MESA-GEO

A GIS EXTENSION FOR THE MESA AGENT-BASED MODELING FRAMEWORK IN PYTHON

Boyu Wang , Vincent Hess, Andrew Crooks

GeoSim ’22, Seattle, WA

November 1, 2022

University at Buffalo The State University of New York

Agenda

Introduction

Background

Architecture

Applications

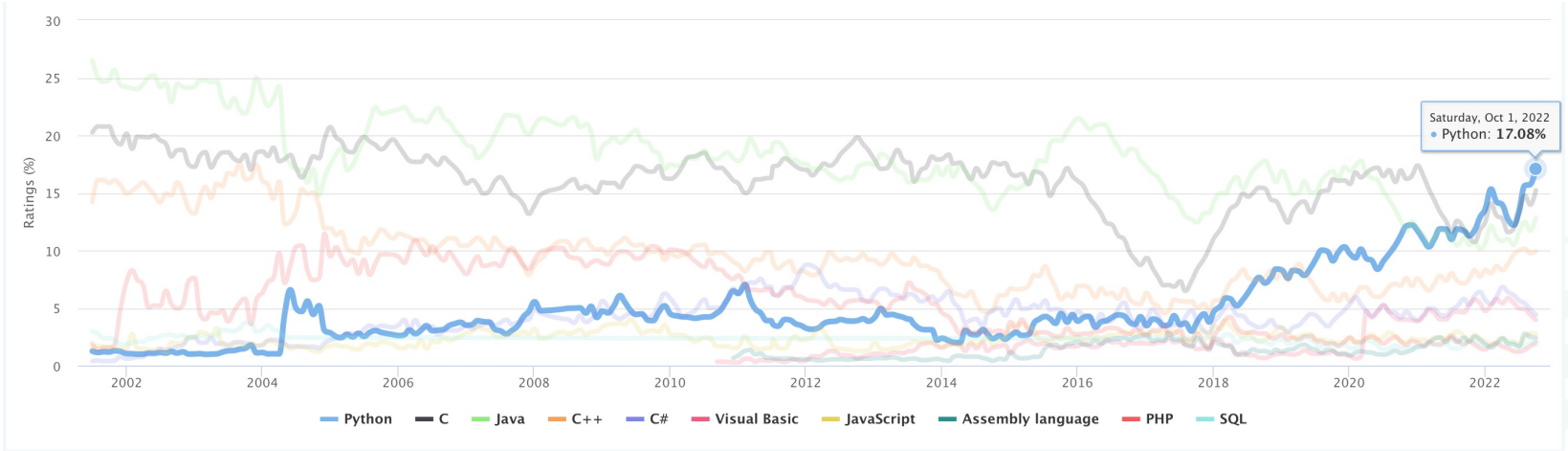
Conclusion

Introduction

- Mesa is an open-source agent-based modeling (ABM) framework written in Python, allowing users to quickly build and visualize agent-based models.
- Mesa-Geo is its GIS extension. Due to the modular design of Mesa, extending its functionalities is relatively straight-forward, without the necessity of modifying its core components.

Why Python?

- There has been a trend of utilizing machine learning methods in and for ABM and GIS.
- Python is currently the dominant programming language for machine learning (ML): NumPy, pandas, scikit-learn, Keras, PyTorch, and of growing interest to GIScience (PySAL, GeoPandas, Rasterio ...).

