Modeling Complex Software Systems Assignment 2

Lien ZHOU 1097463
Fanyu Meng 1026364
Zixin Ye 1101188

Total word count: 1497

Daisyworld Model Overview

Daisyworld is a computer simulation that explores the Gaia hypothesis. It focuses on the interaction between living and nonliving components. In the Daisyworld model, daisies act as the living components and temperature the non-living components.

There are 2 types of daisies in Daisyworld, black and white, with different albedo. White daisies have a higher albedo, which reflects more light, so that absorbed energy is relatively lower, causing the area to cool down. Black daisy however has a lower albedo, hence it absorbs more energy, causing the area to heat up. Both types of daisies have the same life span, and for a daisy to reproduce a certain temperature range must be met. A daisy will die once it reaches its life span.

When the absorbed energy by the ground is lower than the optimal growth temperature, the local temperature where the black daisies grow is higher because of the lower albedo. Since the local temperature is closer to the optimal growth temperature, black daisies will propagate better than the white daisies, where local temperature is lower. As the black daisies increase its population, the overall temperature will increase, putting it beyond the optimal growth temperature. And the white daisies gain advantage. Ultimately this creates a feedback loop, leading to self regulation in the suitable temperature range. It is possible for the temperature to be too extreme so that none of the daisies could survive.

Design of NetLogo Daisyworld

In NetLogo the agents are represented by patches and daisies. A patch's state is determined by 2 components, temperature and daisy which could be black, white or empty (surface). Temperature in a patch represents the local temperature, and the average temperature of all patches represents the global temperature.

Temperature is calculated in each state and is determined by the absorbed energy using the following formula, and not dependent on the previous state.

temperature = (72*ln(absorbed-luminosity) + 80) (degree)

And the absorbed-luminosity is determined by the albedo and the solar-luminosity using the following formula.

absorbed-luminosity = (1 - albedo) * solar-luminosity

The albedos and solar-luminosity are adjustable. By default, surface albedo is 0.4, black daisy 0.25 and white daisy 0.75. Once updated, Once updated, each patch then dissipates 50% of its energy to the neighboring patches.

For each daisy, the reproduction/seeding possibility is determined by the following formula.

Possibility = (0.1457 * Temperature) – (0.0032 * Temperature^2) – 0.6443

Then the monte-carlo method is applied to determine if the seeding is successful. Once successful, a daisy of its kind will spawn in the empty neighboring patch. A daisy has a lifespan of 25, each iteration adds 1, and when it exceeds its life span, its patch becomes empty.

Design of our Java replication & extension

The UML diagram below shows the core architecture design.

Bottom-up system analysis

Class Daisy replicates the Netlogo breed component daisy. Daisy has an attribute age, which records the tick/year a daisy has gone through and an attribute albedo, which records the fraction of energy absorbed from sunlight. And an attribute alive, which turns false when exceeding the maximum age 25. Function updateDaisy replicates the turtle procedure check-survivability. This function first updates the age of the daisy, then checks if the age surpasses the maximum age. If so, the alive attribute is false, and the corresponding patch will dereference this daisy instance and this daisy instance will get GC. Else, the daisy will try to reproduce, using the above-mentioned reproduction/seeding formula, when succeeded, a new instance of this type of daisy will be generated, and put to the neighbouring empty patch.
Class Patch replicates the Netlogo patch. Patch has an attribute localTemp, replicating the Netlogo temperature. For the repetitive neighbours query, Patch implemented an arrayList for iterators to access its neighbours. SurfaceAlbedo records the albedo-of-surface in the Netlogo model. solarLumin records the solar-luminosity in the Netlogo model. The diffuse function

replicates the Netlogo diffuse function, which diffuses 50% of its temperature to the neighbouring patch. The updateTemp uses the above-mentioned temperature formula to calculate the current temperature in a patch.

