

INF200 – BioSim Project

Modelling the Ecosystem of Rossumøya

By Sindre Elias Hinderaker & Mathias Kristiansen

Project structure

Data structure

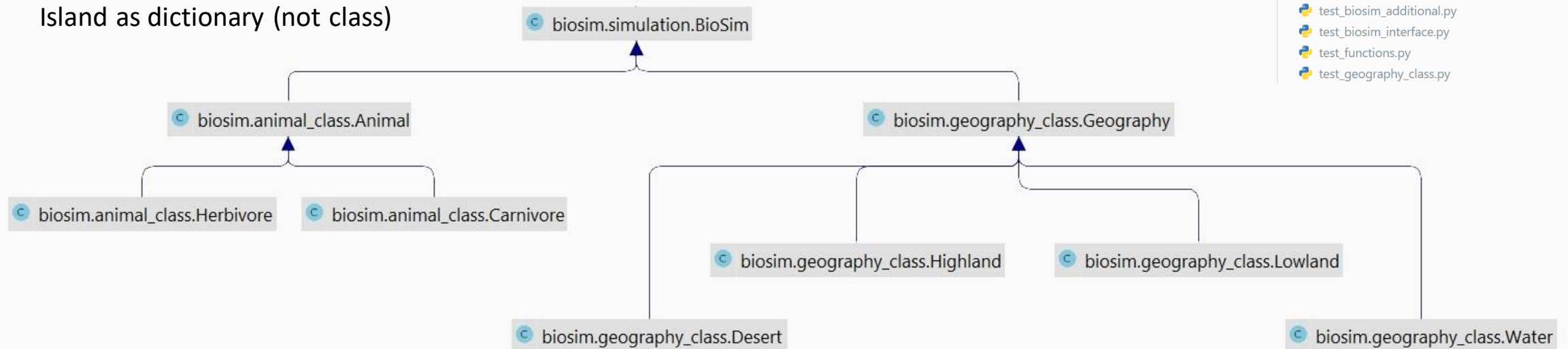
- Alternatives: Class, List, Dict, Set, (Array/Matrix)

Advantages of class-based structure:

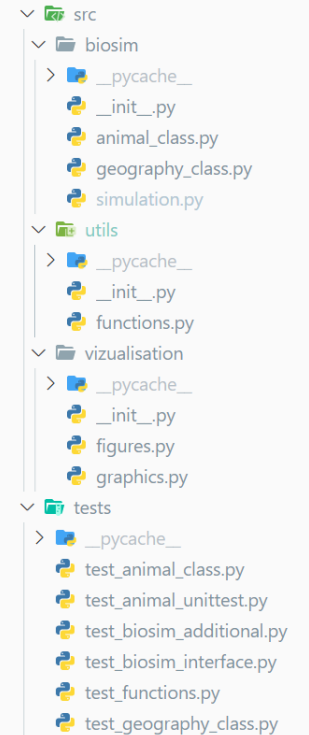
- Inheritance
- 'State' - management

Island as dictionary (not class)

/src/biosim - Class hierarchy



Folder structure



Is it trustworthy?

- How did we ensure quality?
 - Tests, pep8, flake8, tox, etc.
 - CI/CD pipeline with GitLab for automatic tox test-run in docker environment
- Is code maintainable?
 - Classes with Sub-Classes (inheritance)
 - Parametrization and fixtures in test
 - Automatically generated code with Sphinx

Coverage: [pytest in tests](#) ×

100% files, 100% lines covered in 'biosim'

Element	Statistics, %
__init__.py	100% lines covered
animal_class.py	100% lines covered
geography_class.py	100% lines covered
simulation.py	100% lines covered

100% files, 100% lines covered in 'utils'

Element	Statistics, %
__init__.py	100% lines covered
functions.py	100% lines covered

