

# Модификация алгоритма глубинного обучения с подкреплением для модели опорно-двигательного аппарата человека с протезом

Свидченко Олег, ВМР 151

Научный руководитель: Шпильман А. А.

НИУ ВШЭ СПб

# Вступление

- Разработка протезов – дорогой и долгий процесс. Причина в том, что процесс их тестирования очень сложен из-за привлечения людей
- Точные биомеханические симуляции могут быть альтернативным способом тестирования протезов
- Такие симуляции требуют не только точной физической симуляции опорно-двигательной системы человека, но и алгоритм, который будет контролировать такую систему
- Обучение с подкреплением может быть ключом к разработке такого алгоритма

# Обучение с подкреплением



Задача – найти **оптимальную стратегию**  $\pi^*: S \rightarrow A$  которая максимизирует суммарную награду.

# Глубинное обучение с подкреплением

**DQN**<sup>1</sup> приближает суммарную награду при помощи нейронной сети.

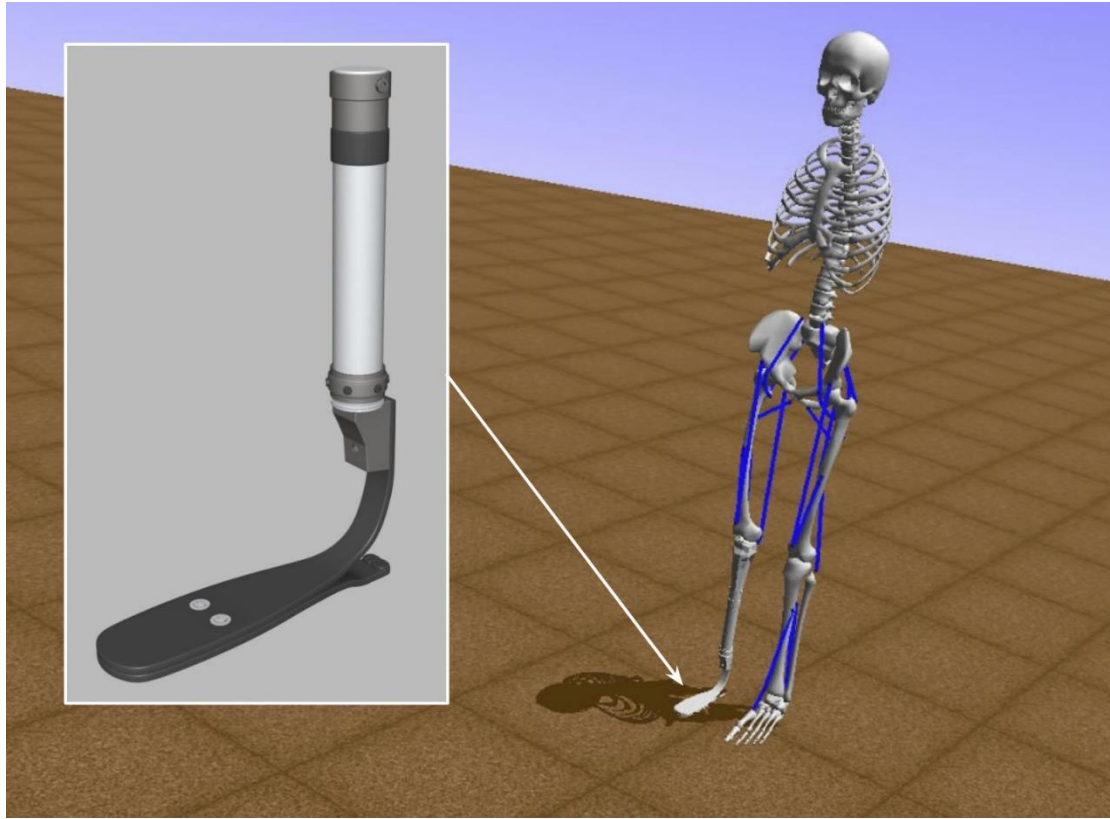
**DDPG**<sup>2</sup> использует еще одну нейронную сеть для получения действий агента.

