Parallel Java implementation of the PPHPC model

Installing/compiling

Make sure you have installed the Ant tool.

1. Download and unzip code or clone the PPHPC git repository
2. Go into the `pphpc/java` folder
3. Download required libraries, either by

```
$ ant get-libs
```

or (Linux/OSX only)

```
$ cd libs
$ ./getlibs.sh
$ cd ..
```

4. Compile the code

```
$ ant
```

Running the model

Windows users: replace `./pp.sh` with `pp.bat`, and replace `/` with `\`, where applicable.

What are the available options? To see the available options, use the following command:

```
$ ./pp.sh -?
```

Parallelization strategies The parallelization strategies are described in detail here. The following table provides a short description of each strategy.

Strategy	Description
ST	Single-thread (no parallelization)
EQ	Divide simulation environment equally among threads
EX	Same as previous, but allows reproducible simulations (a bit slower)
ER	Threads simultaneously process a row of the simulation environment (sync. at end of row)
OD	Threads continuously process blocks of grid cells while they are available

Examples

Example 1 Run simulation with parameters specified in `config400v1.txt` using the EX parallelization strategy with 8 threads.

```
$ ./pp.sh -p ../configs/config400v1.txt -ps EX -n 8
```

Results are saved by default in file `stats.txt`. It's possible to specify another file with the `-s` option.

Example 2 By default the `OneGoCLI` view is used, which performs a simulation from start to finish without user intervention and without providing any feedback on the simulation status. In order to get a simulation status with progress bar, for longer simulations (e.g. `config800v2.txt`), use the `InfoWidget` view to complement the `OneGoCLI` view.

```
$ ./pp.sh -ps OD -p ../configs/config800v2.txt -v InfoWidget -v OneGoCLI
```

This example uses the OD parallelization strategy with number of threads equal to the number of available processors.

Example 3 Run a single-threaded simulation of `config100v1.txt` using a GUI widget to follow simulation progress and an interactive view to control the simulation.

```
$ ./pp.sh -ps ST -p ../configs/config100v1.txt -v InfoWidget -v InteractiveCLI
```

Example 4 The `pp.sh` script (`pp.bat` in Windows) sets the initial memory to 2GB and the maximum memory to 4GB. For larger simulations, e.g. `config3200v2.txt` you might need more memory. Either edit `pp.sh` or `pp.bat` appropriately, or invoke PPHPC directly using the `java` command:

```
java -Xmx8192m -cp bin:lib/* org.laseeb.pphpc.PredPrey -p ../configs/config3200v2.txt
```

For Windows, this command is slightly different:

```
java -Xmx8192m -cp "bin;lib\*" org.laseeb.pphpc.PredPrey -p ..\configs\config3200v2.txt
```

On the other hand, if you have less than 4GB of RAM, you might want to reduce the maximum memory limit.

Alternative: Using Eclipse

It is also possible to create an Eclipse project in the `pphpc/java` folder and perform all the specified operations within the Eclipse GUI.