



November, 2021

ATHENS course : **TP-09**

Social Emergence in Complex System

Micro-study

teaching.dessalles.fr/SECS

Names: **Antonino Orofino** and **Mathieu Serandour**

Green Beard with a nasty gene

Abstract

Having a certain characteristic is usually either good or bad for an individual, however, this isn't always true. Indeed, there can be features for which the dividing line between benefits and obstacles is not so defined. We focused our study in analysing this scenario. In particular, we studied the conditions that led to the spread or disappearance of these kind of features.

Problem

We thought that a feature can be profitable, but still not accepted by peers. For example, having a green beard and being altruistic with other green beard carriers is profitable as we saw in the lab, though it can lead to secondary effects: people might find you less attractive or mock your beard. We wanted to study if the gene that carries a feature like this one could prevail even though it leads to push back from another part of the population, and what conditions rule.

Method

ANTONINO

We chose to begin with the scenario "GreenBeard". First of all, we added a new gene called 'Nasty'. Both the GreenBeard gene and the Nasty one are represented by a single bit. According to the genes' values there can be 3 kinds of agents:

- **Green Beard carriers:** an agent with only the GreenBeard gene;
- **Nasty carriers:** an agent with only the Nasty gene;
- **GreenBeard-Nasty carriers:** an agent who has both genes.

Actually there is also a fourth kind of individual: the one with neither one of two genes.

In order to understand the model we should analyse what happen to an individual when he meet another agent. Each behaviour is related to the genes.

- **Green Beard carriers**
 - altruist among other green beard carriers.
- **Nasty individuals**
 - hate the Green Beard carriers, so steal scores from them by attacking them;
 - they receive a penalty for being nasty.
- **GreenBeard-Nasty carriers**
 - receive a penalty for have betrayed another agent of the same group.

MATHIEU

The original scenario only had two parameters: `GB_cost` and `GB_gift`. These parameters are used to control the value/cost of being altruist among GreenBeard carriers. Then we added other 4 parameters:

- `N_Payback` defines the payback for a nasty after attacking a GreenBeard carrier;
- `N_Attack` defines the cost for a GreenBeard carrier for being attacked;
- `N_Penalty` is used to specify the penalty given to a Nasty individual for behaving in a bad way;
- `GB_N_Penalty` defines the penalty given to a GreenBeard carrier for behaving nastily to anther individual of the same group.

These parameters were added both in `EvolifeConfigTree.xml` and `Expe/GreenBeard.evo` file.

ANTONINO

We can summarize all interactions behaviours in table 1. Let's consider an interaction between an individual and a partner. We represent the individual using columns while the rows for the partner. The cell identified by the individuals couple give us the payoff (or cost if negative) for both the agents that meet each other.

	GreenBeard	Non-GreenBeard
GreenBeard	(-GB_Cost,GB_Gift)	(0,0)
Nasty	(N_Payback - N_Penalty, - N_Attack)	(-N_Penalty,0)
Nasty-GreenBeard	(N_payback - N_Penalty -GB_N_Penalty, - N_Attack)	(-N_Penalty,0)

Table 1: Payoff table

To implement this logic, we chose to overload the `interaction` function in `S_GreenBeard.py`.

```

1  def interaction(self, indiv, partner):
2      """ Genes control the behaviour of 'indiv' toward 'partner'
3          """
4      if (not indiv.gene_value('GreenBeard')) and indiv.gene_value('
        Nasty') and partner.gene_value('GreenBeard'):

```

```

5     indiv.score(self.Parameter('N_Payback'))
6     partner.score(-self.Parameter('N_Attack'))
7     elif indiv.gene_value('GreenBeard') and (not indiv.gene_value('
    Nasty')) and partner.gene_value('GreenBeard'):
8         indiv.score(-self.Parameter('GB_Cost'))
9         partner.score(self.Parameter('GB_Gift'))
10    elif indiv.gene_value('GreenBeard') and indiv.gene_value('Nasty
    ') and partner.gene_value('GreenBeard'):
11        indiv.score(self.Parameter('N_Payback'))
12        indiv.score(-self.Parameter('GB_N_Penalty'))
13        partner.score(-self.Parameter('N_Attack'))
14    if indiv.gene_value('Nasty'):
15        indiv.score(-self.Parameter('N_Penalty'))