Class Grond replicates the global component in Netlogo, it constructs the globe via a grid of patches (cascaded arrayList), and manages the state transition of daisy and patch between tick/year, and manages the length of the simulation by the start year, and end year. Here the year replicates the Netlogo tick. Function seeding calls all the daisies' updateDaisy function, if the temperature is suitable, new daisy spawn. Function updatePatchTemp calls all the patches' updateTemp function, which will calculate the patch temperature based on luminosity. Function scenarioEffect replicates the 5 scenarios of the Netlogo daisyworld model, providing the simulation with corresponding luminosity. Function updateGlobalTemp calculates the mean average temperature of every patch, its result stored in variable globalTemp.

Extension

How will the system behave if a daisy could adapt its albedo to the environment? We implemented such a daisy, the yellow daisy, which will increase albedo when the local temperature is over 30°C and will decrease albedo when the local temperature is below °C.

Experiment, result and finding

We documented 3 experiments. The first experiment focuses on checking initial settings, system behaviour, input and output format. The second experiment explores the scenario "ramp-up-and-ramp-down" and observes the global temperature change. The third experiment explores the impact of our extension, the yellow daisy.

Experiment 1

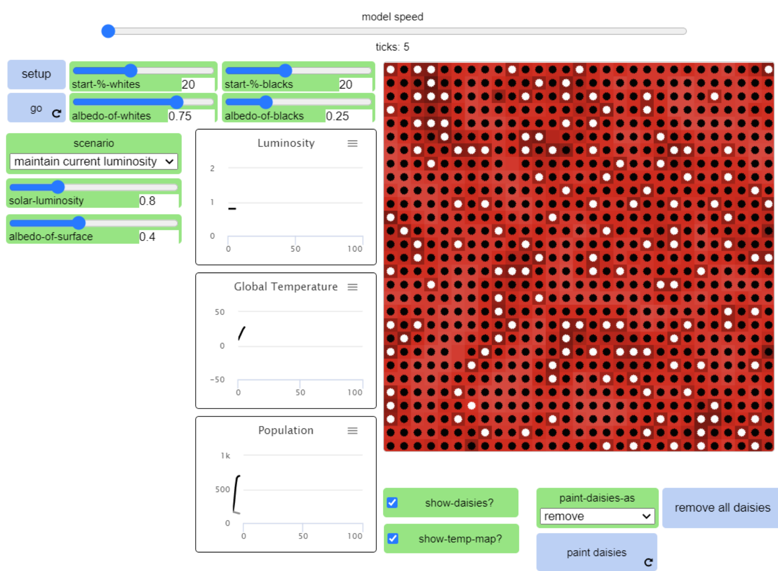
In experiment 1, all the parameters are the default value and have a brief duration to make sure the input has all been included in a correct format and the output's correctness in a good CSV file formatting. Meanwhile, initial stages, seedings and temperature changing are tested.

Parameter settings：
Start-%-white:20; Start-%-black:20; albedo-of-whites:0.75; albedo-of-blacks:0.25
Scenario: maintain; solar-luminosity: 0.8; albedo-of-luminosity: 0.4;
Time: 5 ticks.

The purpose of these experiments is to ensure our system is built as similar as the original system. The CSV output is shown below.

	A	B	C	D	E	F	G	H	I
1	initial states								
2	start-%-wh	start-%-bla	albedo-of	albedo-of	solar-lumi	albedo-of	scenario		end_year
3	20	20	0.75	0.25	0.8	0.4	maintain		5
4									
5									
6	Current ye	White Dais	Black Daisy	Global Ten	Luminosity				
7	0	180	180	8.880368	0.8				
8	1	174	513	13.32055	0.8				
9	2	169	728	18.72303	0.8				
10	3	162	738	23.5184	0.8				
11	4	152	747	26.25047	0.8				
12	5	145	751	28.04702	0.8				
13									
14									

In the first output, we could see that the output has a good format with all the input and information we cared about.



