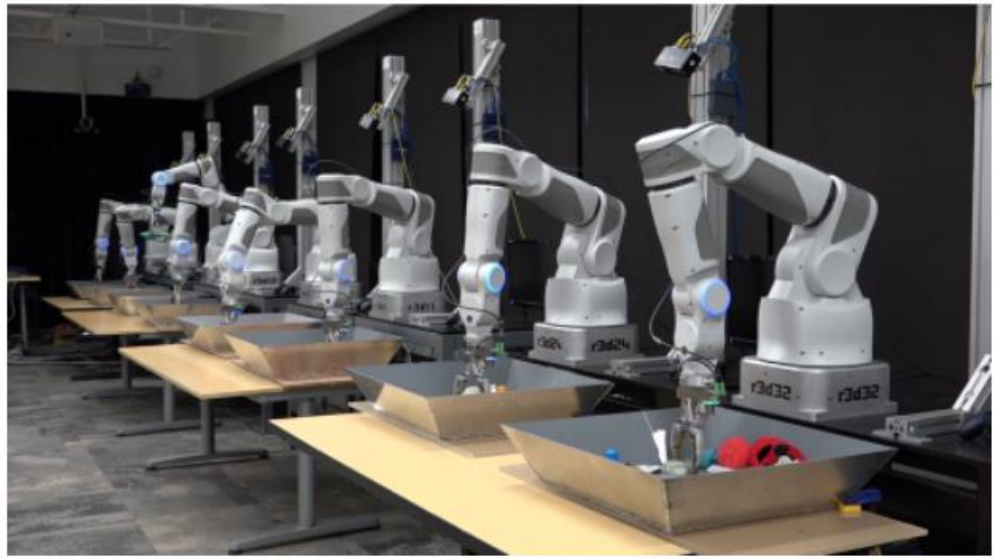


Figure 1. Our large-scale data collection setup, consisting of 14 robotic manipulators. We collected over 800,000 grasp attempts to train the CNN grasp prediction model.

Learning Hand-Eye Coordination for Robotic Grasping with Deep Learning and Large-Scale Data Collection'16

Deep Reinforcement Learning Doesn't Work Yet

Feb 14, 2018

Link

Reinforcement Learning never worked, and 'deep' only helped a bit.

Link

FEBRUARY 23, 2018

DesertFlow 22 января 2019 в 02:25

Что не так с обучением с подкреплением (Reinforcement Learning)?

Link

Машинное обучение, Искусственный интеллект

Проблемы Deep RL

(если я не ошибся в расчетах)

DQN on Atari: 5M шагов, ~100 часов игрового времени

Playing Atari with Deep Reinforcement Learning'13

- **Нужно очень много запусков и данных**

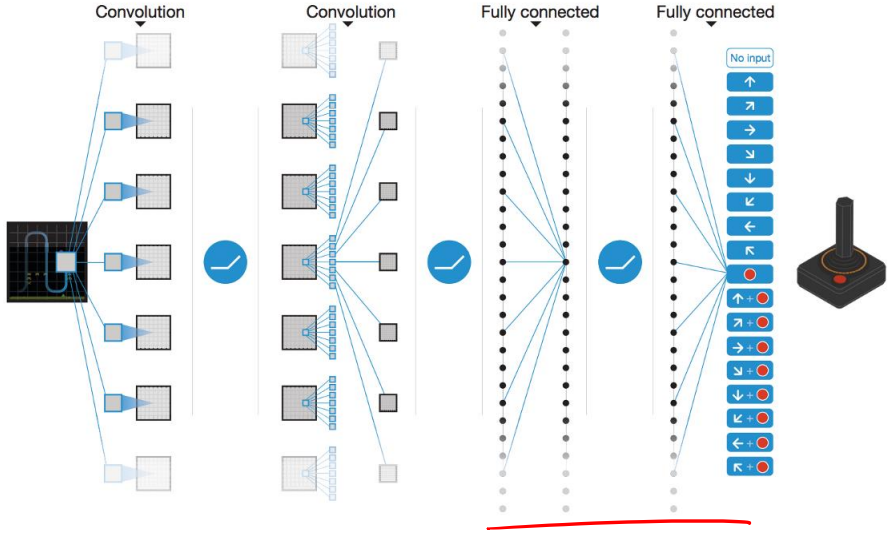
Over the course of training, 4.9 million games of self-play were generated, using 1,600 simulations for each MCTS, which corresponds to approximately 0.4s thinking time per move. Parameters were updated from 700,000 mini-batches of 2,048 positions. The neural network contained 20 residual blocks (see Methods for further details).

