NetLogo Model: Evolution of resource use and social behavior (monitoring, punishment)

Model Overview, Design concepts, Details (ODD)

Model Purpose

What is the purpose of the model?

This model simulates the evolution of a population of agents competing for the same resource. Agents have two types of traits - harvesting preference (sustainable or greedy harvesting), and ability to perceive and punish other agents. Punishing agents have the ability to perceive other agents in their environment to some degree and to react to their behavior. They can kill agents with greedy harvesting behavior, stop them from harvesting in the next iteration, or have them pay a penalty fee to their neighbors. Agents have a cost (energy) to pay for both detection and punishment, so this behavior is altruistic. Punishing agents have the ability to coordinate punishment with a minimum number of punishers to share punishment costs. Due to this social perception and social behavior in the population, greedy harvesting behavior can become less advantageous or even disadvantageous, and sustainable harvesting behavior can spread in the population. Concepts like kin selection, frequency-dependent selection , multi-level selection play a role in the outcomes that are observed.

Entities and variables

What kinds of entities are in the model? By what state variables, or attributes, are these entities characterized? What are the temporal and spatial resolutions and extents of the model?

General environment

Changeable variables:
- *Living-costs:* the costs that each agent has to deduct from his energy per iteration for basic survival
- *Mutation-rate:* The probability with which offspring agents have other traits than their parents
- *Death-rate:* The probability with which agents die independent of their energy level

(these variables could be considered both a factor of the environment and a variable of the agents, but we list them here since they do not vary among agents and are rather "imposed" from the environment)

Patches

There are 60*60 patches in the world.

Constant patch variables (these variables and behaviors are always the same for all of the patches and in each iteration)
- The resources on a patch regrow by a logistic growth function up to the carrying capacity:
  new resource level = current resource level + (Growth-Rate * current resource level) * (1 - (current resource level / carrying capacity)

Changeable patch variables (variables that are changeable by the user on the interface)
- *carrying-capacity*: the maximum amount of resource units on a patch from 1-100
- *growth rate*: the rate at which resources on patches regrow

The maximum sustainable yield is calculated based on carrying capacity and growth rate.

Changing variables during a simulation (variables that change as a result of the simulation)
- actual resource level of a patch in units

Agents

Agents are represented as circles. There can be a maximum of 3600 agents in the world (1 per patch).

Constant agent variables and behaviors (these variables and behaviors are always the same for all of the agents of this type and in each iteration)
- In each iteration, each agent attempts to harvest resources from the patches it is on as well as the eight neighboring patches. If the amount of resources available is lower than

the amount that the agent attempts to harvest, it moves to a neighboring unoccupied patch with the most resources.

Changeable agent variables (variables that are changeable by the user on the interface):

- *Harvest-sustainables:* the amount of resource units harvested by sustainable agents
- *Harvest-greedy:* the amount of resource units harvested by sustainable agents
- *Perception-accuracy*: the probability with which punishing agents notice greedy agents
- *Costs-perception*: the costs in units of energy that punishing agents have to pay in each iteration for perceiving other agents
- *Costs-punishment*: the costs in units of energy that punishing agents have to pay in each iteration for punishing other agents. Costs of punishment are shared between all punishing agents of an agent.
- *Punishment*: the kind of punishing behavior that punishing agents perform
- *Minimum-punishers*: the minimum number of other punishing agents that need to be in the neighborhood of a greedy agent so that punishers will indeed punish.
- *Fine*: if the punishment is “pay fine”, the fine in energy units that punished agents have to pay (shared between all their neighbors)
- *Reproduction:* ability of agents to produce offspring

Changing variables during a simulation (variables that change as a result of the simulation)

- *harvest preference:* the harvest level preference of the agent (sustainable or greedy) as a trait.
- *punisher?:* Whether or not the agent has the ability to perceive and punish other agents
- *harvest-amount*: the actual amount of resource units that is harvested by an agent in the current iteration
- *aware-of-who*: the greedy agents in the neighborhood that the agent is perceiving
- *punished?:* Whether or not the agent is being punished
- *Energy*: the energy level (in resource units) of an agent
- Offspring have a different harvest amount trait from their parents with a probability of *Mutation-rate.*