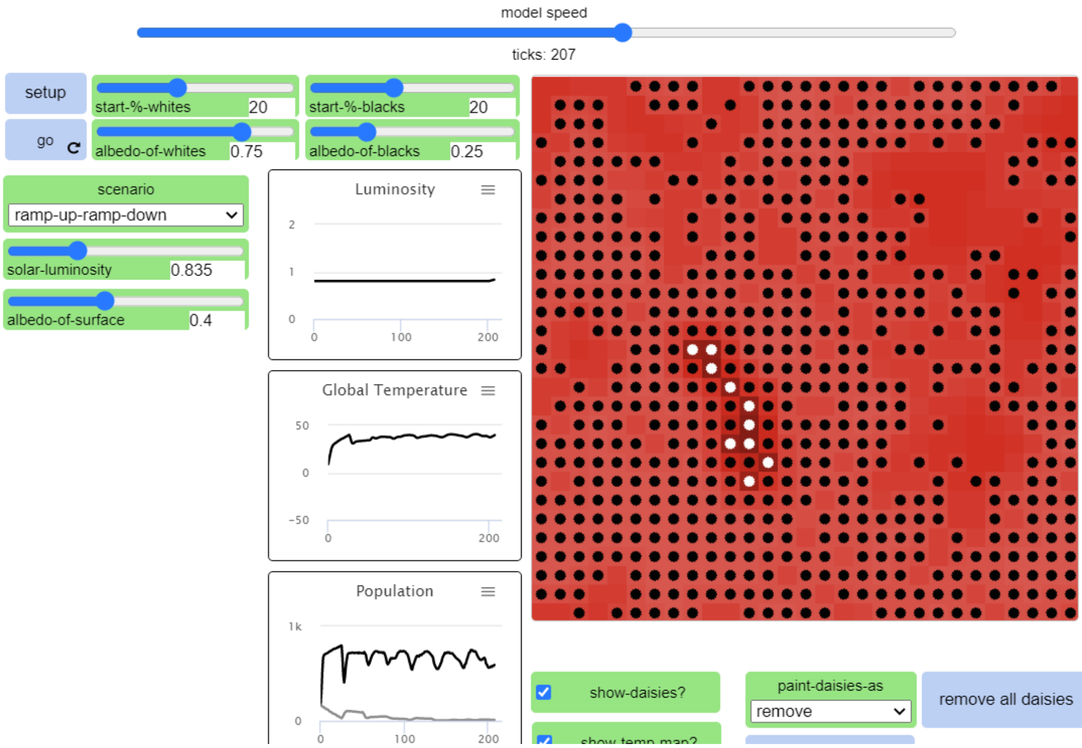
We have produced all the output Netlogo provided except the board. And we have the same trends with Netlogo, which is the number of black daisies increasing, white daisies decreasing, and the global temperature has increased.

Experiment 2

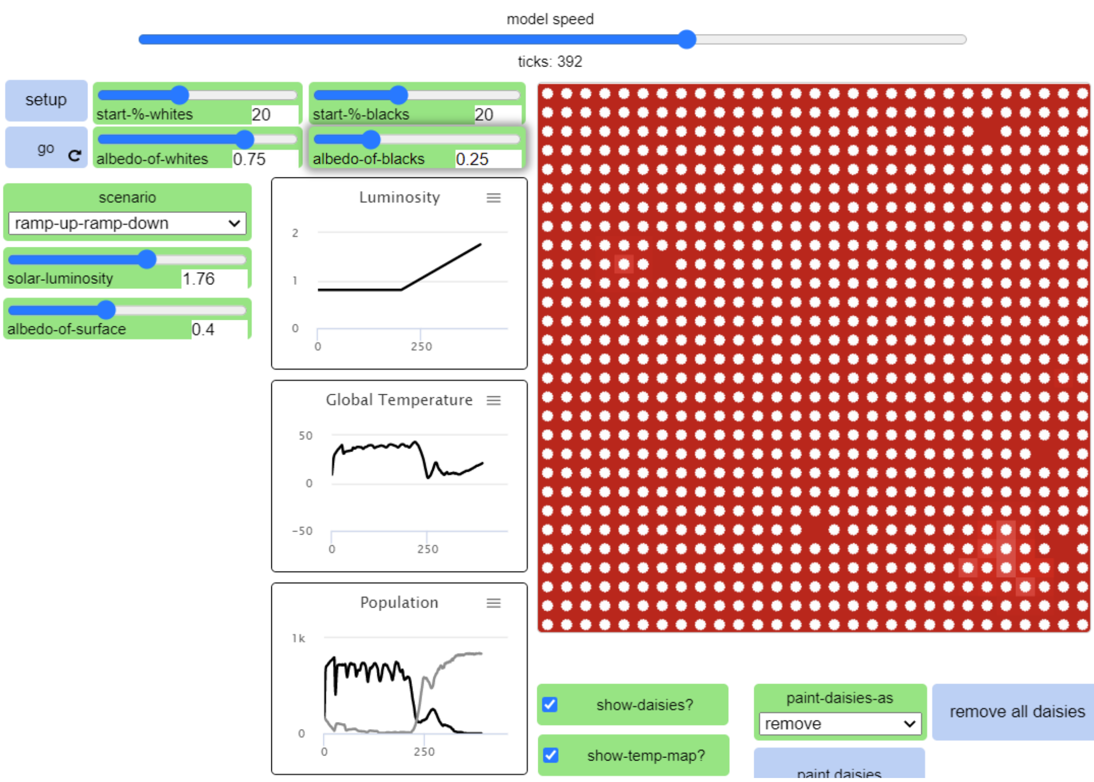
In this experiment, parameters are basically the default values except the scenario is “ramp“ with a much longer time, which is 2000 year, to test if the program performs the same trend with Netlogo.

