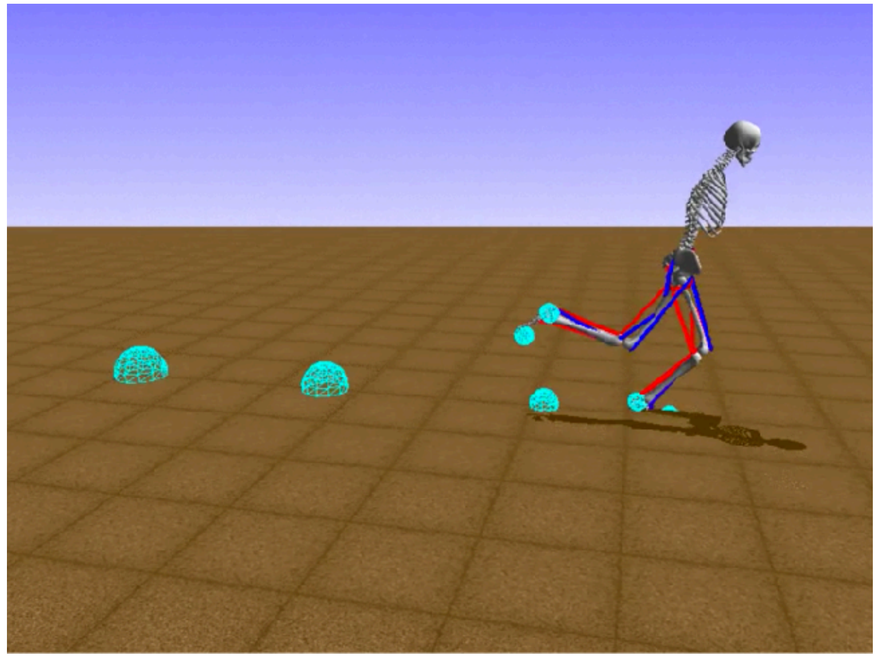

Яндекс

Обучение модели человека бегу (NIPS 2017: Learning to Run)