Model Setup

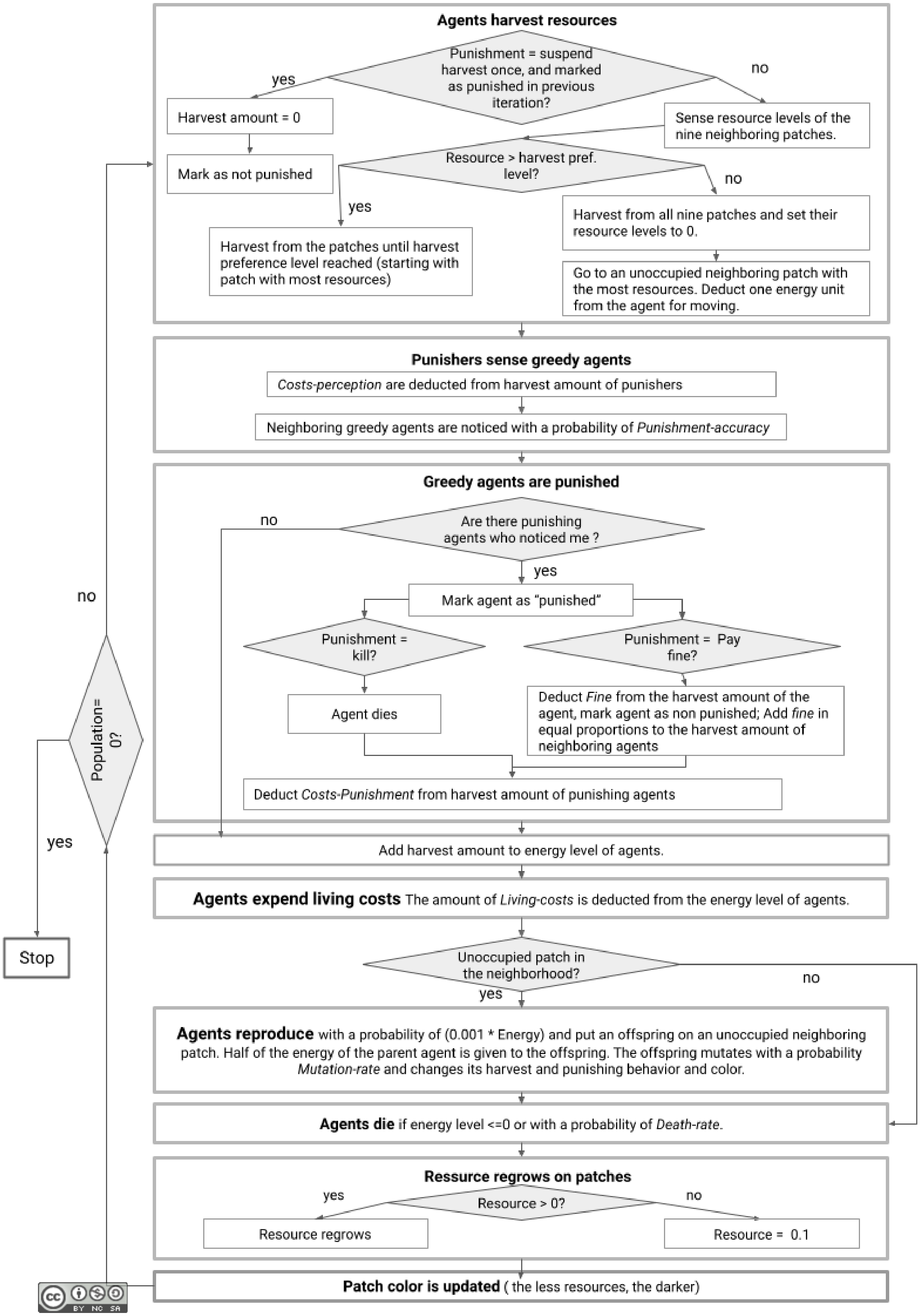
What is the initial state of the model world when one clicks on Setup? Is initialization always the same, or does it vary among simulations?