Mastering the Game of Go without Human Knowledge'17

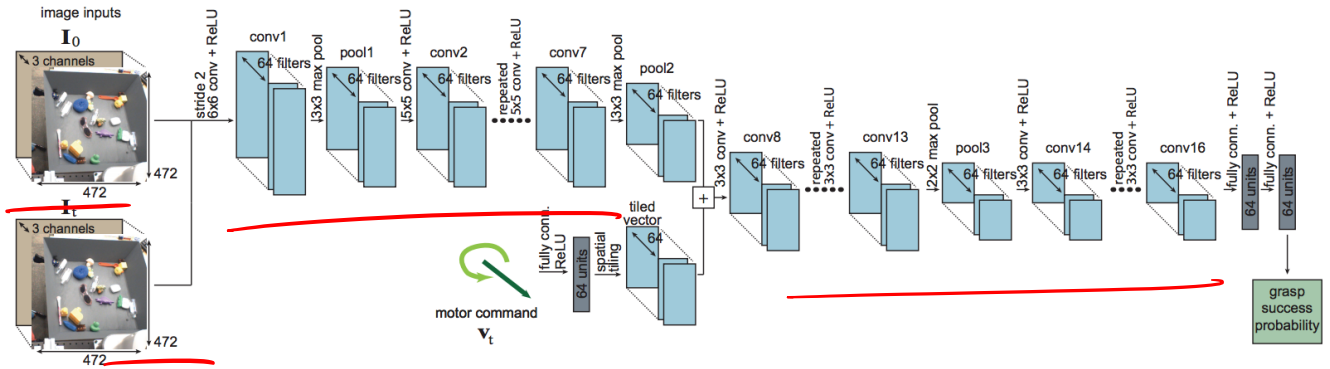
Проблемы Deep RL

- Нужно очень много запусков и данных
- **Получается тренировать не очень большие сети**

