ODD (Overview, Design concepts, and Detail) Analysis of Rationality Model.

Created by Maxim Malikov

August 11th, 2023

# Purpose

The purpose of this NetLogo [1] model is to simulate the interactions of large scale populations of agents with different strategies and utility functions in a mixed scenario environment with additional realistic aspects. The research question that is being examined is whether non-standard utility functions (not maximizing individual’s payoff) survive and perform well in certain environments, despite performing suboptimally in some of them.

# Entities, State Variables, and Scales

## Entities

The simulation contains two types of entities:

1. Patches of grass that represent resources.

2. Bugs (agents) that feed on the grass and interact with each other.

## State Variables (Attributes)

1. Patches have the following attributes:

- grass-amount: This is the resource that agents use.

2. Agents have the following attributes:

- energy: This is the amount of energy that an agent has. When the agent runs out of energy, they die and are erased from the simulation.

- age: Every agent starts at age 0 and gets older with every step, and when they reach the age of 100, they will die.

- strategy: This is the overall strategy that agents use when interacting with each other in the environment. Choices are: dove, hawk, nash, berge, trust.

- choice: This attribute is used to store the choice an agent makes when presented with a game theoretical scenario.

3. There are two global attributes that capture the properties of the Hawk-Dove scenario:

- conflict-total-value: This attribute is used to store the payoffs in the Hawk-Dove scenario when both players choose Hawk.

- conflict-fraction: This attribute is used to calculate the value C in the Hawk-Dove scenario. This is used in calculating the mixed strategy ratio for Nash in the Hawk-Dove scenario.

4. There are also several input variables that can alter the overall behavior of the model:

- number-of-bugs: Governs the number of agents that are spawned to start the simulation

- step-cost: Governs how much energy the agents lose every time they move to another location. Since the agents move every step, it essentially becomes the amount that agents lose every step.

- grass-regrowth-rate: Specifies how much “grass” regrows every step. This variable has a large impact on the carrying capacity of the environment.

- energy-gain-from-grass: How much “grass” the agents consume every time they encounter a patch of grass and turn into their own resources. If the amount of grass on a patch is less than this value, the agent will not collect any grass.

- number-of-SH, number-of-PD, number-of-HD: the number of Stag Hunt, Prisoner’s dilemma and Hawk-Dove scenarios in the environment.

- hawk, dove, nash, berge, trust: these switches determine if one of these strategies participates in the simulation.

## Scales

- Spatial scale: The environment consists of a 50x50 grid of patches, with vertical and horizontal wrapping. The patches are used to represent resources (grass), upon which agents can move freely in any direction and distance. Agent interaction takes place when two or more agents occupy the same patch space.

- Temporal scale: In this simulation, no specific temporal scale is present, however the agents only live for 100 “ticks” of the simulation.

# Process Overview and Scheduling

## Processes

The overall process consists of two components:

- Grass growth: Every patch grows the amount of resources on it defined by the variable grass-regrowth-rate. In turn, the agents that live in this environment consume the grass and replenish their own resources.

- Agent behavior: Every agent moves in a random pattern, from one patch to another. If they do not encounter any other agent, they will simply consume energy-gain-from-grass from the patch and replenish their own energy by that amount.

If the agent does encounter another agent, they will first check if both of them have energy over 20, which would lead them to create an offspring. Each parent of the offspring will contribute 10 energy to the offspring to ensure that they can survive. The behavior of the offspring will be randomly inherited from one of the parents.

If the agents cannot procreate, they will interact randomly according to one of the three scenarios (Stag Hunt, Prisoner’s Dilemma, or Hawk-Dove). The payoffs of these scenarios correspond to the amount of grass that each agent will consume. In case of the negative payoff in Hawk-Dove, no grass will be consumed, and agents will lose energy.

Every step agents age by 1 unit. When they reach 100 units, they will die. If the agents run out of energy, they will also die.

The end result of the simulation is that agents will reach the carrying capacity of the grass within the simulation and remain at the population indefinitely.

## Scheduling

The agents are scheduled in a uniform but random manner. Each agent and patch is activated once per step, but in a random order.

# Design Concepts

## Basic Principles

This simulation explores an iterated environment that contains multiple scenarios. Rather than exploring one specific game theory scenario, the agents have to perform well in a variety of scenarios instead. Because the agents are in a mixed-scenario environment, the strategy approaches that are seen in one scenario cannot be easily translated to other scenarios. Instead, the simulation explores different types of utility functions that agents may wish to maximize, along with the corresponding overall strategy, such as using the Nash equilibrium approach to calculate the optimum strategy for each scenario.

By populating the simulation with agents of different strategies and letting them interact, we can explore which social behaviors perform better in which mix of social interaction types.