**TRPO**<sup>3</sup> и **PPO**<sup>4</sup> используют различные подходы для прямого получения стратегии действий.

На данный момент наибольшую эффективность для задач движения показал алгоритм DDPG

1. V. Mnih et al. "Human-level control through deep reinforcement learning"
2. T. P. Lillicrap et al. "Continuous control with deep reinforcement learning"
3. J. Schulman et al. "Trust Region Policy Optimization"
4. J. Schulman et al. "Proximal Policy Optimization Algorithms"

# Симулятор OpenSim<sup>1</sup>



**Эпизод** – 1000 шагов (10 сек.)

**Падение** – таз ниже 60 см.

**Награда за шаг**

$$10 - \left( \|v_{target} - v\| \right)^2 - 0.001 \cdot \|a\|^2$$

# Цель

**Цель** – создать агента обучения с подкреплением, который максимизирует счет в симуляторе OpenSim

## Задачи

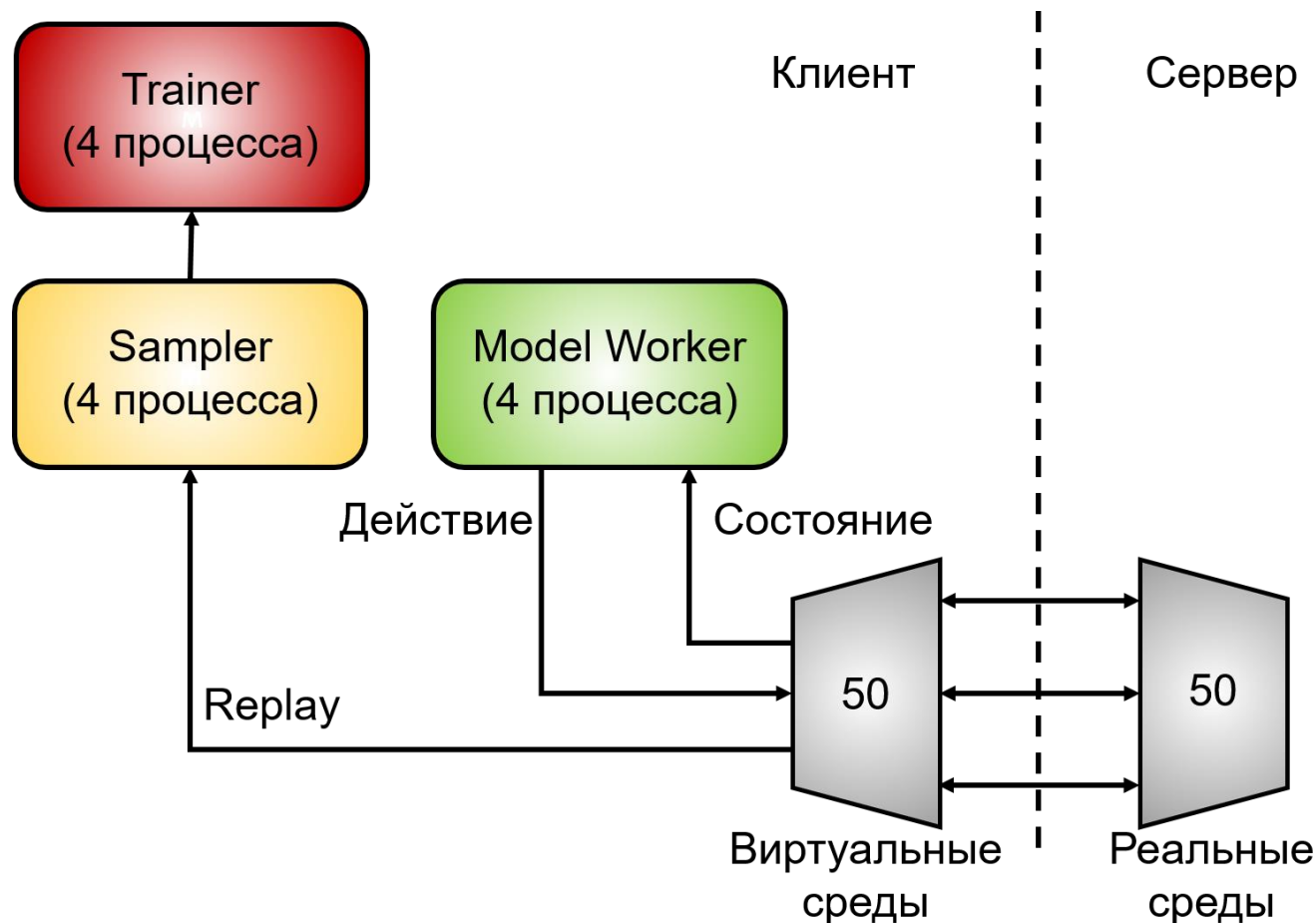
- Разработать фреймворк для распределенного получения данных из среды
- Выбрать и реализовать один из алгоритмов обучения с подкреплением
- Исследовать влияние изменения наблюдаемого агентом состояния и функции награды на скорость сходимости
- Реализовать ансамблирование стратегий агентов
- Провести сравнительное тестирование

# Многопоточное обучение

Симуляция OpenSim  
низкопроизводительная.  
Из-за этого для  
получения данных  
требуется много  
времени