DQN

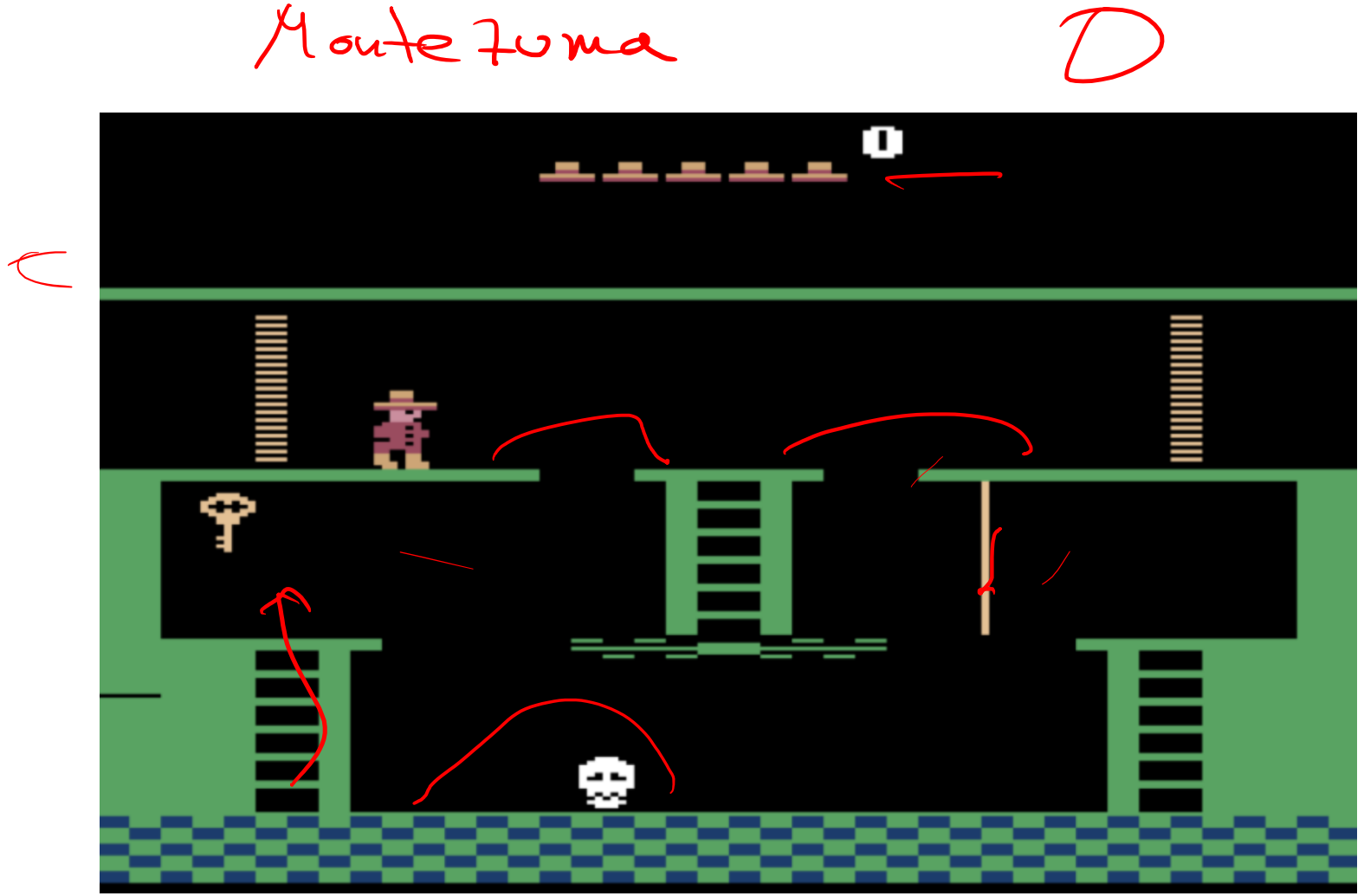


Human-level control through deep reinforcement Learning'15



Learning Hand-Eye Coordination for Robotic Grasping with Deep Learning and Large-Scale Data Collection'16

Проблемы Deep RL



- Нужно очень много запусков и данных
- Получается тренировать не очень большие сети
- **Разреженные награды (sparse rewards)**

Проблемы Deep RL

- Нужно очень много запусков и данных
- Получается тренировать не очень большие сети
- Разреженные награды (sparse rewards)
- **Reward hacking**

OpenAI blog

Проблемы Deep RL

- Нужно очень много запусков и данных
- Получается тренировать не очень большие сети
- Разреженные награды (sparse rewards)
- Reward hacking
- **Не во всех случаях лучше других алгоритмов**

Abstract

The combination of modern Reinforcement Learning and Deep Learning approaches holds the promise of making significant progress on challenging applications requiring both rich perception and policy-selection. The Arcade Learning Environment (ALE) provides a set of Atari games that represent a useful benchmark set of such applications. A recent breakthrough in combining model-free reinforcement learning with deep learning, called DQN, achieves the best real-time agents thus far. Planning-based approaches achieve far higher scores than the best model-free approaches, but they exploit information that is not available to human players, and they are orders of magnitude slower than needed for real-time play. Our main goal in this work is to build a better real-time Atari game playing agent than DQN. The central idea is to use the slow planning-based agents to provide training data for a deep-learning architecture capable of real-time play. We proposed new agents based on this idea and show that they outperform DQN.