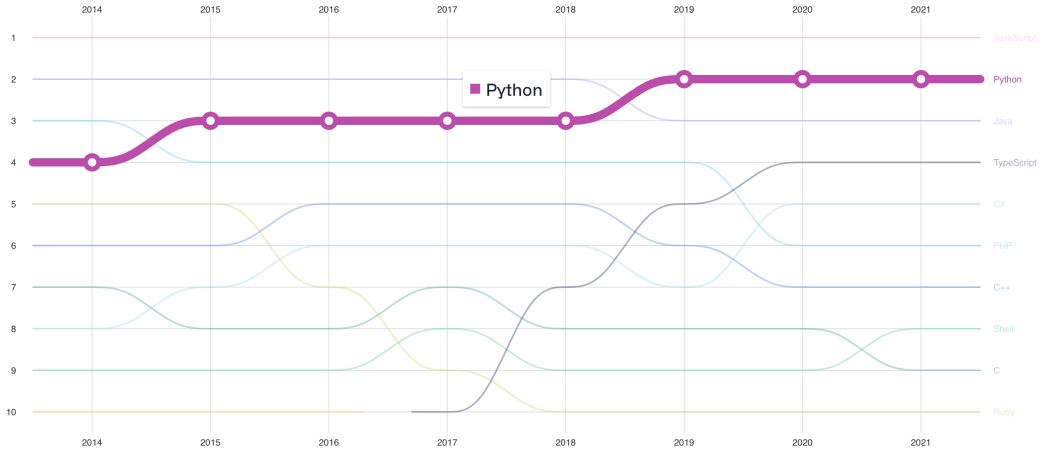


TIOBE Programming Community Index

source: https://www.tiobe.com/tiobe-index

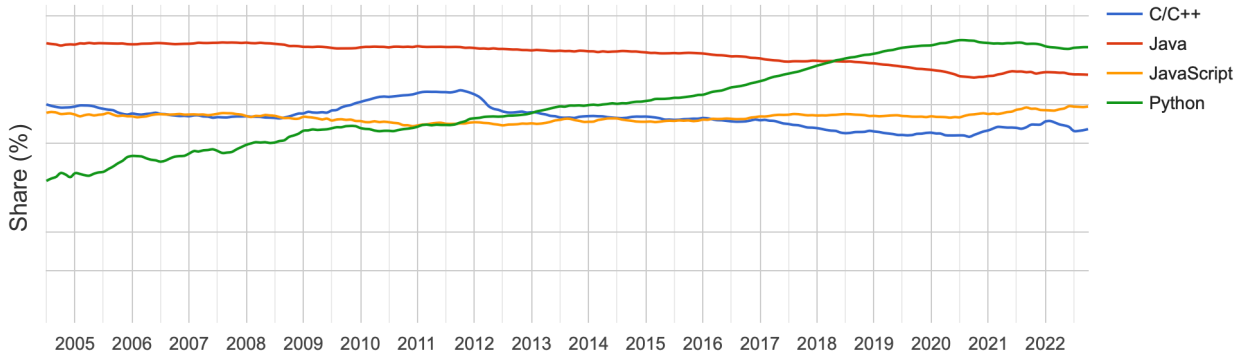
Top languages over the years

source: https://octoverse.github.com



PYPL PopularitY of Programming Language

source: https://pypl.github.io/PYPL.html



Background

	NetLogo	Repast Simphony	AnyLogic	MASON	GAMA	Mesa	AgentScript
Initial Release Year	1999	2000	2000	2003	2009	2015	2018
Latest Release Year	2022	2022	2022	2022	2022	2022	2022
License	GPL	BSD	Proprietary	Academic Free License 3.0	GPLv3	Apache 2.0	GPLv3
Implementation Language	Scala, Java	Java	Java	Java	Java	Python	JavaScript
Modeling Language / Interface	NetLogo	ReLogo, statecharts, Groovy, Java	GUI, Java, UML-RT	Java	GAML (GAma Modeling Language)	Python	JavaScript
Raster Data Support	gis-extension	Yes	Yes	GeoMason extension	Yes	Mesa-Geo extension	Yes
Vector Data Support	gis-extension	Yes	Yes	GeoMason extension	Yes	Mesa-Geo extension	Yes

Background

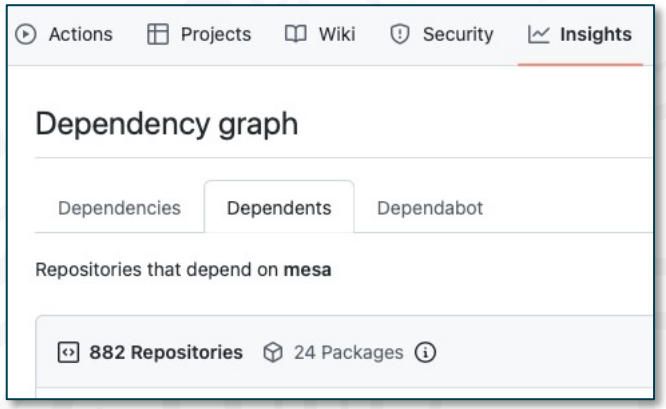
Mesa

- Created in 2015. Source code: https://github.com/projectmesa/mesa
- Over the years, Mesa has been used in a wide range of application areas, from epidemiology, logistics, healthcare, to the modeling of electricity market, auction, food market, building, etc.

Mesa-Geo

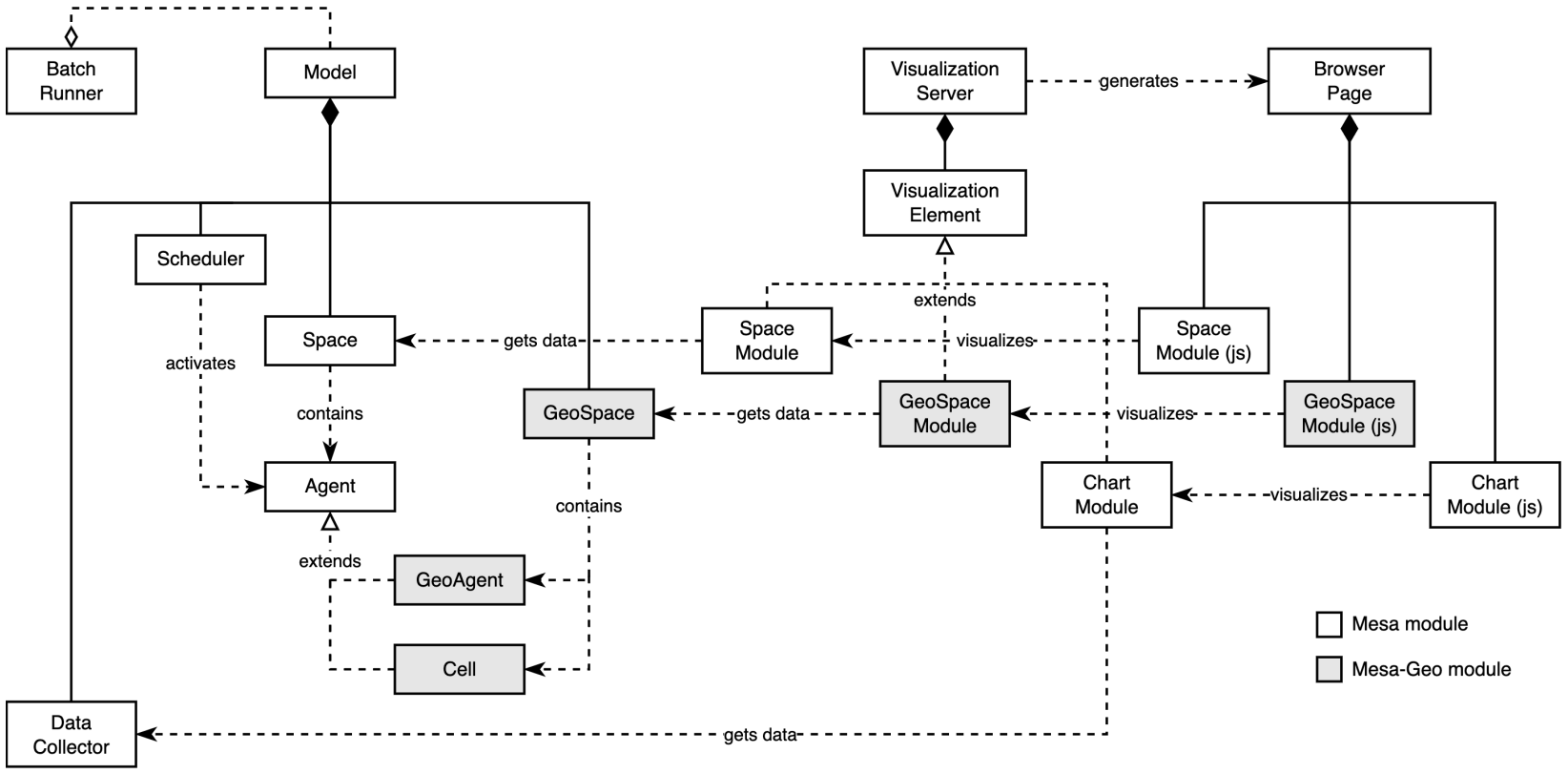
- Created in 2017. Source code: https://github.com/projectmesa/mesa-geo
- Initially focused on vector data support. Our contributions are integrating more sophisticated GIS functionalities such as:
  - Raster data support
  - Consistent coordinate reference system (CRS) settings