https://www.crowdai.org/challenges/nips-2017-learning-to-run

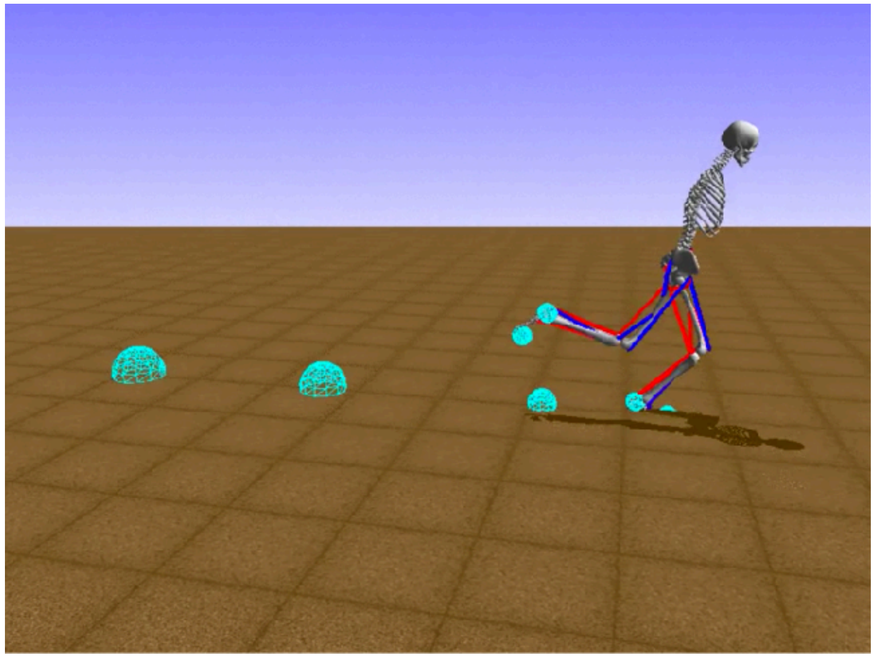
Антон Печенко

Симулятор



Симулятор

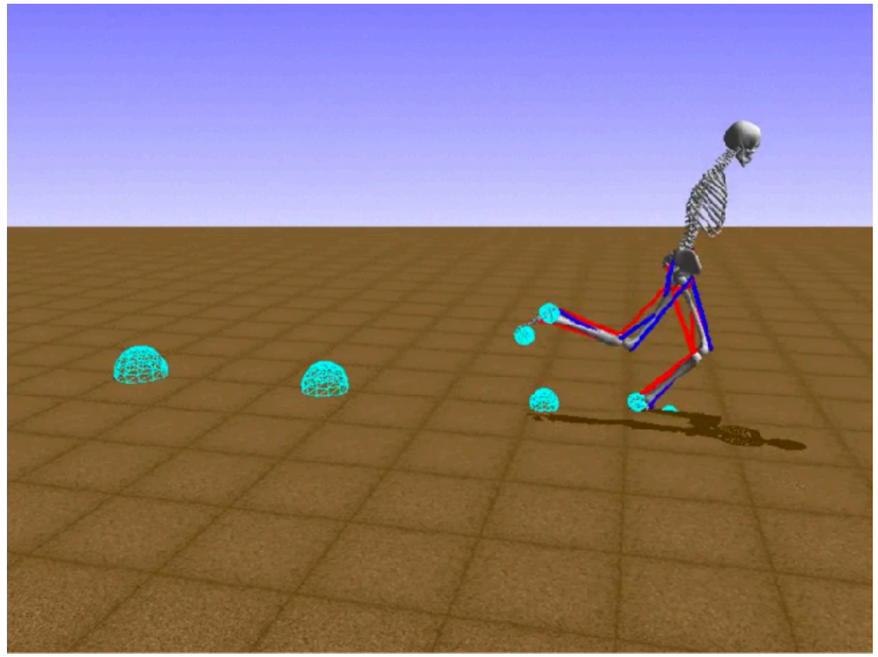
〉 Действие: 18 непрерывных значений (мускулы постепенно меняют усилие, значение применяется не мгновенно)



Симулятор

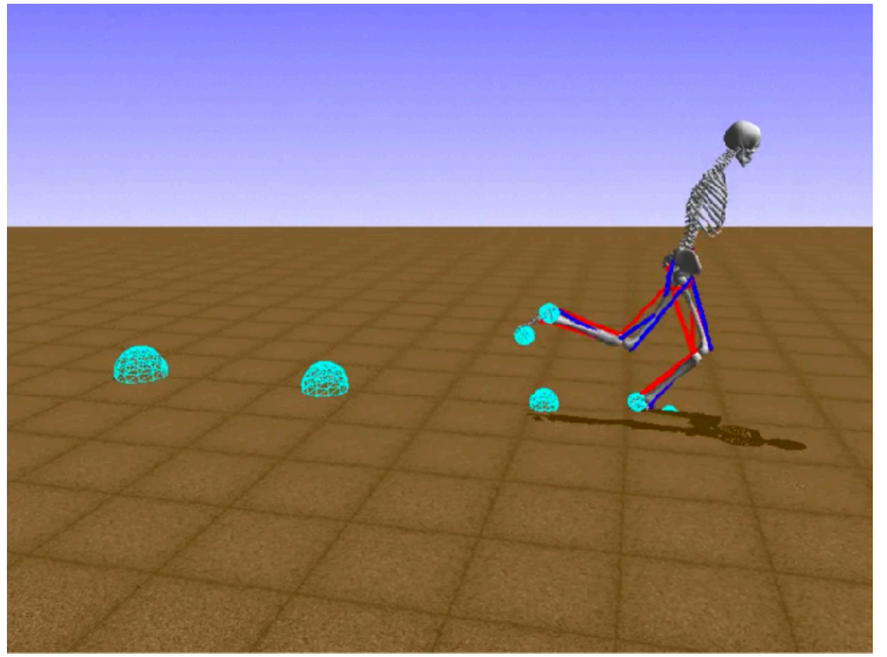
〉 Действие: 18 непрерывных значений (мускулы постепенно меняют усилие, значение применяется не мгновенно)