Deep Learning for Real-Time Atari Game Play Using Offline Monte-Carlo Tree Search Planning'14

Parkour Atlas

Проблемы Deep RL

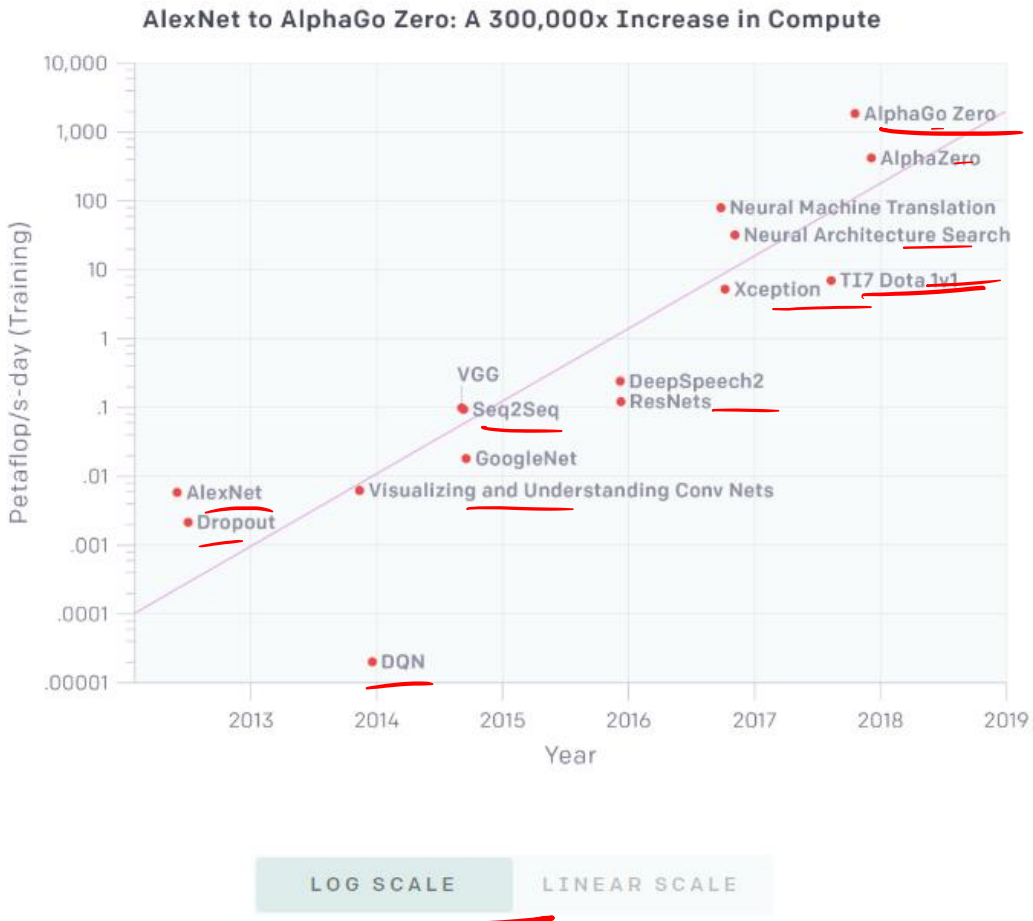
- Нужно очень много запусков и данных
- Получается тренировать не очень большие сети
- Разреженные награды (sparse rewards)
- Reward hacking
- Не во всех случаях лучше других алгоритмов
- **И еще всякое**

И что делать?