## Решение:

Запустить несколько  
сред параллельно



# Изменение наблюдаемого состояния

Представление состояния в базовой реализации алгоритма:

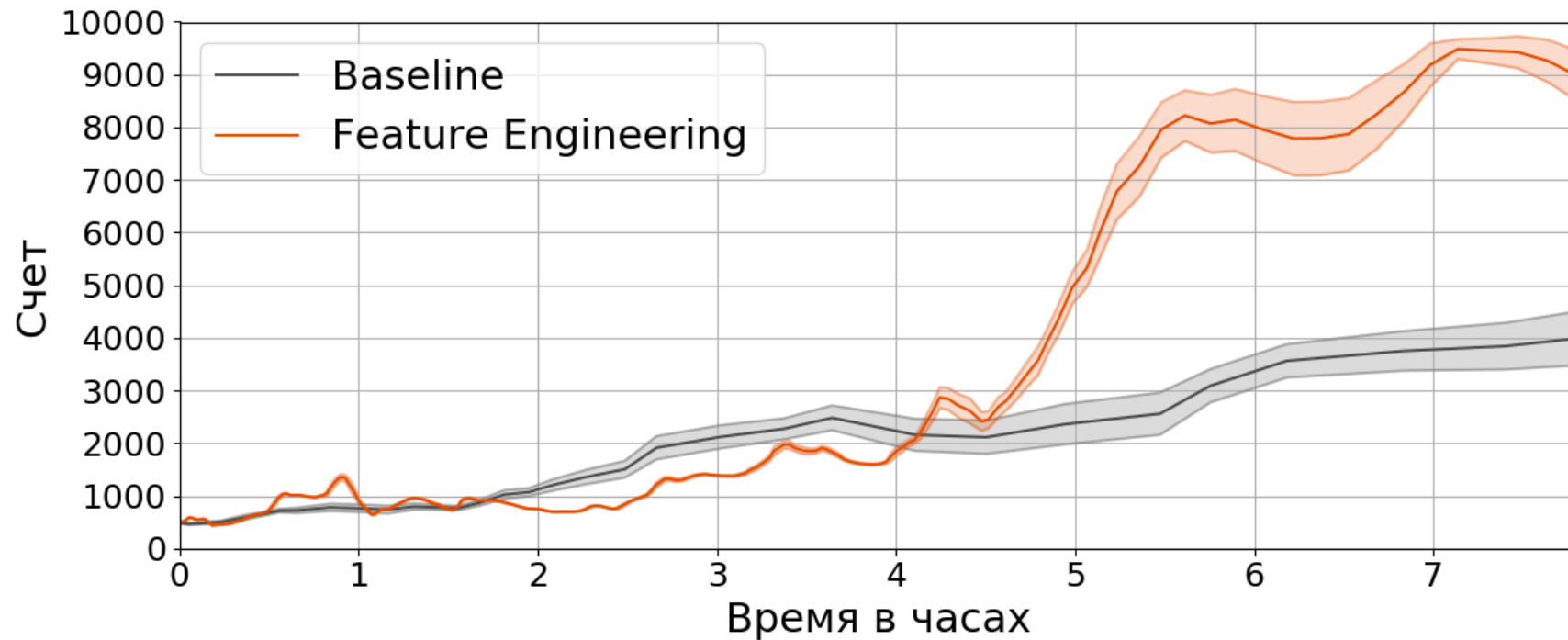
- Относительные координаты частей тела, их скорости и ускорения
- Сила напряжения мышц
- Относительные углы поворота частей тела и угловые скорости

Дополнительные признаки:

- Абсолютные скорости частей тела
- Абсолютные высоты частей тела
- Абсолютные углы поворота частей тела