〉 Наблюдение: 41 непрерывное значение (положение, ориентация, линейные и угловые скорости костей, силы поясничных мышц, расстояние до первого препятствия, сенсоры на ступнях отсутствуют)



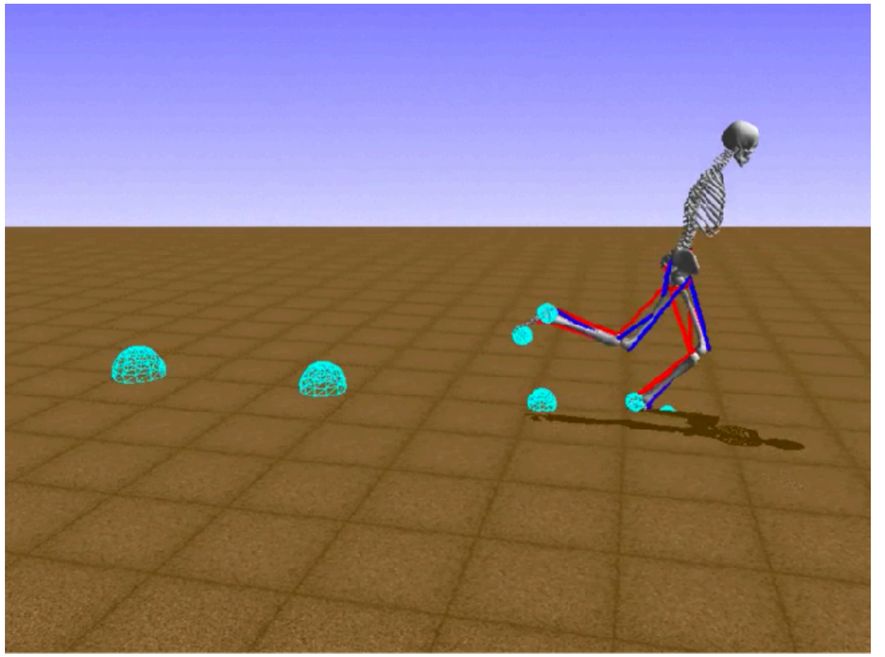
Симулятор

- Действие: 18 непрерывных значений (мускулы постепенно меняют усилие, значение применяется не мгновенно)
- Наблюдение: 41 непрерывное значение (положение, ориентация, линейные и угловые скорости костей, силы поясничных мышц, расстояние до первого препятствия, сенсоры на ступнях отсутствуют)
- Награда: скорость по x минус напряжение в суставах (таки сенсоры есть, но внутри симулятора, а не в наблюдении)



Симулятор

- Действие: 18 непрерывных значений (мускулы постепенно меняют усилие, значение применяется не мгновенно)
- Наблюдение: 41 непрерывное значение (положение, ориентация, линейные и угловые скорости костей, силы поясничных мышц, расстояние до первого препятствия, сенсоры на ступнях отсутствуют)
- Награда: скорость по x минус напряжение в суставах (таки сенсоры есть, но внутри симулятора, а не в наблюдении)
- ОООООчень медленный, чем лучше агент, тем медленнее симулятор



Состояние

Состояние

- Вид от первого лица (относительно тазовой кости :)

Состояние

- Вид от первого лица (относительно тазовой кости :)
- Вычесть угол и координаты тазовой кости

Состояние

- Вид от первого лица (относительно тазовой кости :)
- Вычесть угол и координаты тазовой кости
- Занулить координату X тазовой кости

Состояние

- Вид от первого лица (относительно тазовой кости :)
- Вычесть угол и координаты тазовой кости
- Занулить координату X тазовой кости
- Для правильного предсказания давления на стопу использовать несколько подряд идущих измерений