✓ Tests passed: 154 of 154 tests – 4 sec 998 ms

Test Results	4 sec 998 ms
✓ tests	4 sec 998 ms
> ✓ test_animal_unittest	49 ms
> ✓ test_functions	51 ms
> ✓ test_animal_class	31 ms
> ✓ test_biosim_interface	4 sec
> ✓ test_geography_class	140 ms
> ✓ test_biosim_additional	727 ms

```
test_biosim_additional.py::test_log_file_init
test_biosim_additional.py::test_img_years_vis_years[2-3]
test_biosim_additional.py::test_set_landscape_parameters_unknown[D-params1]

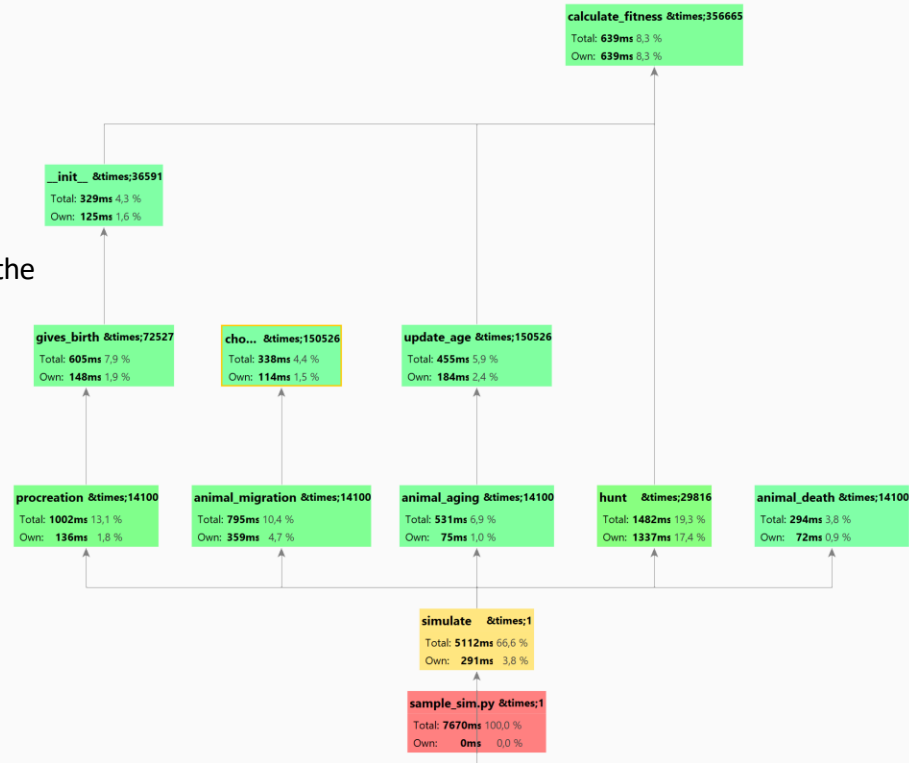
===== 154 passed in 10.41s =====

Process finished with exit code 0
```

Productive code?

- Ease of use
 - Relatively simple user interface (list, dict,)
 - Only necessary to interact with the interface of the BioSim-class
 - Relatively comprehensive documentation
 - Examples and descriptions
- Performance
 - Results, simulations
 - Seems to be sufficiently fast
 - Room for improvements

Profiling – Call Graph



Sphinx – Documentation

Docs • Welcome to BioSim's documentation! [View page source](#)

Welcome to BioSim's documentation!

- Simulation
 - Example
 - The simulation module
- Geography
 - The geography module
- Animal
 - The animal module

Structure of program

Structuring of classes in program with attributes:

Indices and tables

- [Index](#)
- [Module Index](#)
- [Search Page](#)

Next

© Copyright 2022, Mathias & Sindre
Built with Sphinx using a theme provided by Read the Docs.

