

micahlc / ParameterisedAI

<> Code

Issues

Pull requests

Actions

Projects

Wiki

Security

Insights

Settings

A very basic OpenTTD AI, designed for exploring how its behaviour changes as parameters change

GPL-2.0 license

0 stars 0 forks 2 watching 1 Branch 1 Tags Activity

Public repository

main 1 Branch 1 Tags

Go to file

Go to file

+

Add file

<> Code

micahlc

Merge pull request #1 from micahlc/tests/basic-regression-tests 7466240 · 2 months ago

.github/workflows	tests: basic regression test	2 months ago
LICENSE	Initial commit	6 months ago
README.md	docs: show example results	3 months ago
example-results.svg	docs: show example results	3 months ago
info.nut	feat: rename to ParameterisedAI	3 months ago
main.nut	feat: rename to ParameterisedAI	3 months ago
requirements_test.in	tests: basic regression test	2 months ago
requirements_test.txt	tests: basic regression test	2 months ago
test_parameterised_ai.py	tests: basic regression test	2 months ago

README GPL-2.0 license

ParameterisedAI

A very basic OpenTTD AI, designed for exploring how its behaviour changes as parameters change, rather than as a competitor to play against.

The AI currently make a single bus route, and the only parameter is the number of buses to attempt to build on that route.

Usage

A convenient way of running this AI is with [OpenTTDLab](#), with results then processed and visualised using [pandas](#) and [Plotly Express](#).

```
python -m pip install OpenTTDLab==0.0.52 pandas==2.2.0 plotly==5.18.0
```

To run an experiment:

```
from openttdlab import run_experiment, remote_file, bananas_ai_library

results = run_experiments(
    openttd_version='13.4',
    opengfx_version='7.1',
    experiments=(
        {
            'seed': seed,
            'ais': (
                remote_file(
                    'https://github.com/micahlc/SupplyChainLabAI/archive/d3ac662b47267ed4fa84a5b3997c020ef140f1e2.ta
                    ai_name='ParameterisedAI',
                    ai_params=(
                        ('maximum_buses', maximum_buses),
```

```

    ),
    ),
    ),
    # Increase to run for longer
    'days': 365 * 4 + 1,
}
for maximum_buses in [1, 2, 4, 8, 16]
for seed in range(0, 10)
),
ai_libraries=(
    bananas_ai_library('5046524f', 'Pathfinder.Road'),
    bananas_ai_library('4752412a', 'Graph.AyStar'),
    bananas_ai_library('51554248', 'Queue.BinaryHeap'),
),
)
)

```

To then extract results:

```

import pandas as pd

df = pd.DataFrame(
    {
        # It's slightly awkward right now to get at the original AI params
        'max_buses': row['experiment']['ais'][0][1][0][1],
        'seed': row['experiment']['seed'],
        'date': row['date'],
        'money': row['chunks']['PLYR']['0']['money'],
    }
    for row in results
)
df = df.pivot(index='date', columns=('seed', 'max_buses'), values='money')
df = df.T.groupby(level=1).mean().T

```

And then to plot them:

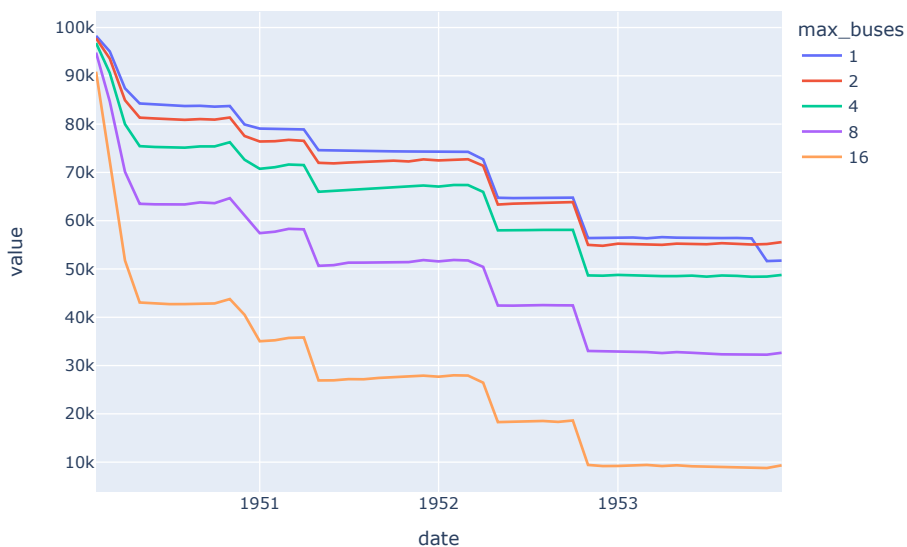
```

import plotly.express as px

fig = px.line(df)
fig.show()

```

To show:



Releases 1

 **v0.0.1** Latest

on Jun 8

Packages

No packages published
[Publish your first package](#)

Languages

