Искусственные общества. 2013-2023

ISSN 2077-5180

URL - <http://artsoc.jes.su>

Все права защищены

Выпуск 3 Том 16. 2021

Среда Mesa для агент-ориентированного моделирования на базе Python

Пигорев Дмитрий Павлович

*Центральный экономико-математический институт Российской академии наук
Российская Федерация, Москва*

Аннотация

Mesa - это лицензированная Apache среда разработки на агент-ориентированных моделях (или ABM) на языке Python. Она позволяет пользователям быстро создавать агент-ориентированные модели с использованием встроенных базовых компонентов (таких как пространственные сетки — spatial grids и планирование агентов — agent schedulers) или пользовательских реализаций; визуализировать их, используя интерфейс в окне браузера; и анализировать их результаты с помощью инструментов анализа данных Python. Целью разработки среды являлось желание авторов создать альтернативу для NetLogo, Repast или MASON на основе языка Python.

Ключевые слова: агент-ориентированное моделирование, имитационное моделирование, mesa, python

Дата публикации: 16.09.2021

Ссылка для цитирования:

Пигорев Д. П. Среда Mesa для агент-ориентированного моделирования на базе Python // Искусственные общества. – 2021. – Т. 16. – Выпуск 3.

URL: <https://artsoc.jes.su/s207751800008176-2-1/>. DOI: 10.18254/S207751800008176-2

¹ Mesa — это лицензированная Apache среда разработки агент-ориентированных моделей на языке Python. Она позволяет пользователям быстро создавать агент-ориентированные модели с использованием встроенных базовых компонентов (таких как пространственные сетки — spatial grids и планирование агентов — agent schedulers) или пользовательских реализаций; визуализировать их, используя интерфейс в окне браузера; и анализировать их результаты с помощью инструментов анализа данных Python. Целью разработки среды являлось желание авторов создать альтернативу для NetLogo, Repast или MASON на основе языка Python. Реализация на языке Python является значительным преимуществом, так как позволяет применять популярные в научной среде библиотеки для работы с данными и облачными сервисами, такими как SciPy, scikit-learn для машинного обучения, Keras, TensorFlow. С другой стороны, учитывая, что язык Python исконно является высокоуровневым скриптовым языком в процессе усложнения моделей могут возникать вопросы, касающиеся оптимизации производительности разрабатываемых моделей. Для этих целей может послужить компиляция с последующей оптимизацией байт-кода¹.

² Mesa имеет модульную архитектуру, это означает, что в Mesa представлены три типа модулей:

- ³
- моделирование;
 - анализ;
 - визуализация

⁴ Аналитические модули Mesa. Не каждая модель нуждается в модулях этого типа, но они предоставляют полезные инструменты для получения данных из прогонов моделей для более систематического изучения.

- ⁵
- Сборщики данных (data collectors) используются для записи данных каждого прогона модели
 - Пакетные запуски (batch runners) автоматизируют многократные прогоны и набор входных параметров - запуская модель с различными параметрами, чтобы увидеть, как они изменяют свое поведение.

⁶ Модули визуализации Mesa. Визуализация позволяет вам непосредственно наблюдать за прогонами моделей, наблюдать за динамикой, возникающей из них, и следить за тем, чтобы все шло так, как вы этого хотите. Mesa обрабатывает визуализации в окне браузера, используя JavaScript. Он предоставляет набор готовых компонентов, которые могут быть созданы для конкретной модели в Python и автоматически генерировать соответствующие объекты в окне браузера.

⁷ Также легко написать свои собственные компоненты обладая некоторыми базовыми знаниями JavaScript. Некоторые модули визуализации, которые мы здесь используем, включают:

- ⁸
- Визуализация Решетки (Grid visualization)
 - Модуль отображения графиков (Chart display),
 - Сам модульный сервер (The Modular Server).

⁹ GitHub репозиторий Mesa² включает некоторую коллекцию классических моделей (многие из которых можно найти среди моделей NetLogo / MASON), реализованных на базе среды Mesa:

- ¹⁰ • **bank_reserves** Модель высокого уровня абстракции, упрощенная модель экономики, в которой только один тип агента и один банк представляют все банки в экономике.
- **color_patches** Модель клеточного автомата, в которой на мнения агентов влияет мнение их соседей. По мере развития модели цветные пятна, представляющие преобладающее мнение в данной области, расширяются, сжимаются, а иногда и исчезают.
- **conways_game_of_life** Реализация игры жизни Конвея (Conway), клеточного автомата, в котором простые правила могут создавать сложные шаблоны.
- **epstein_civil_violence** Модель Джошуа Эпштейна (Joshua Epstein) представляющее то, как децентрализованное восстание может быть подавлено или же может достигнуть критической массы поддержки.
- **boid_flockers** Модель стаи птиц, демонстрирующая поведение агентов, движущихся в непрерывном пространстве по векторам направления.
- **forest_fire** Простые клеточные автоматы пожара, распространяющиеся по лесу ячеек на сетке, основанные на модели NetLogo Fire.
- **hex_snowflake** Игра жизни Конвея на гексагональной сетке.
- **pd_grid** Модель Дилеммы Заключенного на основе сетки, демонстрирующая, как простое подражание может привести к обширному сотрудничеству и как модельный режим активации может изменить его результат. Реализация классической модели сегрегации Шеллинга на базе Mesa.
- **boltzmann_wealth_model** Законченный код, который предлагается вместе с руководством по созданию простой модели того, как простой набор правил приводит к возникновению неравномерного распределения богатства.
- **wolf_sheep** Реализация экологической модели хищничества и размножения на основе модели NetLogo Wolf Sheep Predation.
- **sugarscape_cg** Внедрение модели постоянного роста Sugarscape 2 на основе модели Netlogo Sugarscape 2.
- **virus_on_network** Эта модель основана на модели NetLogo «Вирус в сети».