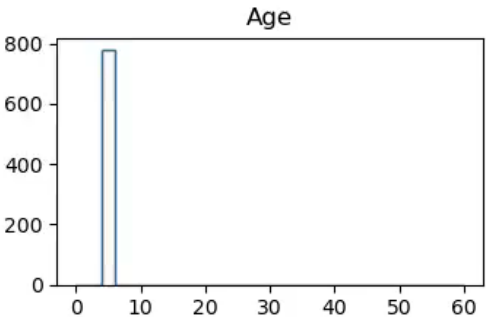
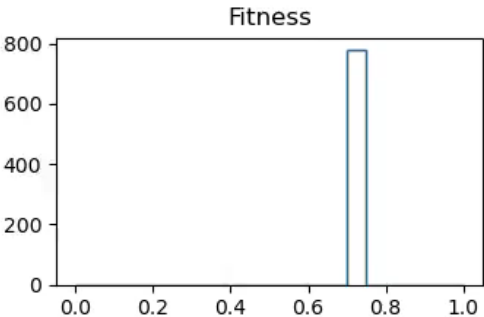
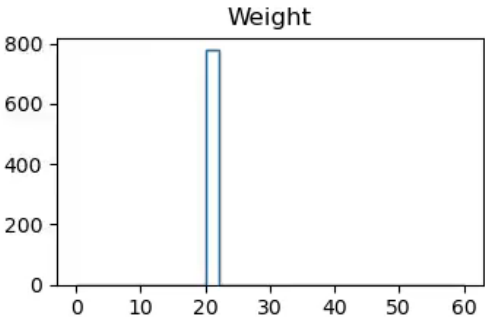
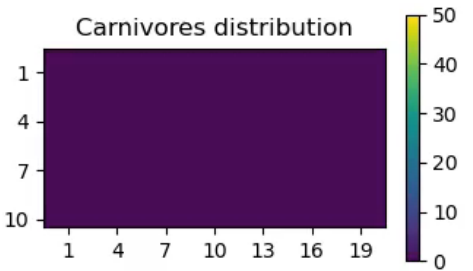
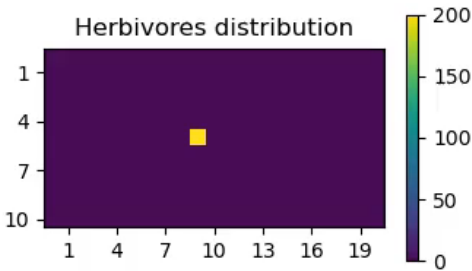
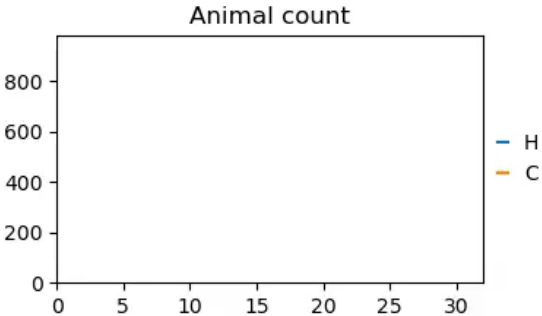
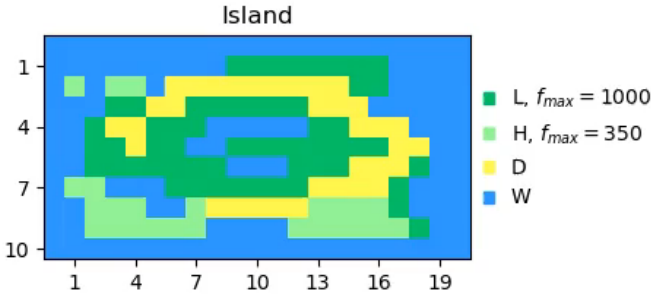
Simulation results

Animation 2

- example_2.py

```
sim.set_animal_parameters('Herbivore', {'zeta': 3.2,
                                         'xi': 1.8})
sim.set_animal_parameters('Carnivore', {'a_half': 70,
                                         'phi_age': 0.5,
                                         'omega': 0.3,
                                         'F': 50,
                                         'DeltaPhiMax': 8.})
sim.set_landscape_parameters('L', {'f_max': 1000})
sim.set_landscape_parameters('H', {'f_max': 350})
```

