

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»
(Университет ИТМО)

Факультет систем управления и робототехники

ОТЧЕТ
по проектному заданию
на тему «Моделирование двуногого шагающего робота в MuJoCo»
по дисциплине
«Имитационное моделирование робототехнических систем»

Студент:
Группа № R4133c

Петрекеев К.С.
505881

Преподаватель:

Ракшин Егор Александрович

Санкт-Петербург 2025

Техническое задание

Цель проекта: разработать модель двуногого шагающего робота и исследовать возможность реализации обучения навыкам ходьбы при проведении симуляции в среде MuJoCo.

В рамках проекта требуется выполнить следующие шаги:

- Разработать модель двуногого робота из примитивов в формате .xml;
- Сформировать конфигурацию и параметры робота, необходимые для проведения обучения в симуляционной среде: параметры звеньев и приводов, начальную позицию робота для каждой итерации;
- Программно описать окружение для проведения обучения, выбрать политику обучения;
- Сформулировать функцию наград для политики обучения;
- Провести обучение, наблюдать за результатом работы робота в среде.

Разработка модели робота

В рамках проектной работы была рассмотрена модель двуногого шагающего робота — бипеда Atom P1 (рис. 1), обладающего кинематической схемой ног открытого типа.

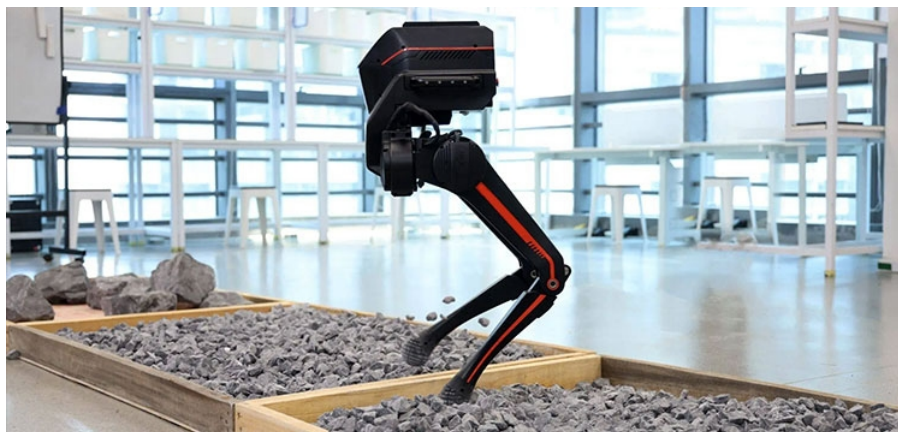


Рисунок 1: Робот Atom P1

Однако, так как пример рассматривался в качестве концептуальной идеи, было решено разработать собственную упрощенную схему робота, имеющего по четыре степени подвижности в каждой из ног: 3 привода в бедре и 1 в колене (в отличие от Atom, имеющего по 2 привода в каждом бедре).

Так, модель робота была создана из примитивов в формате .xml, и итоговое представление в mjoco.viewer представлено на изображении ниже (рис.2).

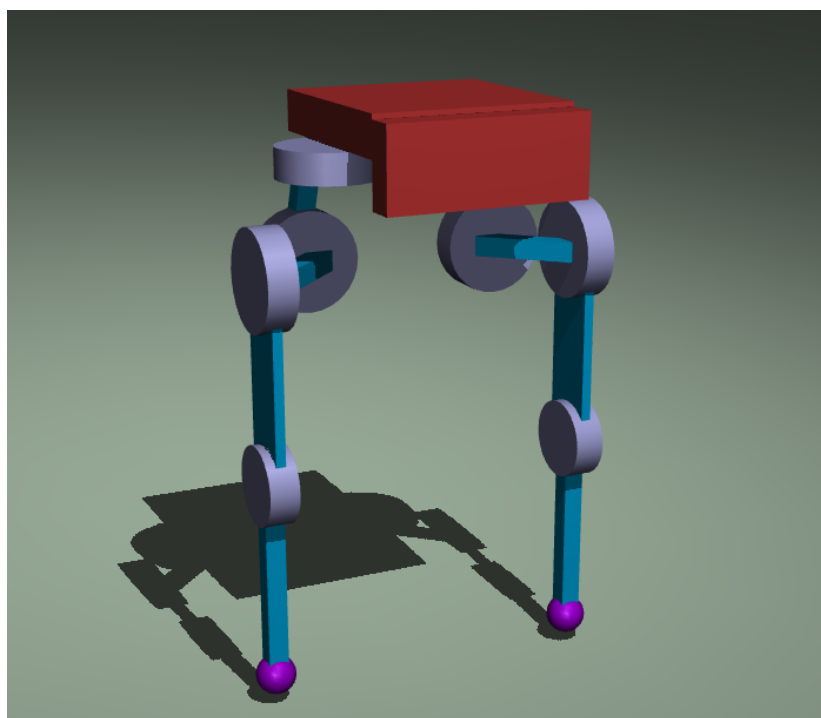


Рисунок 2: Модель разработанного робота в окне mjoco.viewer

В конфигурации робота были описаны необходимые ограничения в углах и скоростях поворотов сочленений; введены оценочные параметры масс и моментов инерции звеньев, необходимые для расчета динамики средствами симулятора. Основная начальная рабочая конфигурация робота, а также визуализация мест соприкосновения ног с опорной поверхностью и местонахождения центра масс системы приведены ниже (рис. 3, 4).