- **Больше вычислительной мощи (и данных)!**

Стоимость сейчас: ~$10M
Общий бюджет на железо в мире: ~$1B

Скорее всего, требует возможности симуляции



~10x в год

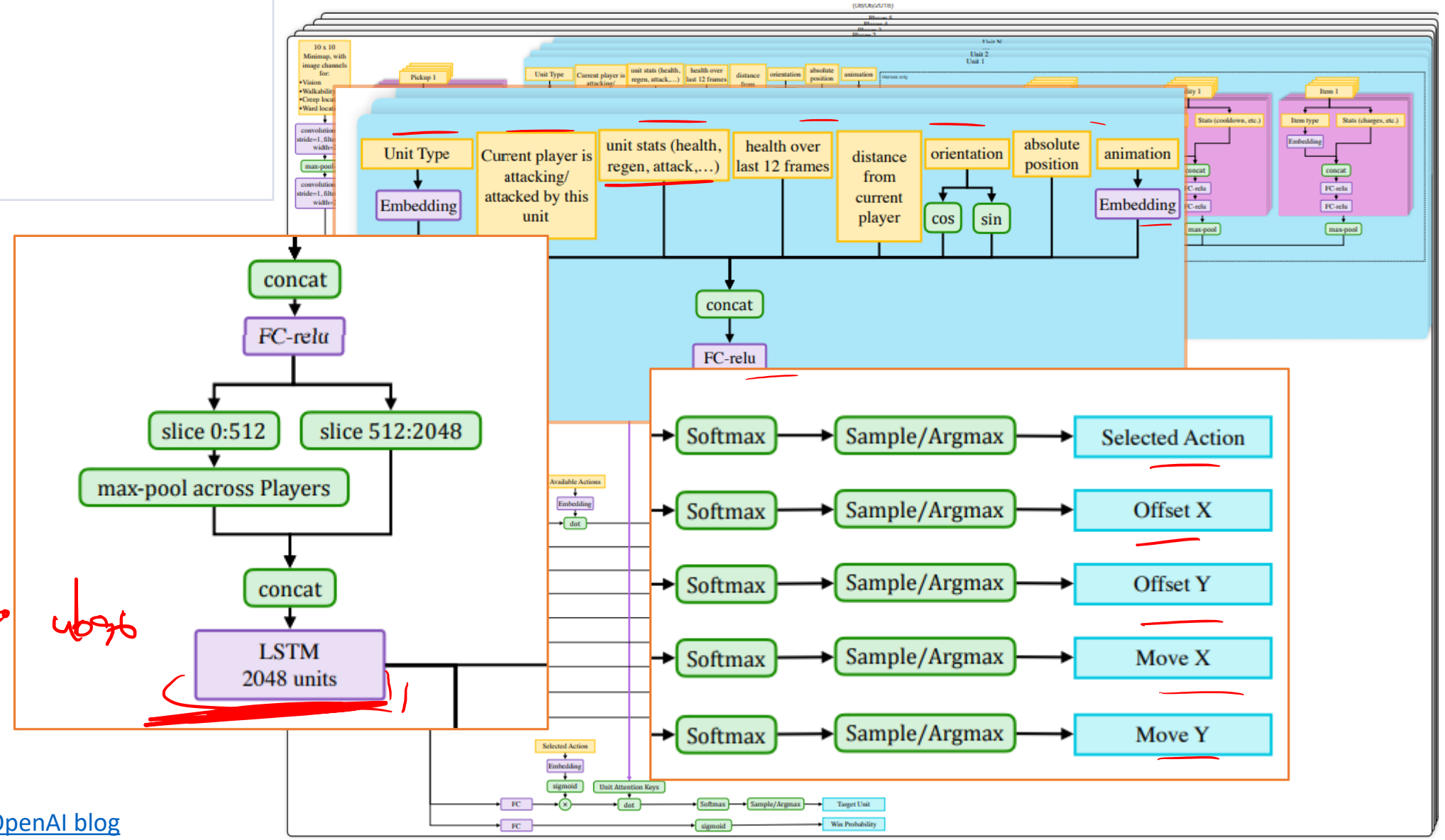
[AI and Compute, OpenAI blog]