  along with other changes including frontend dependencies management, bug fixes, etc.



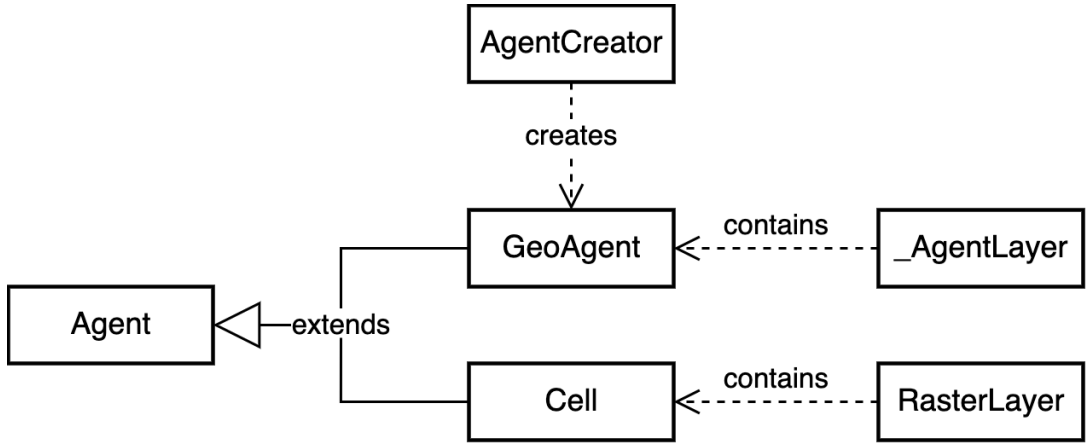
As of October 25, 2022.
Source: https://github.com/projectmesa/mesa/network/dependents