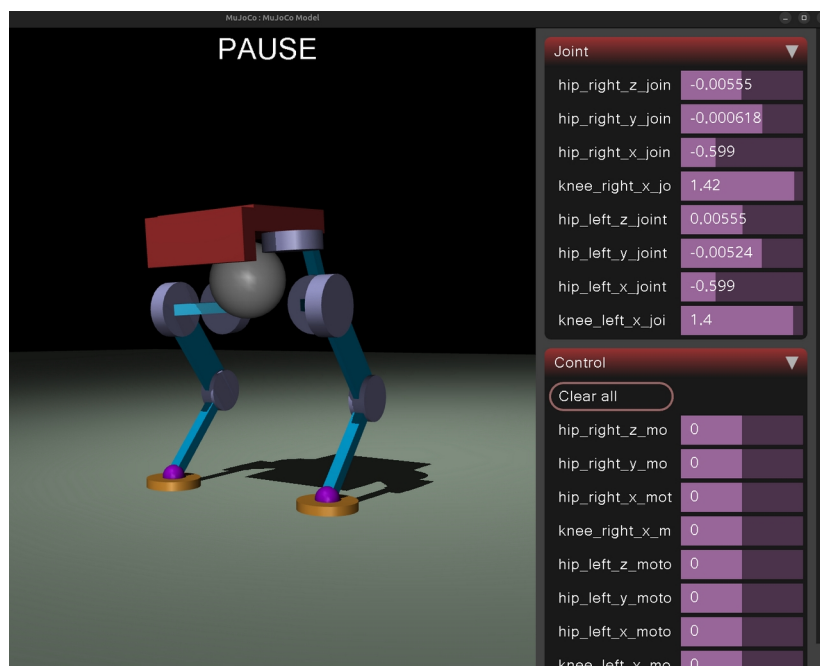


Рисунок 3: Начальная рабочая конфигурация робота

Обучение навыкам ходьбы

В процессе обучения был использован подход с созданием собственного окружения, наследованного от *gym.Env* и сформулирована простая функция наград для работы RL на основе алгоритма PPO:

```
reward = forward_vel + (0.2 if 0.45 < height < 0.6 else -1.0) - 0.25 * (
    np.abs(self.data.qvel[0]) + np.abs(self.data.qvel[2]))
reward = reward * 0.5 + (y_pos - self.prev_y_pos) * 0.5
```

Так как процесс обучения рассматривался в значительном упрощении и не учитывает взаимодействия с конфигурацией ног робота на каждом этапе, что, несомненно, является важным условием для работы RL-алгоритмов управления походкой при имитационном обучении или, например, в SafeRL подходах, текущая реализация позволяет политике управлять состоянием приводов

напрямую и, как описано выше, в качестве награды использует грубую функцию оценки текущего положения корпуса робота и его линейной скорости.

Для общего тестирования работы алгоритма было проведено 50 '000 итераций обучения. Конечно, для получения приемлемого результата используют гораздо большее число подходов, однако в текущей реализации этого оказалось вполне достаточно, чтобы получить от политики управление приемлемый результат. В итоге робот научился некоторым способом двигаться в указанном направлении, хотя, конечно, стоит брать в расчет несовершенство алгоритма управления. Видео-результаты и ссылка на репозиторий проекта приведены в приложении.

Заключение

В рамках проектной работы были выполнены все поставленные задачи: проведена разработка модели робота, сформировано окружение для проведения моделирования поведения робота в MuJoCo и обучение с использованием библиотек `gymnasium` и `stable_baselines3`.

В целом, действительно, средствами симулятора MuJoCo возможно проведение моделирования, может, и несколько упрощенного, систем управления шагающими роботами. Для более масштабных задач обучения в индустрии большее распространение имеет IsaacSim от Nvidia, позволяющий проводить параллельные вычисления, однако, с большей вычислительной стоимостью работы в целом. В текущей же задаче возможность быстрого создания окружения и модели робота с интеграцией RL-политики обучения вполне удобно и просто реализуема в рассмотренном ПО.

Приложение

- Ссылка на виде-запись части процесса обучения:
<https://disk.yandex.ru/i/9C9aJl0HuwyUrQ>
- Ссылка на репозиторий проекта: <https://github.com/t0sster/simrobs.git>