Year: 1
Animals: 780
Herbivores: 780
Carnivores: 0



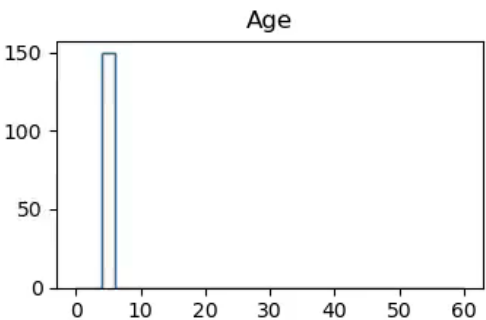
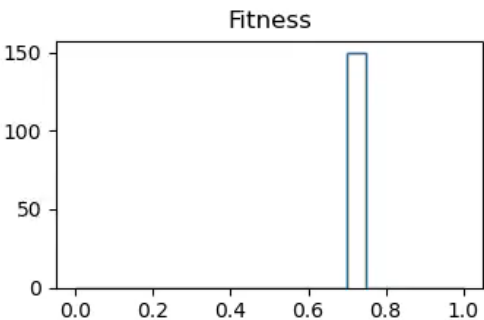
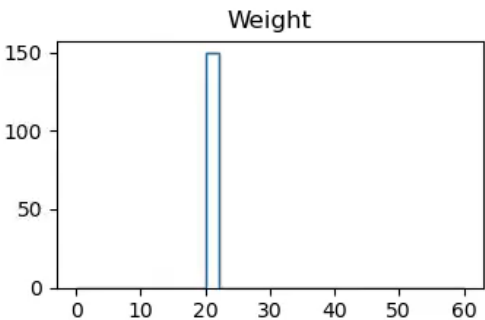
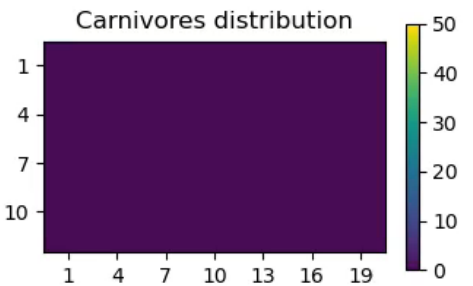
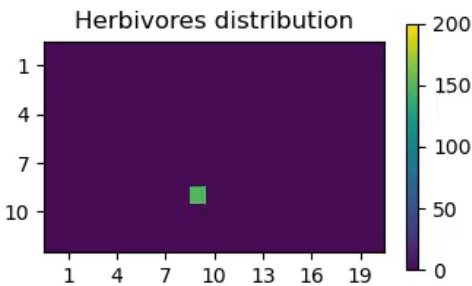
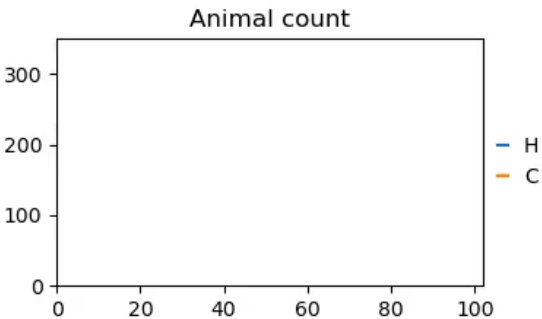
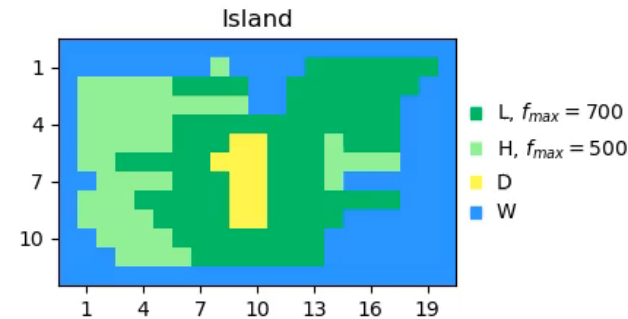
Simulation results

Animation 3

- check_sim.py

```
sim.set_animal_parameters('Herbivore', {'zeta': 3.2,  
                                         'xi': 1.8})  
sim.set_animal_parameters('Carnivore', {'a_half': 70,  
                                         'phi_age': 0.5,  
                                         'omega': 0.3,  
                                         'F': 65,  
                                         'DeltaPhiMax': 9.})  
sim.set_landscape_parameters('L', {'f_max': 700})
```

Year: 1
Animals: 150
Herbivores: 150
Carnivores: 0



Further Development

- Error for invalid animal and geography parameters
- Ensure high quality tests
- Separate Island class (?)
- Profiling and optimization
 - List comprehensions
 - Break out of loops when criteria is met
 - Limit function-calls
 - "Lazy computations"
 - Start with
 - hunt
 - calculate_fitness
 - animal_migration
 - Restructuring?

sample_sim.py (graphics disabled) - before optimization

sample_sim.py		INF20033.pstat	
Statistics		Call Graph	
Name	Call Count	Time (ms)	Own Time (ms)
hunt	29816	1482 19,3 %	1337 17,4 %
calculate_fitness	356665	639 8,3 %	639 8,3 %
cleandoc	32049	644 8,4 %	399 5,2 %
animal_migration	14100	795 10,4 %	359 4,7 %
simulate	1	5112 66,6 %	291 3,8 %
update_age	150526	455 5,9 %	184 2,4 %

sample_sim.py (graphics disabled) - after optimization

sample_sim.py		INF20034.pstat	
Statistics		Call Graph	
Name	Call Count	Time (ms)	Own Time (ms)
hunt	29752	1107 14,2 %	986 12,7 %
calculate_fitness	334189	461 5,9 %	461 5,9 %
cleandoc	32049	672 8,6 %	414 5,3 %
animal_migration	14100	741 9,5 %	292 3,8 %
<built-in method nt.stat>	1474	290 3,7 %	290 3,7 %
<built-in method io.open_code>	274	265 3,4 %	265 3,4 %

~ 27% faster

~ 28% faster