Алгоритм

Deep Deterministic Policy Gradient arXiv:1509.02971

$$Q^{\mu}(s_t, a_t) = \mathbb{E}_{r_t, s_{t+1} \sim E}\left[r(s_t, a_t) + \gamma Q^{\mu}(s_{t+1}, \mu(s_{t+1}))\right]$$

$$L(\theta^Q) = \mathbb{E}_{s_t \sim \rho^{\beta}, a_t \sim \beta, r_t \sim E}\left[\left(Q(s_t, a_t|\theta^Q) - y_t\right)^2\right]$$

$$y_t = r(s_t, a_t) + \gamma Q(s_{t+1}, \mu(s_{t+1})|\theta^Q).$$

$$\begin{aligned}\nabla_{\theta^{\mu}} J &\approx \mathbb{E}_{s_t \sim \rho^{\beta}}\left[\nabla_{\theta^{\mu}} Q(s, a|\theta^Q)|_{s=s_t, a=\mu(s_t|\theta^{\mu})}\right] \\ &= \mathbb{E}_{s_t \sim \rho^{\beta}}\left[\nabla_a Q(s, a|\theta^Q)|_{s=s_t, a=\mu(s_t)} \nabla_{\theta_{\mu}} \mu(s|\theta^{\mu})|_{s=s_t}\right]\end{aligned}$$

Детали и трюки

Детали и трюки

〉 Актор и Критик MLP одинаковой архитектуры 5 слоев по 512 нейронов

Детали и трюки

- Актор и Критик MLP одинаковой архитектуры 5 слоев по 512 нейронов
- Discount factor 0.99, replay buffer 1M, batch size 64

Детали и трюки

- Актор и Критик MLP одинаковой архитектуры 5 слоев по 512 нейронов
- Discount factor 0.99, replay buffer 1M, batch size 64
- Начальная оптимизация Adam 1e-4, нет штрафа за падение

Детали и трюки

- Актор и Критик MLP одинаковой архитектуры 5 слоев по 512 нейронов
- Discount factor 0.99, replay buffer 1M, batch size 64
- Начальная оптимизация Adam 1e-4, нет штрафа за падение
- Тонкая настройка SGD 5e-5, штраф за падение -1

Детали и трюки

- Актор и Критик MLP одинаковой архитектуры 5 слоев по 512 нейронов
- Discount factor 0.99, replay buffer 1M, batch size 64
- Начальная оптимизация Adam 1e-4, нет штрафа за падение
- Тонкая настройка SGD 5e-5, штраф за падение -1
- Обучение сильно ограничено CPU из-за медленного симулятора. Облучал через 7 параллельно запущенных симуляторов, GPU Nvidia GTX 1080 ~ 1% нагрузки. При более менее хорошем агенте примерно 200K итераций симулятора в сутки. Неделя на эксперимент.