Architecture

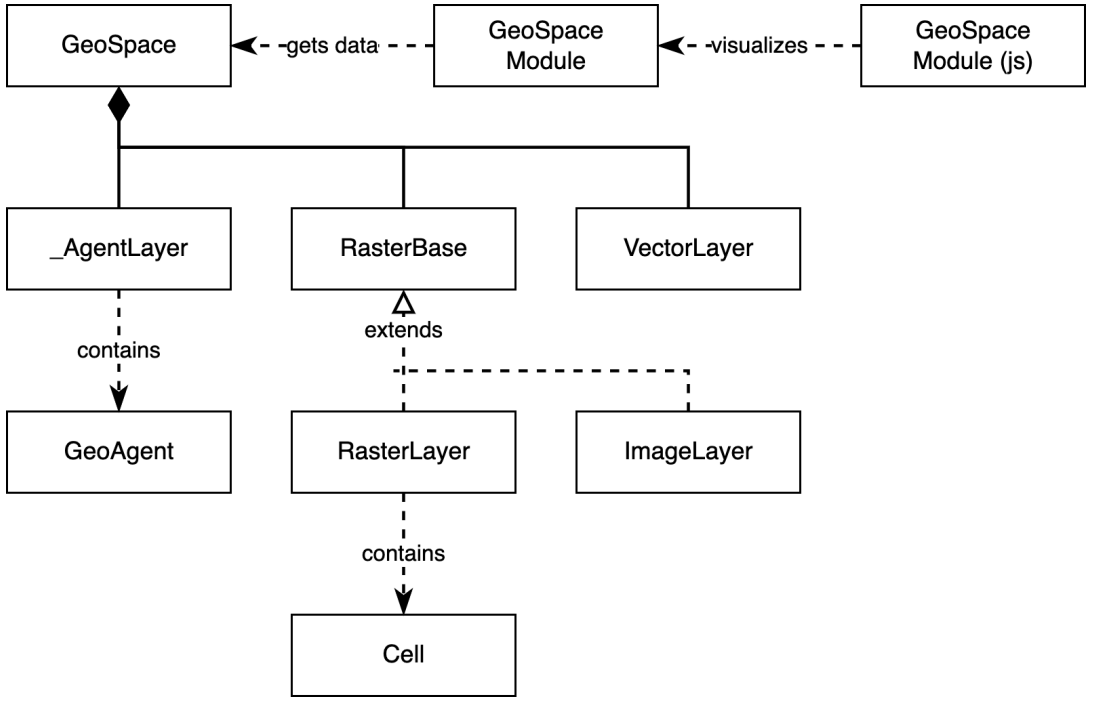


High-level component diagram of Mesa & Mesa-Geo

Architecture



Class diagram of the Agent, GeoAgent, and Cell classes



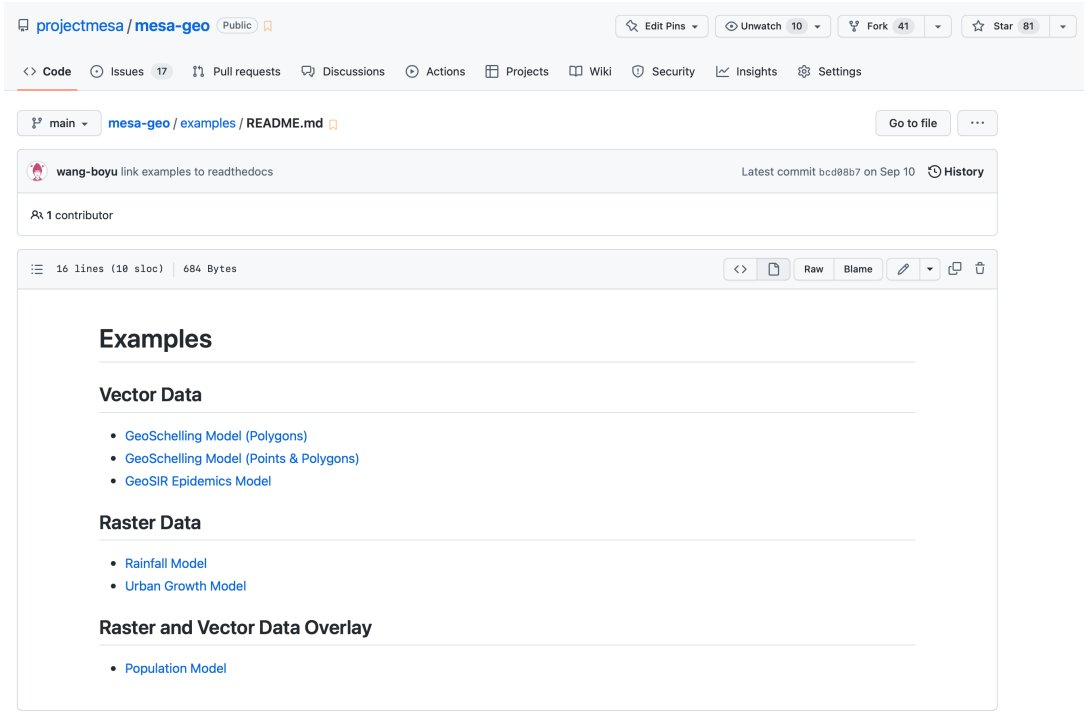
Class diagram of GeoSpace and its related classes

Applications

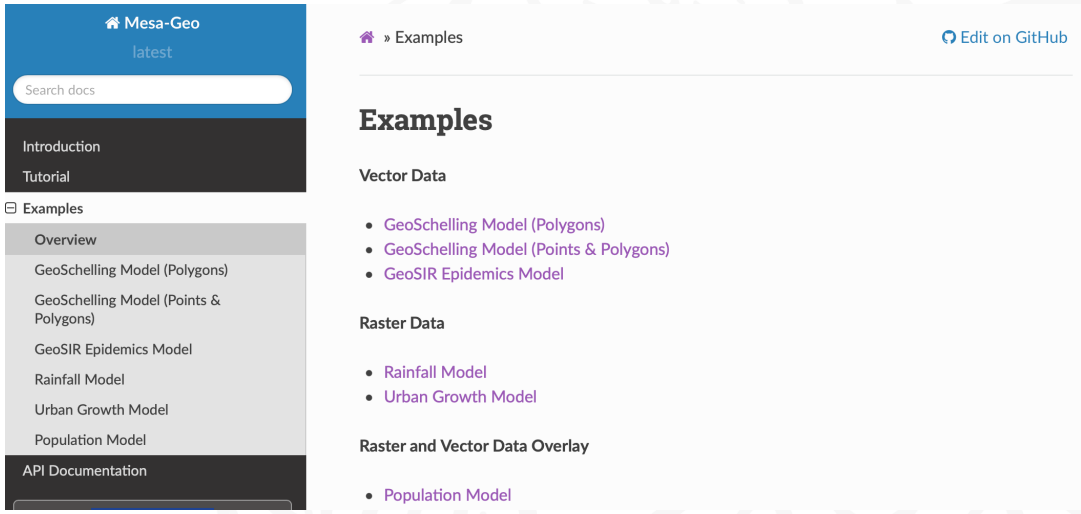
Examples of GIS functions commonly needed in ABM:

- Basemaps
- GIS data import/export
- Raster data:
  - Digital elevation (e.g., pedestrian evacuation)
  - Neighborhood queries (e.g., Moore, von Neumann)
- Vector data:
  - Buffers
  - Shortest path queries (e.g., road network)
  - Contains/Within/Intersects/…
- Raster & vector data overlay

Example Models



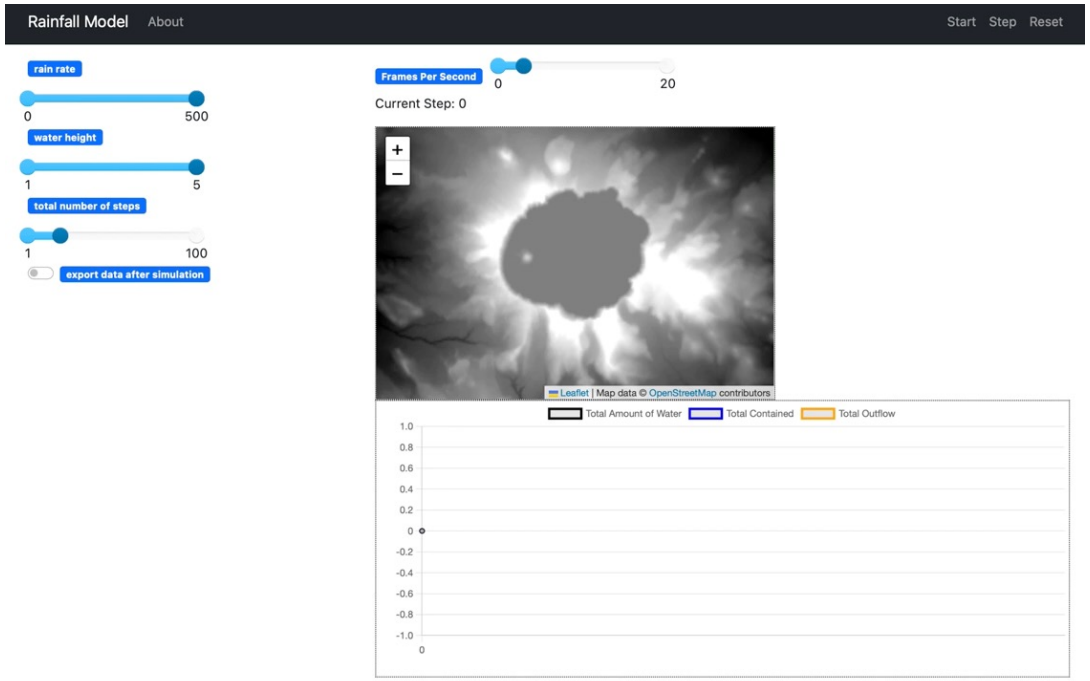
Link: https://github.com/projectmesa/mesa-geo/blob/main/examples



Link: https://mesa-geo.readthedocs.io/en/latest/examples/overview.html

Example: Digital Elevation Model

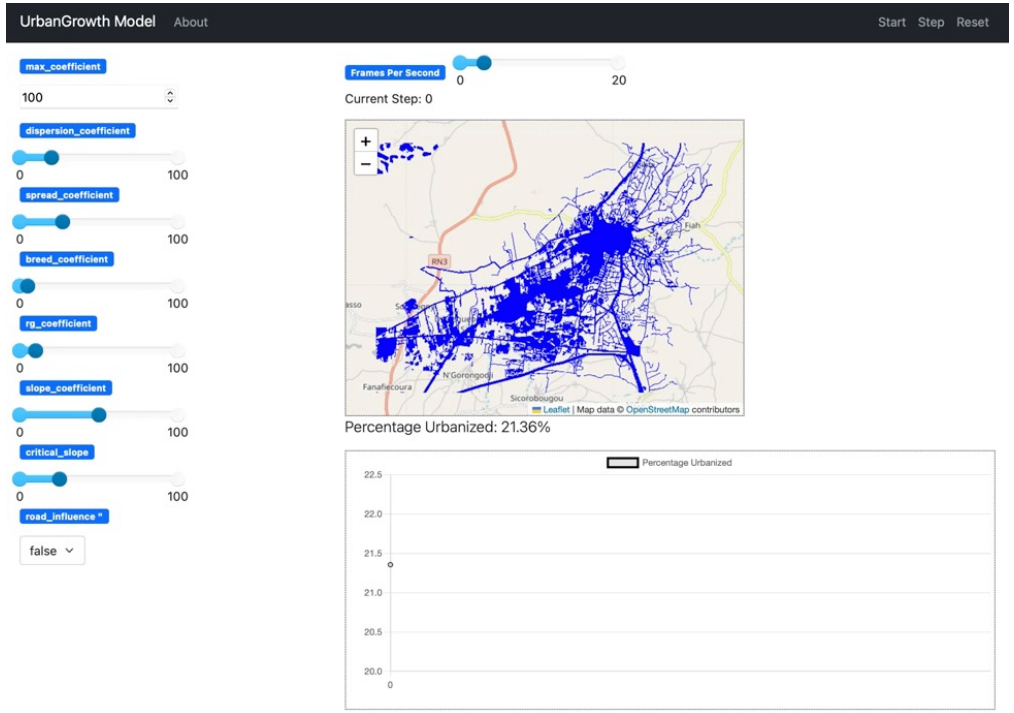
RAINFALL MODEL



- **GeoSpace**: a raster layer representing elevations.
- **GeoAgents**: raindrops.
- At each time step, raindrops are randomly created across the landscape to simulate rainfall.
- The raindrops flow from cells of higher elevation to lower elevation based on their eight surrounding cells (i.e., Moore neighbourhood).