- A world with 3600 patches is created.
- The initial amount of resource units on a patch is set to *Carrying-capacity*.
- A number of agents, set by the parameter *Number-agents*, are randomly distributed on patches (maximum of one agent per patch).

- A fraction of agents, set by the parameter *Percent-sustainables*, gets the variable *harvest preference* set to “sustainable”
- A fraction of agents, set by the parameter *Percent-punishers*, gets the variable *punisher?* set to true.
- The agents are assigned colors based on their combination of traits:
    - red: greedy, non-punishing
    - orange: greedy, punishing
    - blue: sustainable, non-punishing
    - green: sustainable, punishing
- The initial level of energy of agents is set at *living costs* + 1.

Model Processes

What happens in each iteration? Which entities do what, and in what order? When are state variables updated?


Agents harvest resources
Punishment = suspend harvest once, and marked as punished in previous iteration?
yes
no
Harvest amount = 0
Mark as not punished
Sense resource levels of the nine neighboring patches.
Resource > harvest pref. level?
yes
no
Harvest from the patches until harvest preference level reached (starting with patch with most resources)
Harvest from all nine patches and set their resource levels to 0.
Go to an unoccupied neighboring patch with the most resources. Deduct one energy unit from the agent for moving.
Punishers sense greedy agents
Costs-perception are deducted from harvest amount of punishers
Neighboring greedy agents are noticed with a probability of Punishment-accuracy
Greedy agents are punished
Are there punishing agents who noticed me ?
no
yes
Mark agent as "punished"
Punishment = kill?
Punishment = Pay fine?
Agent dies
Deduct Fine from the harvest amount of the agent, mark agent as non punished; Add fine in equal proportions to the harvest amount of neighboring agents
Deduct Costs-Punishment from harvest amount of punishing agents
Add harvest amount to energy level of agents.
Agents expend living costs The amount of Living-costs is deducted from the energy level of agents.
Unoccupied patch in the neighborhood?
yes
no
Agents reproduce with a probability of (0.001 * Energy) and put an offspring on an unoccupied neighboring patch. Half of the energy of the parent agent is given to the offspring. The offspring mutates with a probability Mutation-rate and changes its harvest and punishing behavior and color.
Agents die if energy level <=0 or with a probability of Death-rate.
Ressurce regrows on patches
Resource > 0?
yes
no
Resource regrows
Resource = 0.1
Patch color is updated (the less resources, the darker)
Population= 0?
no
yes
Stop

Outputs

What kinds of model outcomes can be observed on the interface? How do they emerge from model parameters, agent behaviors, and interactions?

In the world

Resource levels on patches change as they are being harvested and as they regrow. Agents reproduce and die. Since agents place offspring on neighboring patches and do not move unless there are no more resources, clusters of "related" agents tend to form.

Output diagrams and monitors

Populations (% of carrying capacity): The state of the resource and of the agent population in the world as percentage of total carrying capacity; resulting from resource harvesting behavior and resource regrowth, agent reproduction and death.

Trait frequencies: The relative frequencies of agents in the population by trait, resulting from mutations, different reproduction rates, and death

Average harvest per iteration: average harvested amounts of agents per iteration by trait, resulting from harvested resource units, minus costs for monitoring and punishing (for punishing agents), minus fines (for punished agents in case of punishment "Pay fine")

Average energy of agents: Average energy levels of agents by trait, resulting from resource harvest, minus living costs and reproduction, minus energy expenditure for moving.

Concepts and Principles

Which important concepts or principles are represented in the model?

The resource is characterized by two ecologically significant parameters:

- the **carrying capacity**: this is the largest possible amount of resources that can be present on a patch / in a certain area. In ecology, carrying capacity is often represented by the **letter k**. In this model, it is represented by the maximum growth height of the trees. In the real world, carrying capacity is influenced by biotic and abiotic factors like temperature and humidity or availability of other resources.

- the **resource growth rate**: this is the rate at which a resource grows back from one time step to the next. It is often represented in ecology by the **letter r**.