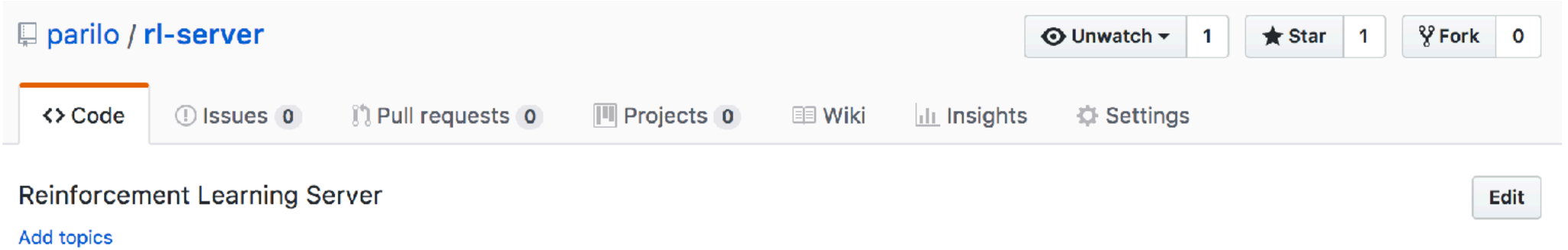
Детали и трюки

- Актор и Критик MLP одинаковой архитектуры 5 слоев по 512 нейронов
- Discount factor 0.99, replay buffer 1M, batch size 64
- Начальная оптимизация Adam 1e-4, нет штрафа за падение
- Тонкая настройка SGD 5e-5, штраф за падение -1
- Обучение сильно ограничено CPU из-за медленного симулятора. Облучал через 7 параллельно запущенных симуляторов, GPU Nvidia GTX 1080 ~ 1% нагрузки. При более менее хорошем агенте примерно 200K итераций симулятора в сутки. Неделя на эксперимент.
- Часть агентов первые 40-20 шагов делали случайные действия и/или помещались в среду пониженного уровня сложности с вероятностью 0.1-0.3. Это нужно чтобы разнообразить буфер опыта, помогает для более быстрого старта

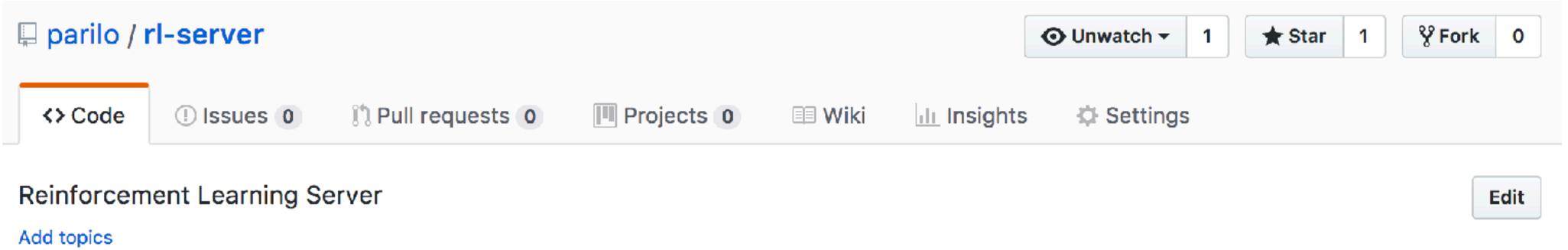
Хаотическая система

```
s = json.loads(json.dumps({’s':state}))['s']
action = sess.run (act,{
    obs: [s]
})[0]
action = np.array(json.loads(json.dumps({'action': action}))[‘action'])
```

Софт

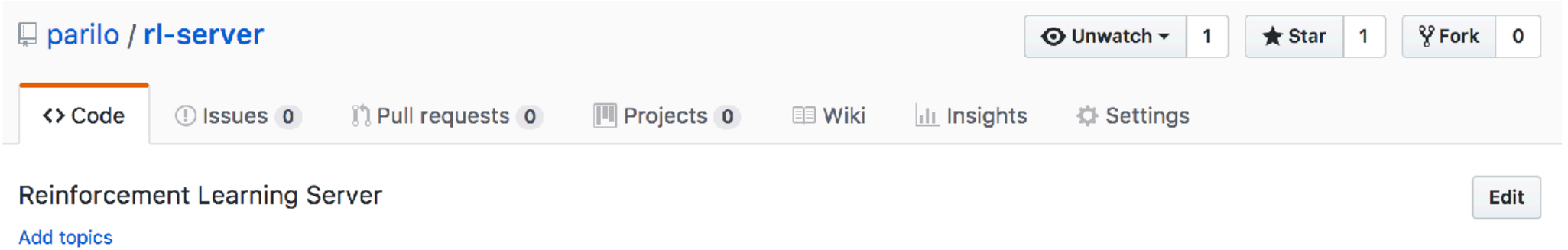

parilo / rl-server
Unwatch 1
Star 1
Fork 0
Code
Issues 0
Pull requests 0
Projects 0
Wiki
Insights
Settings
Reinforcement Learning Server
Edit
Add topics