```

Listing 1: Overloading of interaction function

MATHIEU

Finally, we thought of adding a new feature that would make our model closer to reality. In real life, in fact, a GreenBeard carrier would not choose a Nasty individual as a partner for reproduction. In order to reach this goal we added another parameter, called `N_Segregation`, and overload the method `parents` only if this parameter is `True`.

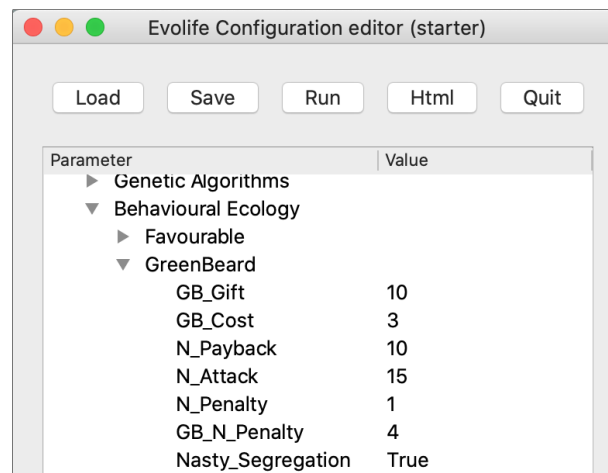


Figure 1: List of the parameters in Evolife

```

1  def parents(self, candidates):
2      """
3      If it is set to True, then GreenBeard carriers will not
4      reproduce with nasty carriers.
5      """
6      if not self.Parameter('Nasty_Segregation'):
7          return super().parents(candidates)
8      try:
9          for i in range(10):
10             m = random.choice(candidates)
11             f = random.choice(candidates)

```

```

11         if m[0].gene_value('Nasty') and f[0].gene_value('GreenBeard
12             '):
13                 continue
14         if f[0].gene_value('Nasty') and m[0].gene_value('GreenBeard
15             '):
16                 continue
17         return (m,f)
18     return None
19 except IndexError: return None

```

Listing 2: Overloading of parents function

Results

In all the results, the red curve represents the rate of nasty gene in the population while the green one represents the rate of green beard carriers. We chose to make all the tests with a **Selectivity** value of 10 and no **SelectionPressure**, as we saw in the lab that it promotes altruism.



(a) Graph

▼ GreenBeard	
GB_Gift	10
GB_Cost	3
N_Payback	10
N_Attack	15
N_Penalty	1
GB_N_Penalty	4
N_Segregation	False

(b) Parameters

Figure 2: Spread of the genes with N_Segregation effect

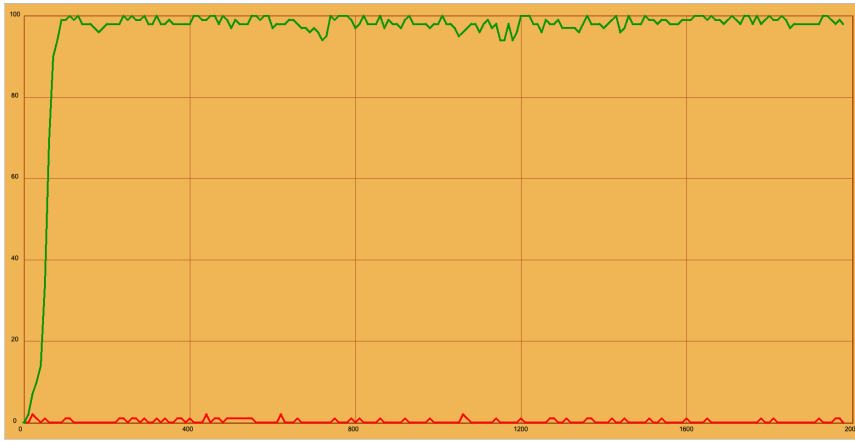


(a) Graph

▼ GreenBeard	
GB_Gift	10
GB_Cost	3
N_Payback	10
N_Attack	15
N_Penalty	1
GB_N_Penalty	4
N_Segregation	True