# Изменение наблюдаемого состояния



**Baseline** – реализация базового алгоритма

**Feature Engineering** – обучение агента с дополнительными признаками

# Трехэтапное обучение и ансамблирование

**Первый этап** – обучение агента с наградой

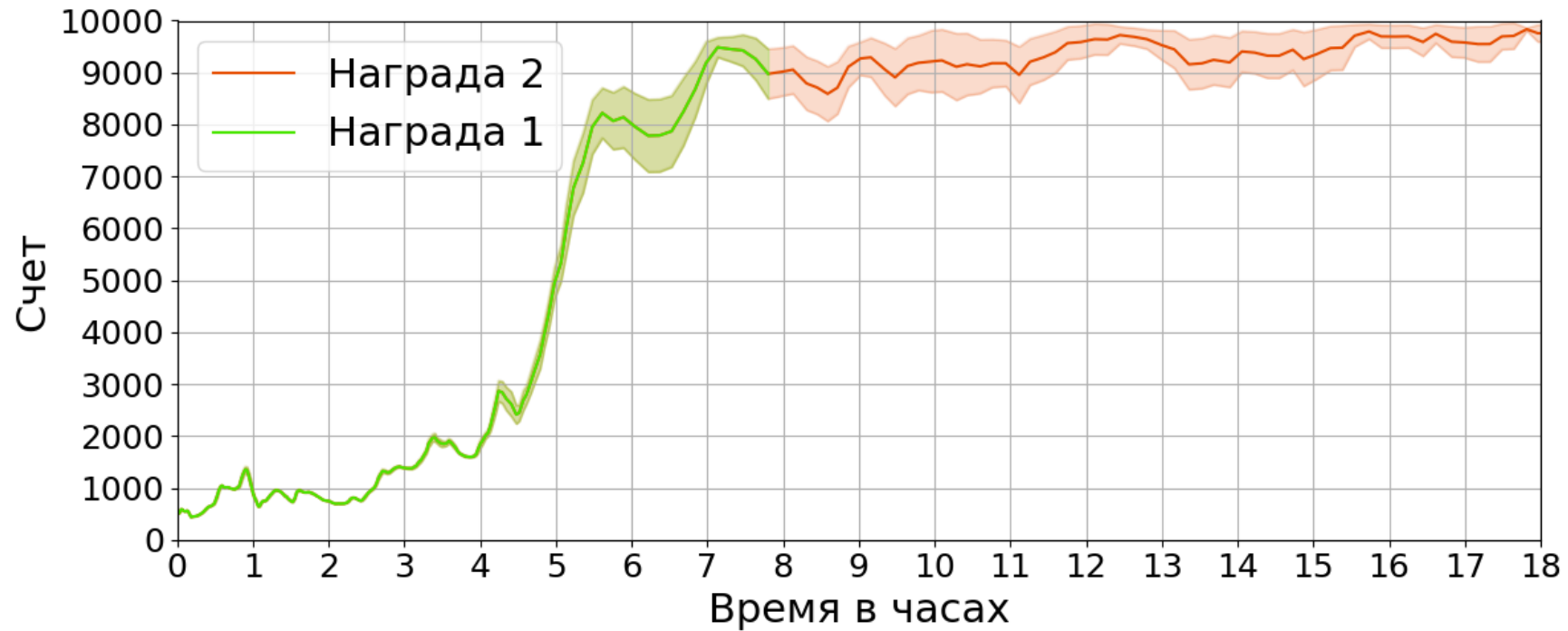
$$r_1 = 1 - \frac{\|v_{target} - v_{current}\|^2}{\|v_{target}\|}$$

**Второй этап** – дообучение агента с наградой

$$r_2 = \begin{cases} 2 \cdot r_{orig} - 19 & \text{Если } r_{orig} \in [9, 10] \\ -1 & \text{иначе} \end{cases}$$

**Третий этап** – получение нескольких стратегий агента при помощи SGDR для построения ансамбля

# Трехэтапное обучение



**Награда 1** – обучение с наградой 1  $-\frac{\|v_{target} - v_{current}\|^2}{\|v_{target}\|}$

**Награда 2** – дообучение с наградой  $\begin{cases} 2 \cdot r - 19 & \text{Если } r \in [9, 10] \\ -1 & \text{иначе} \end{cases}$

# Ансамблирование

| Описание               | Счет                                 | Частота падений |
|------------------------|--------------------------------------|-----------------|
| DDPG                   | $4041.10 \pm 539.23$                 | N/A             |
| DDPG + FE              | $8354.70 \pm 439.3$                  | 0.36            |
| DDPG + FE + En         | $9673.75 \pm 75.03$                  | <b>0.02</b>     |
| DDPG + RS + FE         | $9097.65 \pm 344$                    | 0.21            |
| DDPG + RS + FE + En    | <b><math>9846.72 \pm 29.6</math></b> | <b>0.02</b>     |
| Теоретический максимум | 10000                                | 0.0             |

Локальное тестирование на большом числе эпизодов

# Результаты

- Реализован фреймворк для распределенного обучения
- Реализован алгоритм DDPG с необходимыми модификациями
- Повышена эффективность обучения агента при помощи модификации наблюдаемого состояния и функции награды
- Реализован метод ансамблирования, значительно повышающий производительность итоговой стратегии агента

## **Кроме того:**

- Занято 6 место<sup>1</sup> в соревновании NeurIPS 2018: AI for Prosthetics Challenge

1. <https://www.crowdai.org/challenges/neurips-2018-ai-for-prosthetics-challenge/leaderboards>