Софт



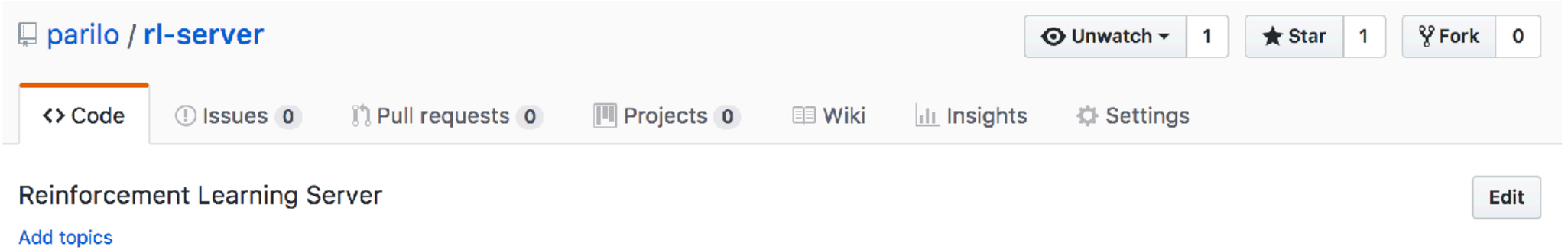
〉 Использует Tensorflow, есть DQN и DDPG

Софт



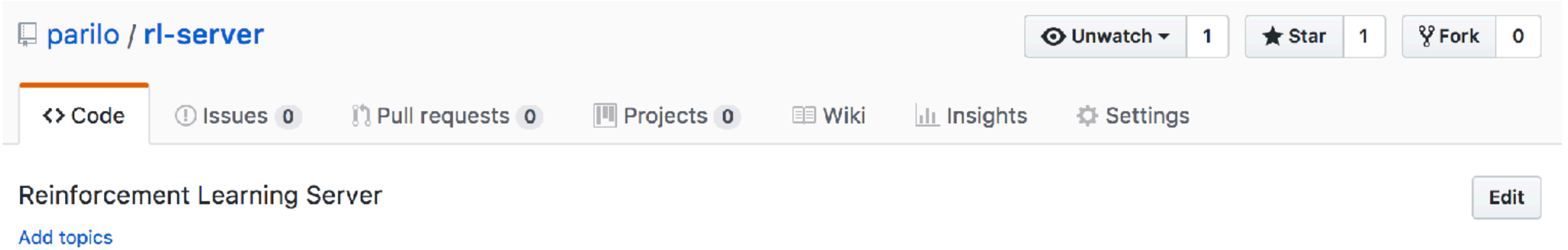
- Использует Tensorflow, есть DQN и DDPG
- Работает в отдельном приложении, вы имеете возможность перезапускать/выключать на время среды без остановки эксперимента

Софт



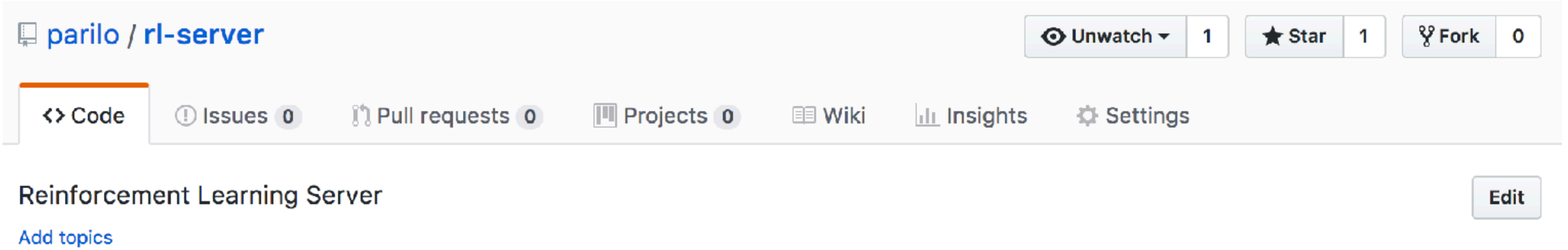
- Использует Tensorflow, есть DQN и DDPG
- Работает в отдельном приложении, вы имеете возможность перезапускать/выключать на время среды без остановки эксперимента
- Легкий клиент на python или C++ (можно легко написать свой) общается с сервером через websocket и json