## Emergence

One interesting aspect of this simulation is that the performance of the strategies against each other as calculated head to head is not necessarily indicative of their performance on a large scale. This can be explained by the fact that on a large scale, agents not only interact with agents of other strategies, but also with agents of their own strategy, which can favor certain types of strategies in coordination scenarios.

## Adaptation

This simulation does not explore adaptation or evolution of the agents or patches.

## Objectives

This simulation does have any specific objects for the agents.

## Learning

This simulation does not implement agent learning.

## Prediction

This simulation does not implement prediction.

## Sensing

The agents do not perceive the environment beyond knowing the full payoff values for each of the game scenarios that they encounter.

## Interaction

If there is only one agent on a patch, then this agent interacts with patches (“grass”) by consuming the amount “energy-gain-from-grass” from the patch. That is, if this variable amount is greater than the current available amount of resources on the patch, the agent will increase their “energy” attribute by that amount, and the patch will decrease its “grass-amount” attribute by the same amount.

If there are two agents on a patch, and each agent’s “energy” attribute is greater than 20, then these agents will reproduce. This means that a new agent will be created, with the “strategy” attribute randomly selected between the two parents. This new agent will also be given 20 “energy” by its parents: the “energy” attribute of the new agent will be set to 20, and the “energy” attribute of each parent will be reduced by 10.

If there are two agents on a patch, but at least one of them has less than 20 “energy” attribute, then instead the agents will interact through one of the 3 game theory scenarios: Prisoner’s Dilemma, Stag Hunt or Hawk-Dove. The ratio of these three scenarios is determined by the “number-of-HD”, “number-of-SH”, and “number-of-PD” sliders. The payoff from these scenarios determines the amount of “energy” that each agent will gain from the grass patch – this is done instead of the normal interaction where the agent collects the full value of “energy-gain-from-grass” when they are alone on the patch. The payoffs for each scenario are as follows – note that V is equivalent to “energy-gain-from-grass” external variable:

|  |  |  |  |
| --- | --- | --- | --- |
| **PLAYER 1** |  | **PLAYER 2** | |
|  | **Hawk** | **Dove** |
| **Hawk** | -V  -V | 0  V |
| **Dove** | V  0 | 1/2V  1/2V |

**Figure 1**. Hawk-Dove payoffs used in the simulation. V = amount of resources gained from grass in one turn. C = 3V.

|  |  |  |  |
| --- | --- | --- | --- |
| **PLAYER 1** |  | **PLAYER 2** | |
|  | **Stag** | **Hunt** |
| **Stag** | V  V | 3/4V  0 |
| **Hunt** | 0  3/4V | 1/2V  1/2V |

**Figure 2**. Stag Hunt payoffs used in the simulation. V = amount of resources gained from grass in one turn.

|  |  |  |  |
| --- | --- | --- | --- |
| **PLAYER 1** |  | **PLAYER 2** | |
|  | **Cooperate** | **Defect** |
| **Cooperate** | 1/2V  1/2V | V  -1/2V |
| **Defect** | -1/2V  V | 0  0 |

**Figure 3**. Prisoner’s Dilemma payoffs used in the simulation. V = amount of resources gained in one turn.

When there are more than 2 agents on the same patch, each agent will select a random agent among the others, and interact with that agent only. This includes the processes of reproduction and social interactions.

## Stochasticity

This model is highly stochastic.

- Agents move around in a random manner

- When there are multiple agents on the same patch, each agent selects a random other agent to interact with

- The scenario used for interaction between agents is randomly selected

- In the case of reproduction, the strategy of the newborn agent is selected randomly among the two parents.

## Collectives

While agents do not form groups or collectives, they do potentially share a strategy. However, no special considerations are made to account for other agents that with similar strategies.

## Observation

This simulation records the counts of agents over time. Due to the stochastic nature of the simulation, multiple runs are needed to establish patterns with a certain degree of certainty. Therefore this simulation also uses BehaviorSpace tool to capture counts of agents over time across multiple runs.

# Initialization

The default settings for the model are as follows:

- number-of-bugs: 1200

- step-cost: 2.0

- grass-regrowth-rate: 1.0

- energy-gain-from-grass: 4.0 (should be larger than step-cost)

- number-of-SH, number-of-PD, number-of-HD: Can be varied. One recommendation is 1 for number-of-PD, 7 for number-of-HD and 22 for number-of-SH.

- hawk, dove, nash, berge, trust: in a mixed scenario environment, only berge, trust and nash should be enabled. Enabling these three strategies is also the recommended setting.

# Input Data

This model does not use input data – all initialization is done through the model interface.

# Submodels

This model does not contain any submodels.

# References

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

2. Grimm, Volker, Uta Berger, Donald L. DeAngelis, J. Gary Polhill, Jarl Giske, and Steven F. Railsback. "The ODD protocol: A review and first update." *Ecological Modelling* 221 (2010): 2760-2768.

# Appendixes

Source code (NetLogo):