(b) Parameters

Figure 3: Spread of the genes without N_Segregation effect

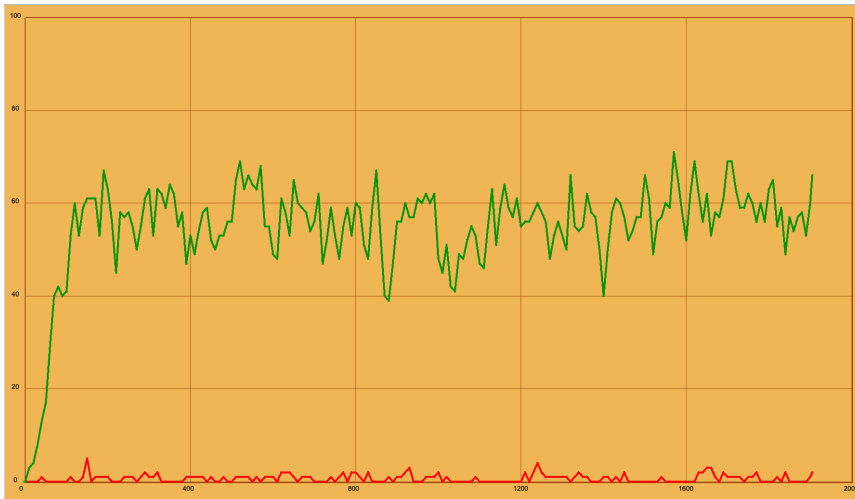


(a) Graph

▼ GreenBeard	
GB_Gift	10
GB_Cost	3
N_Payback	1
N_Attack	15
N_Penalty	1
GB_N_Penalty	4
N_Segregation	True

(b) Parameters

Figure 4: Spread of the genes with a low N_Payback

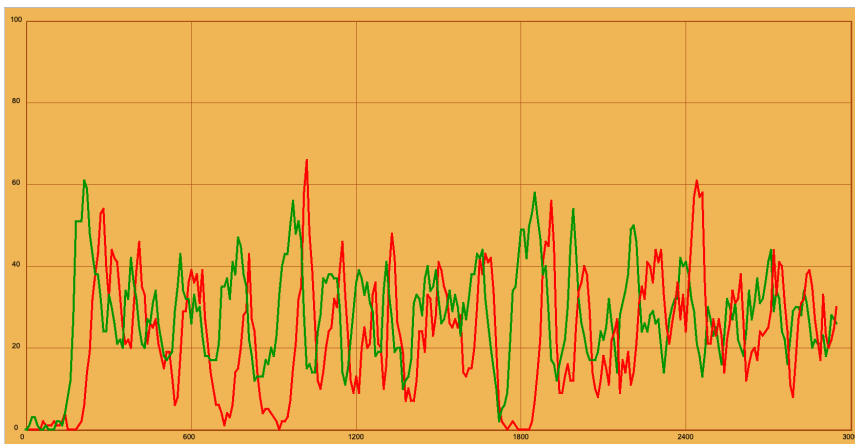


(a) Graph

▼ GreenBeard	
GB_Gift	3
GB_Cost	10
N_Payback	0
N_Attack	15
N_Penalty	1
GB_N_Penalty	4
N_Segregation	True

(b) Parameters

Figure 5: Spread of the genes with N_Payback equals to zero and GB_Cost > GB_Gift



(a) Graph

▼ GreenBeard	
GB_Gift	3
GB_Cost	10
N_Payback	2
N_Attack	15
N_Penalty	1
GB_N_Penalty	4
N_Segregation	True

(b) Parameters

Figure 6: Spread of the genes with a bit of N_Payback ($> N_Penalty$) and GB_Cost > GB_Gift

Discussion

ANTONINO

Figure 2a

If there is too few green beard carrier in the population, we expect the red curve to fall as there is no one to be nasty against, and being nasty has a cost. We also expect the red curve to rise if there is enough green beard carriers, as it is profitable to be nasty if there is a lot of individuals to be nasty against.