Example: Multiple Raster Layers

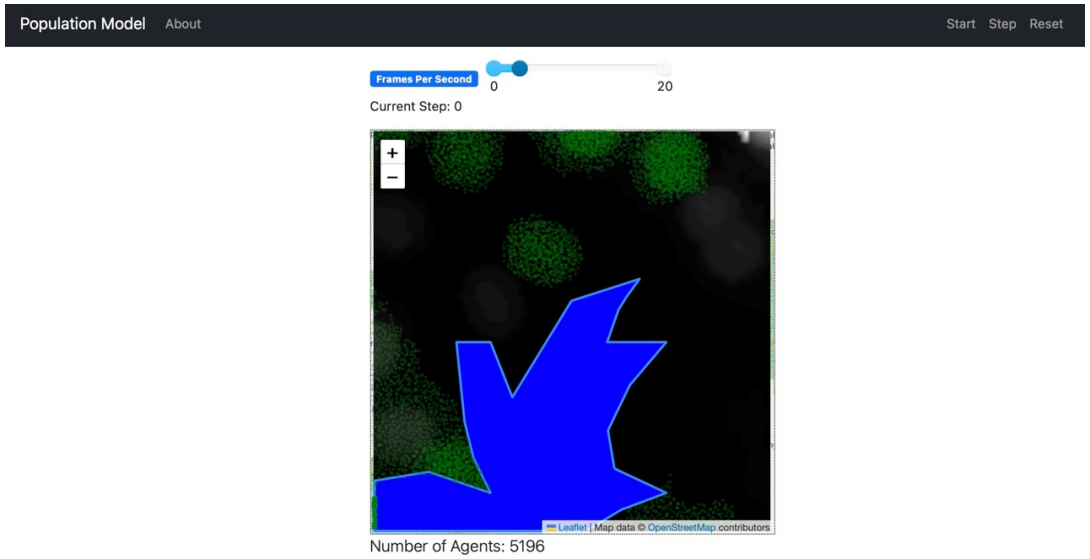
URBAN GROWTH MODEL



- **GeoSpace**: multiple raster layers representing slope, road, land use, urban area, and so on.
- **Cells**: land parcels.
- At each time step, each land parcel is decided whether it is suitable to be urbanized, based on the input raster layers as well as the user defined coefficients.

Example: Raster & Vector Overlay

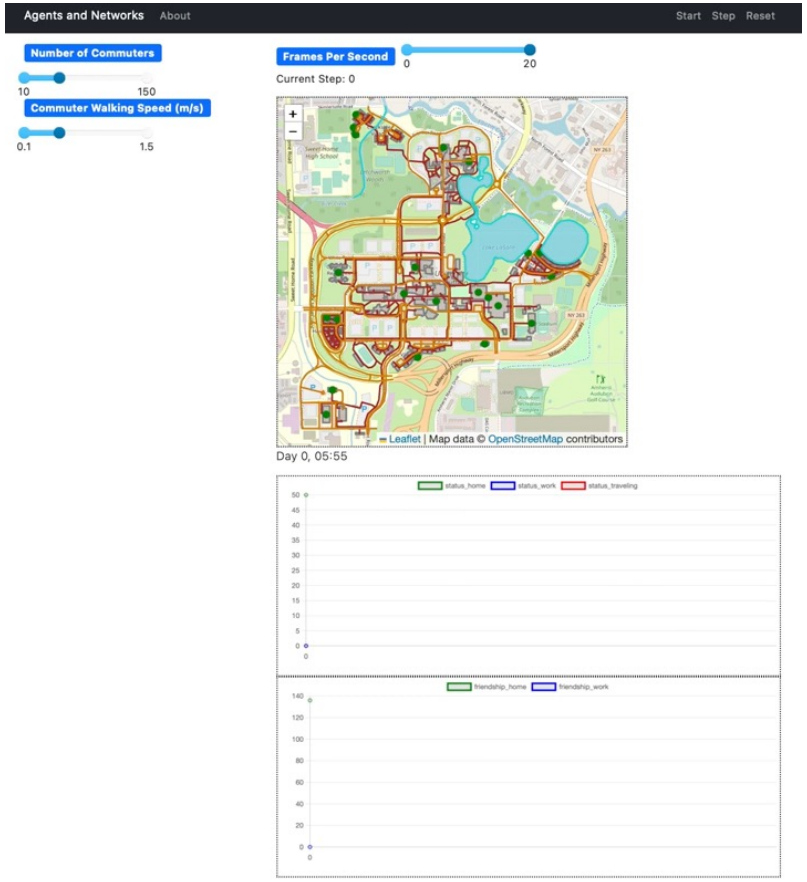
POPULATION MODEL



- **GeoSpace**:
  - a raster layer of population data for each cell.
  - a vector layer representing a lake.
- **GeoAgent**: people, created based on the population data.
- The agents move randomly to neighbouring cells at each time step.