In the “ramp-up-ramp-down” scenario, the luminosity is initially 0.8, and increases to 1.8 from 200 tick to 400 tick and then maintains. Luminosity decreases when reaching 600 ticks, and maintains at 1.18. Under these conditions, the global temperature will increase and decrease accordingly.

Based on the Netlogo Daisyworld default settings, the temperature would first increase and the number of black daisies would increase with time. With the advantage of black daisies and the high temperature, most of white daisies will die.



When the solar luminosity increases, if some white daisies have survived, their population will increase, whereas the population of black daisies declines and eventually becomes extinct.



Finally, with only white daisies left, the global temperature would become very unstable. The worst case is that all white daisies also die out after the luminosity rebound.

As the output below shown, black daisies would take the advantage of high luminosity quickly, increase the temperature, and grab the living space for white daisies in the first 200 years.

195	6	735	40.32734	0.8
196	7	719	40.09109	0.8
197	8	694	39.79513	0.8
198	9	650	39.38899	0.8
199	9	612	38.75817	0.8
200	9	605	38.10358	0.8

However, after 200, white daisies would bloom because higher luminosity makes the temperature higher and there is no need for black to keep the world warm. That causes black daisies to go extinct.

338	895	2	9.367751	1.49
339	892	2	9.446767	1.495
340	893	2	9.711532	1.5
341	895	1	9.928696	1.505
342	894	1	10.07772	1.51
343	897	0	10.30626	1.515
344	894	0	10.42516	1.52

As a result, not all white daisies die out, but the temperature becomes extremely unstable.

1979	873	0	-7.48934	1.175
1980	852	0	-6.90011	1.175
1981	820	0	-5.8701	1.175
1982	781	0	-4.23449	1.175
1983	728	0	-2.05096	1.175
1984	675	0	0.896802	1.175
1985	628	0	4.226676	1.175
1986	580	0	7.537494	1.175
1987	535	0	10.8738	1.175
1988	493	0	14.1178	1.175
1989	493	0	17.21059	1.175
1990	519	0	18.75698	1.175
1991	538	0	18.61969	1.175

Experiment 3

The parameters we use on our experiment including yellow daisies under maintain scenarios are shown as follows:

1	initial states										
2	start-%-white	start-%-blacks	start-%-yellows	albedo-of-whites	albedo-of-blacks	albedo-of-yellows	solar-luminosity	albedo-of-surface	scenario		end_year
3	20	20	5	0.75	0.25	0.5	0.8	0.4	maintain		600

(Parameters on scenario maintain including yellow daisy)

At the end of the experiment, all white daisies died out and only a small amount of black daisies could survive. Most of the live daisies were yellow daisies. The global temperature remained at a relatively low level.