As the put a gift way bigger than the cost for the green beard carriers, we also expect that the green curve rises up as long as the red one is not too high. Thus, we expected an oscillating behaviour.

The results are quite similar to our expectations. However, We were surprised to observe that the sum of the two curves was higher than 100: it means that a non negligible proportion of the population presents both genes. It is surprising because the penalty for being both nasty and green beard carrier was quite high: `GB_N_Penalty=4` and `N_Penalty=1`.

Figure 3a

It seems more intuitive to us that the nasty individuals don't want to reproduce with the green beard carriers, that is why we added the `N_Segregation` parameter. When setting it to `True`, we could expect that if the green beard carriers dominates the population, it would be hard for nasty individuals to appear as they don't reproduce with green beard carrier.

The results are quite similar to the the last during some intervals: the two genes struggle to dominate the other, but eventually the green beard gene becomes dominant and prevails for a pretty long time. When it eventually decays due to random mutations, the struggling comes back.

We were surprised to see that the sum of the two curves always is 100: this time, no one have both genes, which is explainable because no nasty individual reproduces with a green beard carrier and the penalty for being both nasty and green beard carrier is still high. The surprising fact is that no one don't present any of the two genes. It is always better to be either nasty or green beard carrier than to be neutral.

MATHIEU

Figure 4a

We tried to put a high value of the nasty stealing for the green beard carriers along with a small value for the their payoff. At first sight, we expected some struggling between the two genes still as the payback was equal to the penalty for being nasty, so being nasty should be neutral when the green beard rate is very high. Besides, the presence of nasty individuals should still diminish the number of green beard carriers. Though we observe that the red curve never goes above zero. This is because the altruism of the green beard gene is more beneficial than being neutral.

We obtained the exact same curves when putting different values to `N_Payback`, up to 9. Though we expected that some nasty individuals would remain as their payback is superior to the penalty. This is because we put `Selectivity` to 10 and `SelectionPressure` to 0: this configuration promotes elitism and therefore the green beard carriers that did not cross any nasty individual but got gifts from other green beard carriers would gain more than the nasty individuals would get from steeling.

Figure 5a

This curve is a reminder of what we observed in the corresponding lab. When $GB_Cost < GB_Gift$ for the green beard carriers, we would expect the green curve to stay to 0, but we observe that around 60% of the population presents the green beard gene. We get the same curves if the gift is very low. This is for the same reason as the preceding discussion: the **Selectivity** parameter promotes elitism. A few green beard carriers get gifts and never give some. They still get points so they get sectioned.

Figure 6a

Figure 5 shows that when the natural selection promotes elitism, an altruism gene can prevail even when the Cost is superior to the Gift. Figure 2 also shows that the altruism gene can prevail even when the altruist individuals are targets to nasty behaviour.

Figure 6 is an interesting curve: it shows that when we sum these two factors: a cost superior to the gift and nasty behaviours against green beard carriers, altruism is impacted. We put a small payback, and observe that the green curve is much lowered in comparison to Figure 5. This is because even though selection is done in an elitism manner, fewer individuals both get gifts and never cross any nasty individual.

To conclude, in the case we focused on: nasty individuals don't reproduce with altruist green beard carriers, we can say that the condition that assures that altruism prevails is that the gift is higher to the cost. When it is not the case, as long as nasty individuals get satisfaction from bullying the altruists, altruism is to diminish.