Example: Road Network

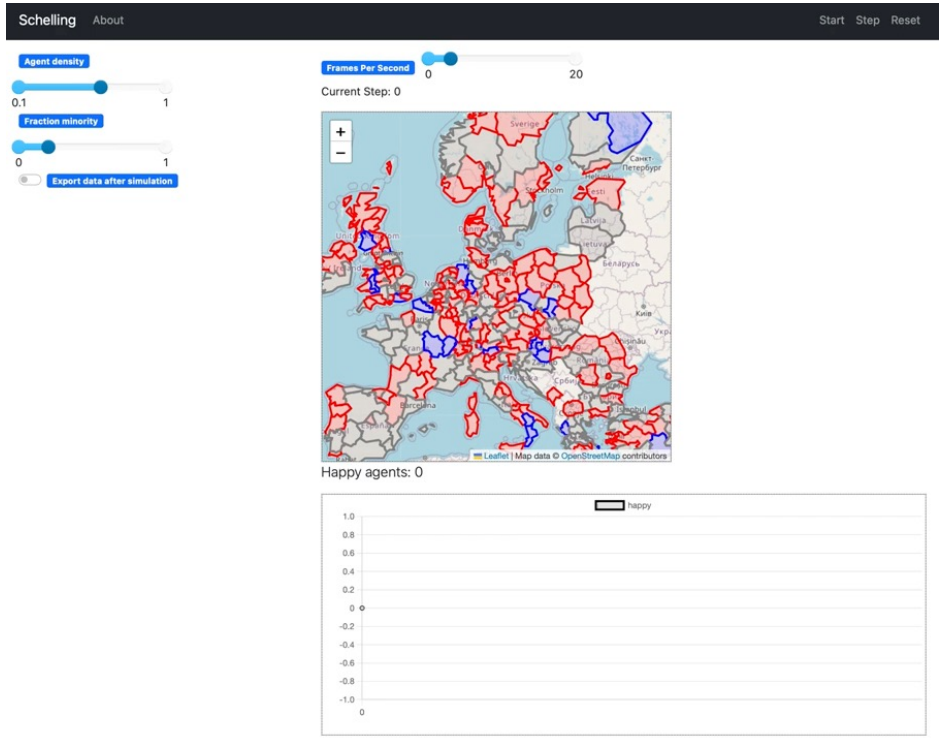
AGENTS AND NETWORKS MODEL



- **GeoSpace**: multiple vector layers, including buildings, lakes, and a road network. The road network is constructed from polyline data.
- **GeoAgent**: commuters.
- Buildings are randomly assigned to agents as their home and workplaces.
- Agents' commute routes can be found as the shortest path between entrances of their home and workplaces. Their movements are constrained on the road network.
- Source code available at https://github.com/wang-boyu/agents-and-networks-in-python

Example: Points & Ploygons

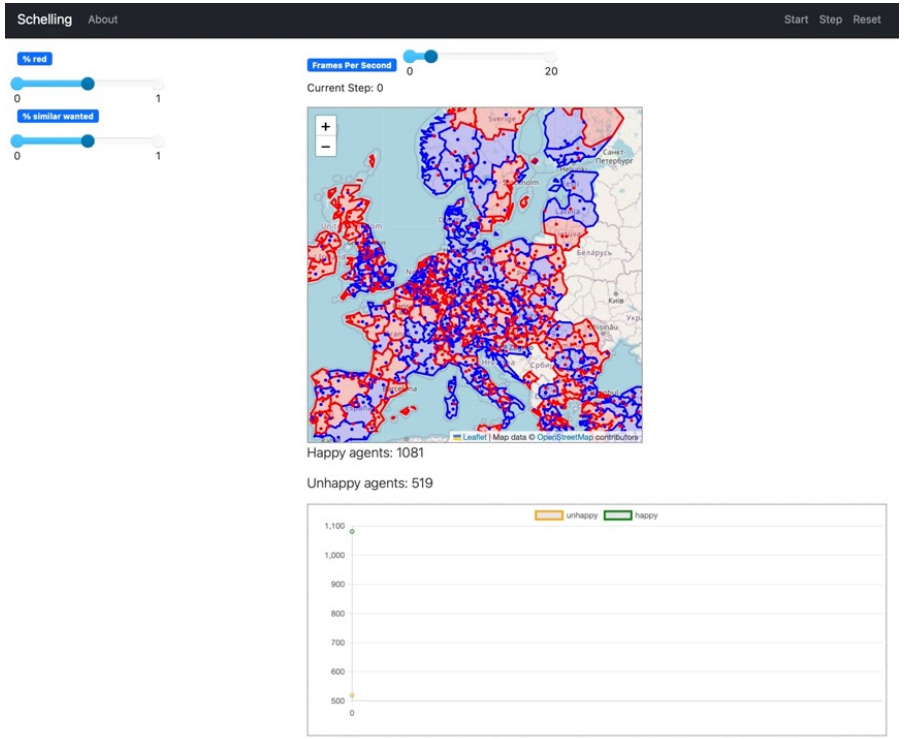
GEO SCHELLING (POLYGONS)



- **GeoSpace**: only the agent layer containing GeoAgents.
- **GeoAgents**: the Level 2 European Nomenclature of Territorial Units for Statistics (NUTS-2) regions.
- During the running of the model, a polygon queries the colors of the surrounding polygon and if the ratio falls below a certain threshold (e.g., 40% of the same color), the agent moves to an uncolored polygon.