Софт



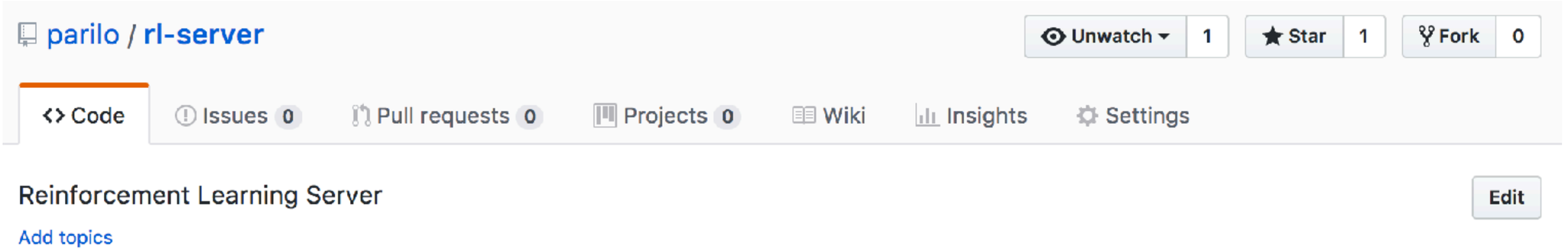
- Использует Tensorflow, есть DQN и DDPG
- Работает в отдельном приложении, вы имеете возможность перезапускать/выключать на время среды без остановки эксперимента
- Легкий клиент на python или C++ (можно легко написать свой) общается с сервером через websocket и json
- Клиент отправляет опыт на сервер и получает оттуда действия которые нужно совершать

Софт



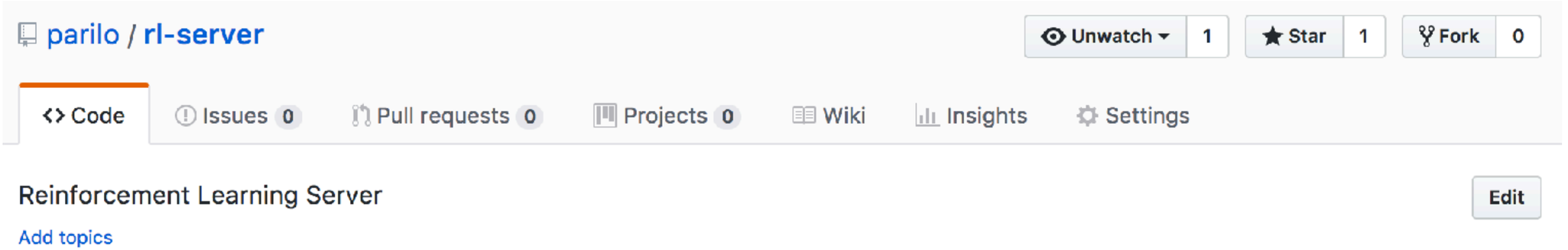
- Использует Tensorflow, есть DQN и DDPG
- Работает в отдельном приложении, вы имеете возможность перезапускать/выключать на время среды без остановки эксперимента
- Легкий клиент на python или C++ (можно легко написать свой) общается с сервером через websocket и json
- Клиент отправляет опыт на сервер и получает оттуда действия которые нужно совершать
- Исправляете баги, тренируете агентов

Софт



- Использует Tensorflow, есть DQN и DDPG
- Работает в отдельном приложении, вы имеете возможность перезапускать/выключать на время среды без остановки эксперимента
- Легкий клиент на python или C++ (можно легко написать свой) общается с сервером через websocket и json
- Клиент отправляет опыт на сервер и получает оттуда действия которые нужно совершать
- Исправляете баги, тренируете агентов
- ...

Софт



- Использует Tensorflow, есть DQN и DDPG
- Работает в отдельном приложении, вы имеете возможность перезапускать/выключать на время среды без остановки эксперимента
- Легкий клиент на python или C++ (можно легко написать свой) общается с сервером через websocket и json
- Клиент отправляет опыт на сервер и получает оттуда действия которые нужно совершать
- Исправляете баги, тренируете агентов
- ...
- Profit!

Вопросы?

Антон Печенко

 forpost78@gmail.com