PiperABM: A Python Library for Resilience-Based Agent Modeling

Aslan Noorghasemi [1¶] **and Christopher McComb** [1]

1 Department of Mechanical Engineering, Carnegie Mellon University, USA ¶ Corresponding author

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Open Journals

Reviewers:

- @openjournals

Submitted: 01 January 1970
Published: unpublished

Summary

`PiperABM` is an open-source Python library designed to support resilience-based agent modeling on complex infrastructure networks. It provides modular tools for constructing agent-based simulations where individual agents interact over dynamic networks subject to progressive degradation and adaptive decision-making. Built with extensibility in mind, PiperABM leverages a bootstrap architecture that allows users to customize agent behaviors. Core features include dynamic network loading, failure propagation models, accessibility and travel-distance metrics, and visualization utilities. PiperABM is framework-agnostic and integrates seamlessly with common scientific Python ecosystems (NumPy, NetworkX, Matplotlib).

Statement of need

Noorghasemi_KeepDelta_A_Python_2025

by supplying their own `decision_making.py` modules.

Infrastructure resilience is a critical concern for urban planners, emergency managers, and researchers seeking to understand how disruptions (e.g., natural hazards, maintenance backlogs) affect community access to essential services.

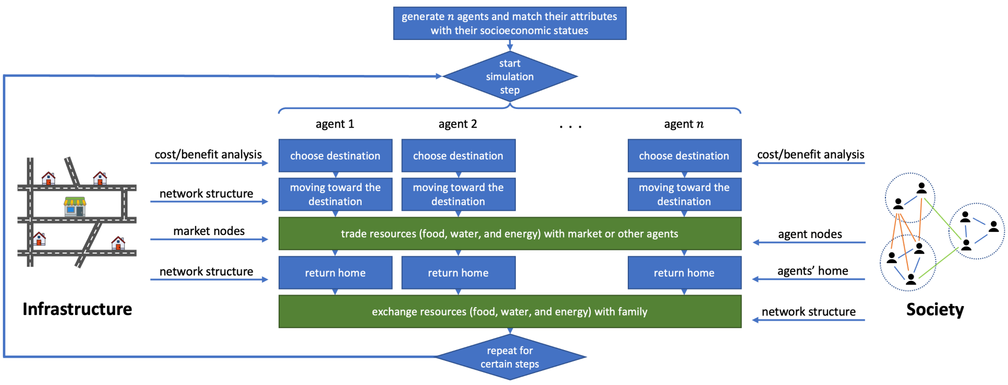

Figure 1: The computational model emulates the relation between the elements of infrastructure and social networks.

Measurements

...

Accessiblity

Each agent's accessibility to resources is assessed at every time step to monitor their well-being and ability to meet their needs. The term accessibility $A_{i,t,r}$ for agent i at time t for resource r is computed as:

$$A_{i,t,r} = \frac{R_{i,t}}{R_i^{\max}}$$

where $R_{i,t}$ is the amount of resource r that agent i possesses at time t, and $R_i^{\max}$ is the maximum capacity of resource r that agent i can have. A value of 1 indicates full accessibility.

To aggregate across the R different resources for each agent, we use the geometric mean:

$$A_{i,t} = \left(\prod_{r=1}^{R} A_{i,t,r} \right)^{\frac{1}{R}}$$

This ensures that low accessibility in any single resource strongly impacts the overall score. If any $A_{i,t,r} = 0$, then $A_{i,t} = 0$ and the agent is considered dead.

Across all N agents at each time step, the community's average accessibility is:

$$A_t = \frac{1}{N} \sum_{i=1}^{N} A_{i,t}$$

Finally, a time-weighted overall accessibility over the simulation duration T is

$$A = \frac{\int_0^T A_t \, \mathrm{d}t}{\int_0^T A_{\max} \, \mathrm{d}t}$$

where $A_{\max} = 1$ is the maximum possible accessibility.

Travel Distance

In the context of agent-based modeling, *traveled distance* serves as a metric for assessing the efficiency and functionality of transportation networks within a simulated environment. This measurement tracks the cumulative distance agents must traverse between various points, e.g. from home to market.

When this measurement yields a low value, it indicates that the system is operating with high efficiency, allowing agents to traverse shorter distances between points to satisfy their needs. Alternatively, it could signal that various barriers, constraints, or issues are impeding agents' access to essential network nodes, thus limiting their ability to move freely within the system and reach their goals. This dual interpretation helps in diagnosing the underlying causes of system performance, guiding targeted improvements in urban planning and resource distribution.

Comparison to Existing Tools

PiperABM's strength lies in its opinionated support for resilience metrics, built-in animation utilities, and its minimal barrier for user-defined agent policies. Unlike Mesa or NetLogo, which require extensive boilerplate or domain-specific scripting, PiperABM users can implement new decision-making modules by inheriting from a common superclass. Compared to Repast,

The Journal of Open Source Software

PiperABM remains lightweight and fully Pythonic, benefiting from the broad data science ecosystem without Java dependencies.

References