98	591	0	60	840	16.0281989173726	0.8
99	592	0	61	839	16.0007493706763	0.8
00	593	0	60	840	16.0032432016525	0.8
01	594	0	61	838	15.9882715128163	0.8
02	595	0	60	840	16.0042971349942	0.8
03	596	0	57	842	15.9887984794871	0.8
04	597	0	56	844	15.9396862010323	0.8
05	598	0	57	809	15.8916185952089	0.8
06	599	0	55	819	16.1317607138613	0.8
07	600	0	54	825	16.1610516663647	0.8

(Columns from left to right: current year, the number of white daisy, the number of black daisy, the number of yellow daisy, global temperature, solar luminosity)

We also used another scenario, ramp-up-ramp-down on Experiment 3. The settings are shown below.

initial states										
start-%-white	start-%-blacks	start-%-yellows	albedo-of-whites	albedo-of-blacks	albedo-of-yellows	solar-luminosity	albedo-of-surface	scenario		end_year
20	20	5	0.75	0.25	0.5	0.8	0.4	ramp		1200

initial states										
start-%-white	start-%-blacks	start-%-yellows	albedo-of-whites	albedo-of-blacks	albedo-of-yellows	solar-luminosity	albedo-of-surface	scenario		end_year
20	20	5	0.75	0.25	0.5	0.8	0.4	ramp		1200

(Parameters on scenario ramp-up-ramp-down including yellow daisy)

The results reveal that after 1200 years, only yellow daisies can survive and the global temperature never increased beyond 30 celsius degrees.

1191	0	0	900	28.6138323060378	1.175
1192	0	0	900	28.6124600094156	1.175
1193	0	0	900	28.5758263048104	1.175
1194	0	0	899	28.6081238253027	1.175
1195	0	0	900	28.6232128739334	1.175
1196	0	0	894	28.5909212239014	1.175
1197	0	0	897	29.9963717272822	1.175
1198	0	0	899	30.8022618523599	1.175
1199	0	0	900	30.8511169554636	1.175
1200	0	0	900	30.0437686037131	1.175

(Columns from left to right: current year, the number of white daisy, the number of black daisy, the number of yellow daisy, global temperature, solar lumino

Both of the two scenarios can prove that when a kind of daisy can adjust its albedo somehow, it will have very strong survivability. The global temperature will remain within the range when the daisy will not change its albedo.

Appendix A

Task breakdown

Lien Zhou: Working on any functions and classes related to daisy. Report write up part 1,2,3.

Zixin Ye: Working on any functions and classes related to patches of the ground.

Fanyu Meng: Working on any functions and classes related to simulation initialisation and input/output of simulation. Helping with developing and testing. Report writing for experience 1 and 2.

Success and challenges

Challenges arise when interpreting the semantic of the Netlogo model, our team needs to constantly communicate to ensure each teammate understands the model in the same way. And a minor typo in the key formula caused some time in debugging. Fortunately we overcome these challenges and deliver our Java implementation successfully.

Modifications to original report

1. Our original model suggests there would be 2 classes implemented, Patch and Daisyworld. Patch implements the behaviour of individual agents, while Daisyworld is the driver of agents, as well as calculating the overall result, such as global temperature etc.. In our final daisy world model, now we implement 5 classes. Class Daisy and Patch replicate the behaviour of individual agents in Netlogo, respectively. Class Ground serves as a driver for Patch and Daisy, its function includes environment initialization, agent state update between ticks, and calculation of system status. Class Sim provides the parameters for the simulation reading from the file, and produces the output of the simulation result.

Reference List

- Novak, M. and Wilensky, U. (2006). NetLogo Daisyworld model. http://ccl.northwestern.edu/netlogo/models/Daisyworld. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.
- Wilensky, U. (1999). NetLogo. http://ccl.northwestern.edu/netlogo/. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.
- NetLogo. *NetLogo Help: diffuse.* http://ccl.northwestern.edu/netlogo/docs/dict/diffuse.html.
- NetLogo. *NetLogo Help: neighbors* http://ccl.northwestern.edu/netlogo/docs/dict/neighbors.html