Пример: OpenAI Five

OpenAI Five, OpenAI blog

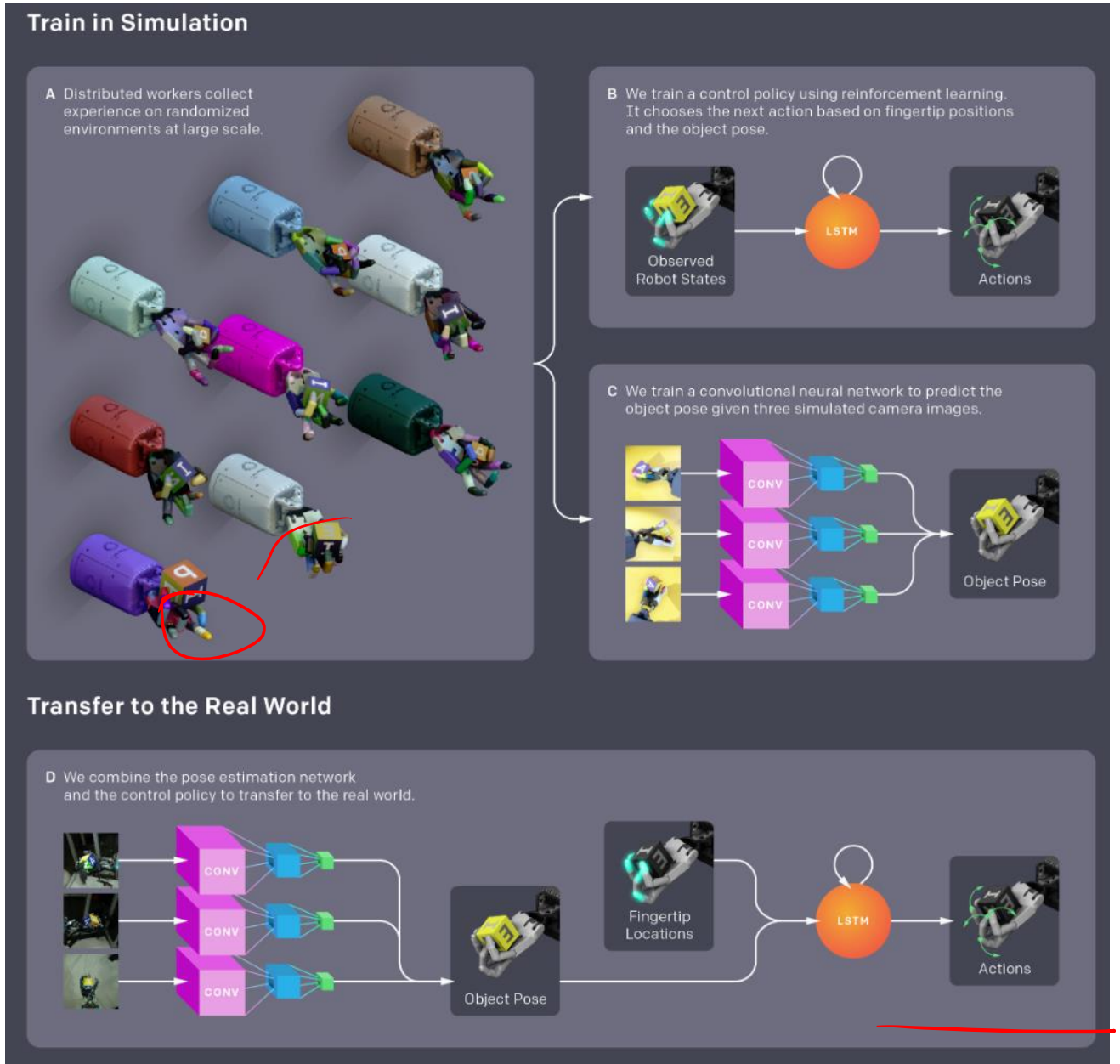
Пример: OpenAI Five

45000 years of
Dota self-play



[How to Train Your OpenAI Five, OpenAI blog]