Example: Points & Ploygons

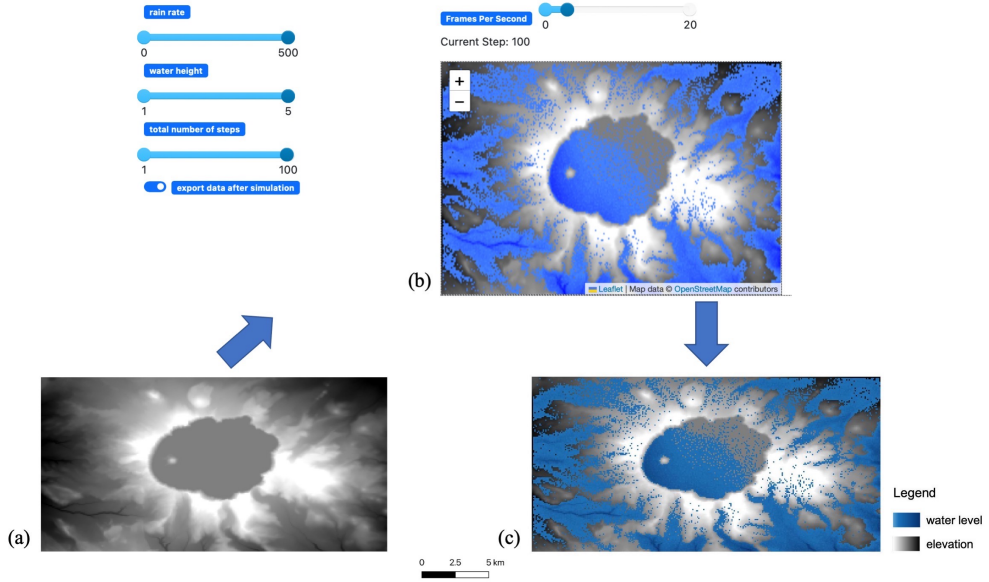
GEO SCHELLING (POINTS & POLYGONS)



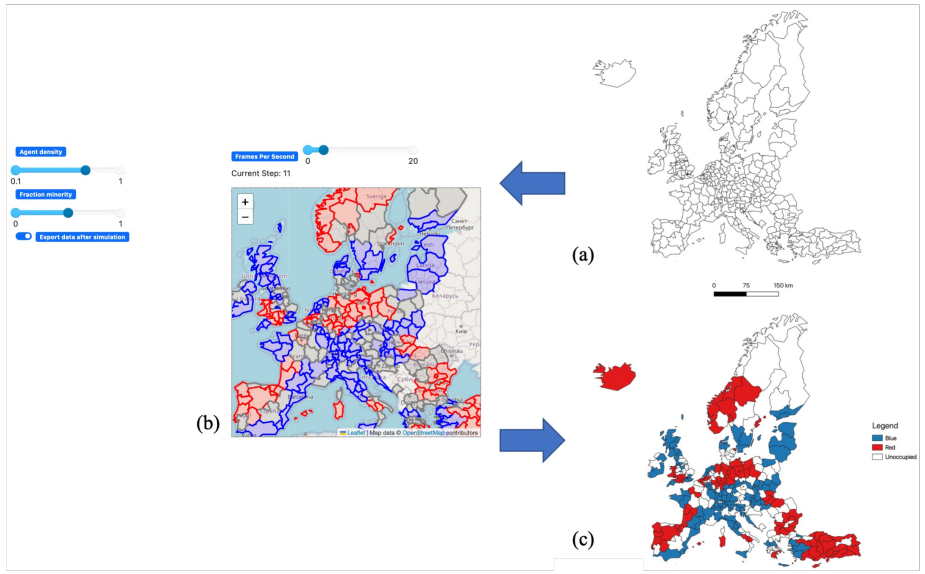
- **GeoSpace**: only the agent layer containing GeoAgents.
- **GeoAgents**:
  - NUTS-2 regions.
  - People residing in NUTS-2 regions.
- Each person resides in a randomly assigned region and checks the color ratio of its region against a pre-defined "happiness" threshold at every time step.
- If the ratio falls below a certain threshold (e.g., 40%), the agent is found to be "unhappy", and randomly moves to another region.

Data Export

- Raster data



- Vector data



Conclusion

- Mesa-Geo: an open-source GIS extension for the Mesa agent-based modeling framework in Python.
- By utilizing the rich software ecosystem of open-source scientific libraries (e.g., Rasterio, GeoPandas, NetworkX), users can import, manipulate and visualize georeferenced data in ABM.

https://github.com/projectmesa/mesa-geo

https://mesa-geo.readthedocs.io

Acknowledgments

- Mesa-Geo is a community effort. Our special thanks to rht, Jackie Kazil and Tom Pike among many others for their invaluable support.

https://github.com/projectmesa/mesa-geo

https://mesa-geo.readthedocs.io

Join us!

- Matrix chat room: https://matrix.to/#/#mesa-geo:matrix.org
- Monthly dev session: https://github.com/projectmesa/mesa/discussions
- Mesa-Geo discussions: https://github.com/projectmesa/mesa-geo/discussions
- Contributors guide: https://github.com/projectmesa/mesa-geo/blob/main/CONTRIBUTING.md

https://github.com/projectmesa/mesa-geo

https://mesa-geo.readthedocs.io

University at Buffalo The State University of New York

THANK YOU FOR LISTENING!

WELCOME COMMENTS, QUESTIONS AND SUGGESTIONS.

bwang44@buffalo.edu
vincent.hess@posteo.de
atcrooks@buffalo.edu

@BoyuWang_
@AndyCrooks

@wang-boyu
@Corvince
@acrooks2