The resource grows along a **logistic (S-shaped, sigmoid) population growth function** according to the following formula:
Resource state at the next point in time = current resource state + (r * current resource state) * (1 - current resource state / k)

Trait variation
Two types of traits vary in the population: harvesting behavior (sustainable or greedy), and ability to notice and punish other agents (present or not present). New trait variants can be introduced randomly in the population through the parameter *mutation rate*.

Reproductive fitness
Agents that have higher energy levels produce more offspring. They have higher fitness relative to agents with lower energy levels.

Inheritance
Agents create offspring and inherit their trait variants to their offspring (with some variability based on mutation rate)

Natural selection
The frequencies of traits in the agent population change as a result of trait variation, differential reproduction (fitness), and inheritance of traits.

Kin selection, multi-level selection, frequency-dependent selection, altruism
The behavior of sustainable resource harvesting as well as of punishment is altruistic because these agents harvest lower amounts of resources such that the resource can be sustained, and they pay costs for noticing and punishing greedy harvesting behavior. In order for this trait to spread in the population through natural selection, altruists need to interact (directly and indirectly, see Interactions below) with other altruists. This is made possible in the model by the fact that agents do not move much and place their own offspring on neighboring patches. This way clusters of (related) altruists benefit from the sustaining of the resource, and they do not have to pay high costs for punishment because there are less greedy agents around. Punishment reduces the fitness benefits of high harvesting levels, promoting the selection of altruistic traits over time.

Sensing and Information Processing:
- Agents sense their environment in a radius of 1 patch: if a patch is occupied, the resource levels on each patch; as well as whether neighboring agents are greedy (with

an accuracy of *Perception-accuracy*).
- Punishing agents store the number of greedy agents they have noticed (aware-of-who) within one iteration (they “forget” between iterations).
- Punished agents store whether they are being punished from one iteration to the next (so they will be suspended from harvesting in the following iteration)

Objectives and goal-directed behavior:
- Agents attempt to harvest their predetermined harvest amount (or until there are no more resources) from the patch they are on as well as the 8 neighboring patches.
- If there aren’t enough resources around, agents move to a neighboring unoccupied patch with the most resources.
- Punishing agents punish greedy agents through the parameters set by the kind of punishment

Learning and Adaptation:
- Agents do not learn and do not adapt their behavior over their lifetimes.

Interactions:
- Agents interact directly through noticing each other and punishing each other.
- Agents interact indirectly through competition for limited resources and space, and through their changing of environmental conditions through resource extraction and placing of offspring

Role of randomness:
- Agents are being distributed randomly in the world at the beginning of a simulation.
- Agents take on their traits (harvest preference and ability to notice and punish) randomly based on the probability *Percent-sustainables* and *Percent-punishers*.
- The order in which agents harvest within one iteration is random.
- Agents move to a randomly selected patch if several patches fulfill the objectives.
- The order in which punishing agents notice greedy agents within one iteration is random.
- Greedy agents get noticed by punishing agents with a probability of *Perception-accuracy*.
- The order at which greedy agents that have been noticed get punished within one iteration is random.
- The order in which agents produce offspring within one iteration is random.
- Agents produce offspring with a probability of (0.001 * Energy).
- Agents place offspring on a randomly selected unoccupied neighboring patch.
- Offspring mutate with a probability of *Mutation-rate*.
- Agents die with a probability of *Death-rate*.

References

Grimm, V., Berger, U., Deangelis, D. L., Polhill, J. G., Giske, J., & Railsback, S. F. (2010). The ODD protocol: A review and first update. Ecological Modelling, 221, 2760–2768. https://doi.org/10.1016/j.ecolmodel.2010.08.019

Wilensky, U. (1999). NetLogo. Center for Connected Learning and Computer-Based Modeling, Northwestern University. http://ccl.northwestern.edu/netlogo.

Author: Susan Hanisch Susan@GlobalESD.org

If you would like to use this material in your classroom, we would be happy if you contact us. For more information and teaching materials, visit our websites:
http://www.GlobalESD.org
http://openevo.eva.mpg.de

This material was developed in collaboration with the department of Comparative Cultural Psychology at the Max Planck Institute for Evolutionary Anthropology.

https://www.eva.mpg.de/comparative-cultural-psychology/education.html