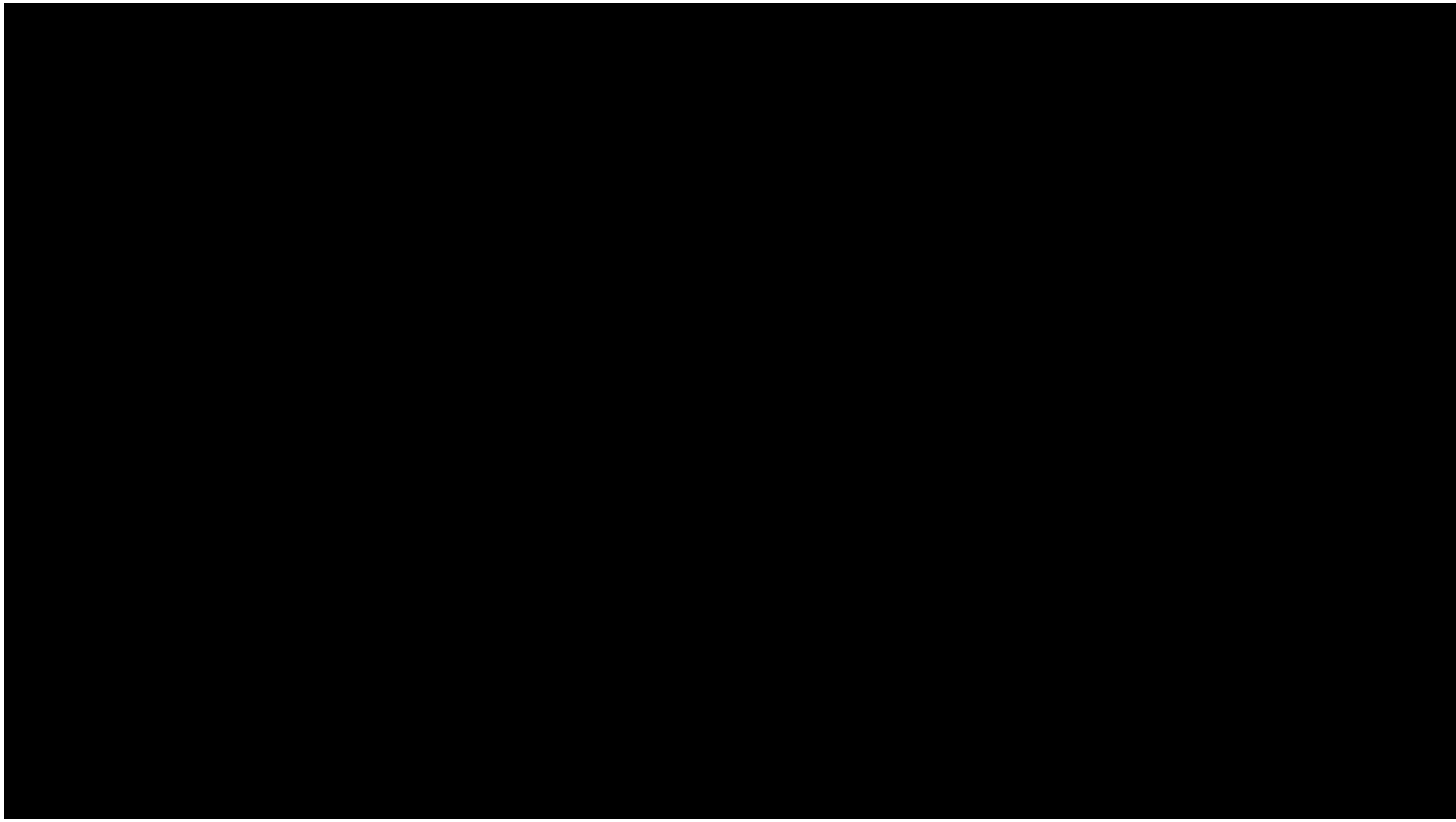
Пример: симуляции физики



[Learning Dexterous In-Hand Manipulation'18]

И что делать?