¹¹ Платформа Mesa представляет интерес для разработчиков агент-ориентированных моделей, в том числе для моделирования социально-экономических систем. Однако, для целей построения геопространственных моделей могут потребоваться дополнительные модули-расширения, такие как Mesa-geo³. Безусловным преимуществом является полная открытость среды к доработкам и кастомизации в соответствии с потребностями создаваемых моделей.

¹² Установка и запуск. Для быстрого запуска и начала работы с Mesa фреймворк может быть установлен с использованием установщика Python пакетов pip (при условии, что сам Python у вас уже установлен):

13 \$ pip install mesa

14 Так же существует возможность скачать дистрибутив из GitHub:

15 \$ pip install -e git+https://github.com/projectmesa/mesa#egg=mesa

16 Для целей экспериментов с инструментарием автором статьи был создан Docker-контейнер в ОС Windows. Дистрибутив платформы для контейнеризации Docker можно скачать на официальном сайте⁴. Далее следуя инструкции⁵ Mesa framework по разворачиванию Mesa в Docker-контейнере это может быть выполнено с использованием Docker composer, установив который необходимо исполнить следующий перечень команд:

17 \$ docker-compose build --pull ... \$ docker-compose up -d dev # start the docker container

18 \$ docker-compose exec dev bash # enter the docker container that has your current version of Mesa installed at /opt/mesa

19 \$ mesa runserver examples/schelling # or any other example model in examples

20 Файл Docker-compose выполняет две важные задачи:

- 21 • Он связывает порт 8521 Docker контейнера с портом 8521 вашей хост-системы, чтобы вы могли взаимодействовать с работающей моделью как обычно, по адресу localhost: 8521 в своем браузере.
- Он монтирует корневой каталог mesa (относительно файла docker-compose.yml) в / opt / mesa и запускает pip install -e в этом каталоге, чтобы ваши изменения в mesa отражались в работающем контейнере.

22 Если вы являетесь разработчиком модели и хотите запустить Mesa (при условии, что вы в настоящее время находитесь в корневом каталоге модели с файлом run.py) вам необходимо выполнить следующую команду в консоли Вашего docker контейнера:

23 \$ docker run --rm -it -p127.0.0.1:8521:8521 -v\${PWD}:/code comses/mesa:dev mesa runserver /code

24 Справка и дополнительная информация. Для получения большей информации рекомендуется использование официальной документации на веб-сайте фреймворка Mesa:

- Введение
(http://mesa.readthedocs.org/en/master/tutorials/intro_tutorial.html)
- Документация ([>>>>](#))
- Форум ([>>>>](#))
- PyPi (<https://pypi.python.org/pypi/Mesa/>)

25 Для ознакомления с примерами использования фреймворка для целей агент-ориентированного моделирования на веб-сайте представлена коллекция демонстрационных моделей: [>>>>](#) .

Примечания:

1. >>>>
 2. >>>>
 3. >>>>
 4. >>>>
 5. >>>>
-

Библиография:

1. Акопов А.С. Имитационное моделирование. Учебник и практикум для академического бакалавриата. М.: Юрайт, 2015.
2. Емельянов, А. А. Имитационное моделирование экономических процессов. М.: Финансы и статистика, 2009
3. Лычкина, Н.Н. Имитационное моделирование экономических процессов. Инфра-М, 2012 г.
4. Макаров В.Л., Бахтизин А.Р., Сушко Е.Д. Агент-ориентированные модели как инструмент апробации управленческих решений. – М.: Управленческое консультирование, 2016.
5. Макаров В.Л., Бахтизин А.Р. Социальное моделирование – новый компьютерный прорыв (агент-ориентированные модели) – М.: Экономика, 2013.

Mesa Python platform for agent-based modelling

Dmitry Pigorev

*Central Economics and Mathematics Institute of the Russian Academy of Sciences
Russian Federation, Moscow*

Abstract

Mesa is a licensed by Apache development environment for agent-based models (or ABM) in Python. It allows users to create agent-based models using built-in basic components (such as spatial and agent schedulers) or custom implementations; visualize them using the interface in a web browser; and analyze their results using Python data analysis tools. The purpose of the development of the environment was the desire of the authors to create an alternative for NetLogo, Repast or MASON based on the Python language.

Keywords: agent based modelling, mesa, python, abm, imitation modelling

Date of publication: 16.09.2021

Citation link:

Pigorev D. Mesa Python platform for agent-based modelling // Artificial societies. – 2021. – V. 16. – Issue 3. URL: <https://artsoc.jes.su/s207751800008176-2-1/>. DOI: 10.18254/S207751800008176-2