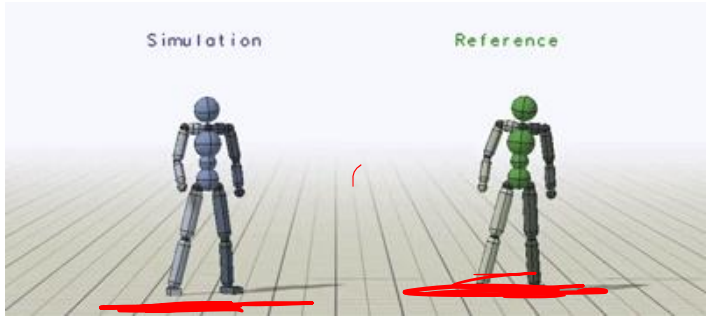
- Больше вычислительной мощи (и данных)!
- **Curiosity (любопытство)**

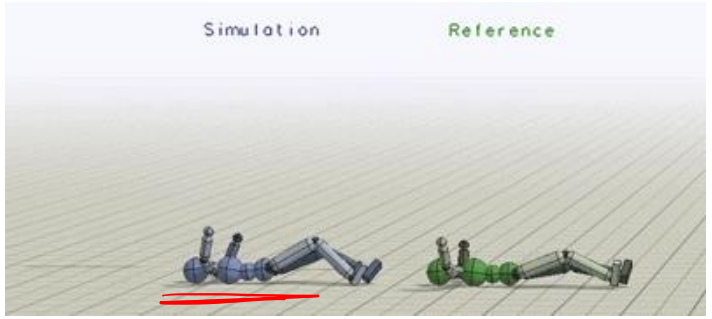


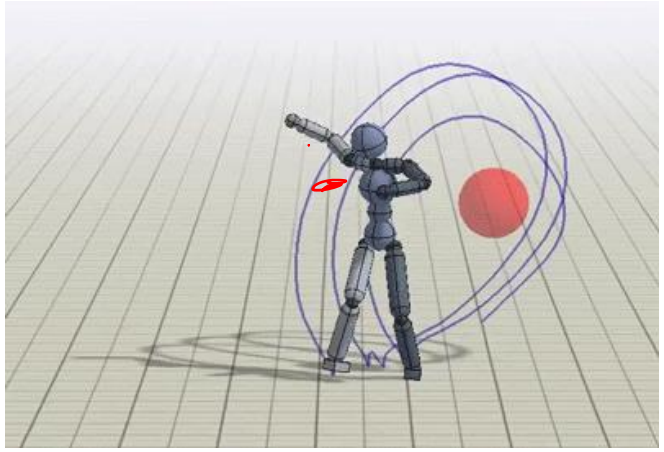
Reinforcement Learning with Prediction-Based Reward, OpenAI Blog

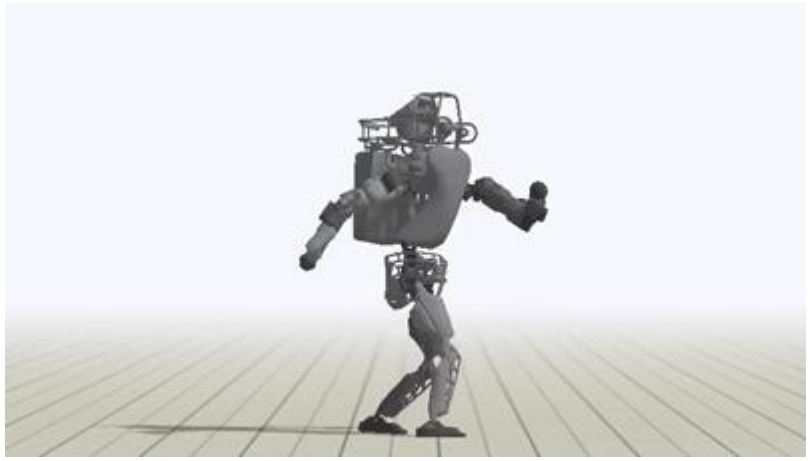
И что делать?

- Больше вычислительной мощи (и данных)!
- Curiosity (любопытство)
- **Imitation learning**









[DeepMimic'18]

И что делать?

RL

- Больше вычислительной мощи (и данных)!
- Curiosity (любопытство)
- Imitation learning
- ????


DLCOURSE.AI
ЗАКОНЧИВШИЕ КУРС
imgflip.com