Opdracht 2 NatComp

l.nies s4136748, g.zuidhof s4160703

8 March 2015

1 Nash equilibria and ESS's

a There are two nash equilibria for this payoff matrix: [S,S] and [H,H].
Both S and H are evolutionary stable strategies. When a minority plays the other strategy, this minority would do worse than the majority and thus be unable to invade.

b The Nash equilibria for this payoff matrix is [S,S]. The strategy S is evolutionary stable.

c In payoff matrix there are two Nash equilibria, [S,H] and [H,S]. Neither S and H are evolutionary stable strategies.

$P(S) = 0 * (1 - x) + -1 * x = -x$
$P(H) = -20 * (1 - x) + 1 * x = -20 + 21x$

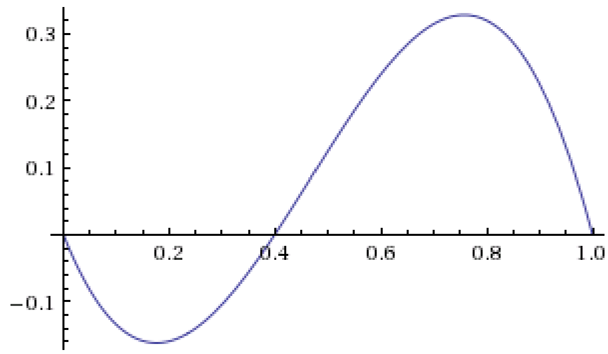
d If the nash equilibria are opposing, there is no ESS. Otherwise, it seems that in these problems the Nash equilibria's strategies coincide with the evolutionary stable strategies.

2 Replicator Dynamics

a
$P(S) = 5p + 0(1 - p) = 5p$
$P(H) = 2(1 - p) + 2p = 2$

$P(average) = p * P(S) + (1 - p) * P(H)$
$= 5p^2 + (1 - p) * 2$
$= 5p^2 + 2 - 2p$

b
$equ = p(1-p)(P(S) - P(H))$
$= p(1-p)(5p-2)$
$= (p - p^2)(5p-2)$
$= -5p^3 + 7p^2 - 2$



c

d $p(1-p)(5p-2) = 0$
$p = 0|1 - p = 0|5p - 2 = 0$
$p = 1|5p = 2p = 0.4$

So the fixed point are $p = 0, p = 0.4 and p = 1$. If $p = 0.4$ then neither strategy is dominant and both will continue to exist next to each other. If $p < 0.4$, strategy H is dominant $(P(S) < P(H))$ and if $p > 0.4$) strategy S is dominant $(P(S) > P(H))$.

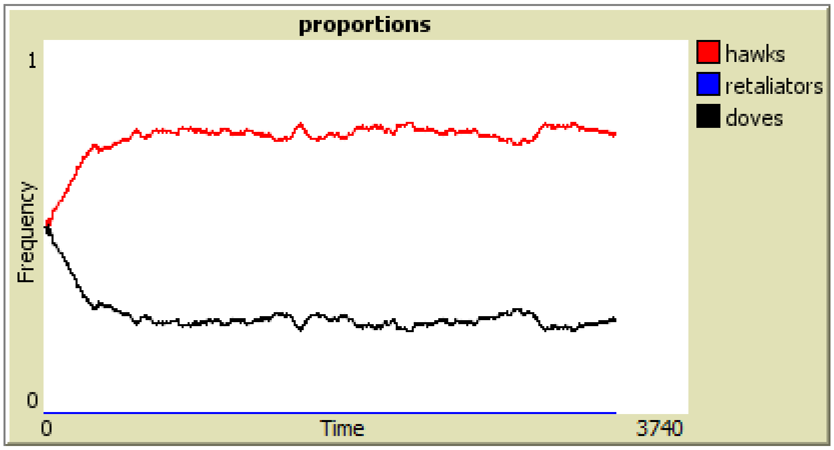
3 Experimenting with existing software

a *First setting: 50 percent doves, 50 percent hawks. Reward value 5, cost of fighting 2*

The hawks quickly dominate the population. With a reward value of 5, and a cost of only two it seems to be worth fighting for half the reward at such a low cost. A dove takes no reward when encountering a hawk, and they quickly disappear.

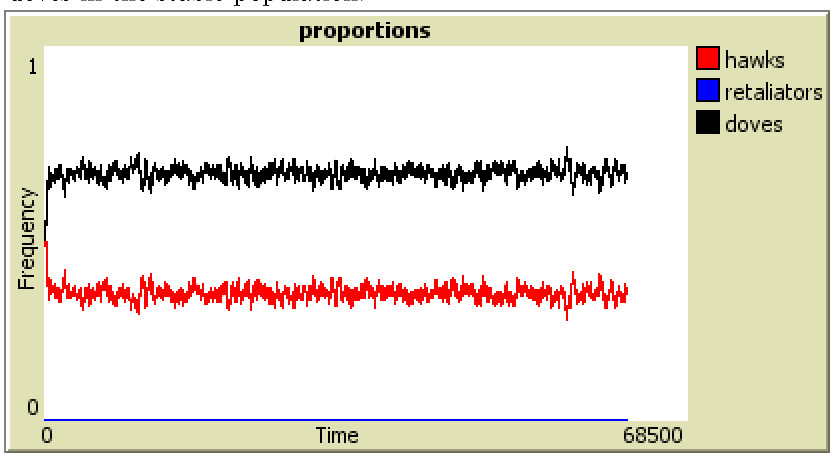
Second setting: 50 percent doves, 50 percent hawks. Reward value 3, cost of fighting 2

A stable balance is found between the doves and hawks. Two out of three appear to be hawks in the population. There is a balance between fighting and sharing, sometimes fighting pays off, sometimes it doesn't, and thus there is this balance between the two strategies.



Second setting: 50 percent doves, 50 percent hawks. Reward value 2, cost of fighting 3

A similar balance is found, however the tables are turned. Two out of three are doves in the stable population.

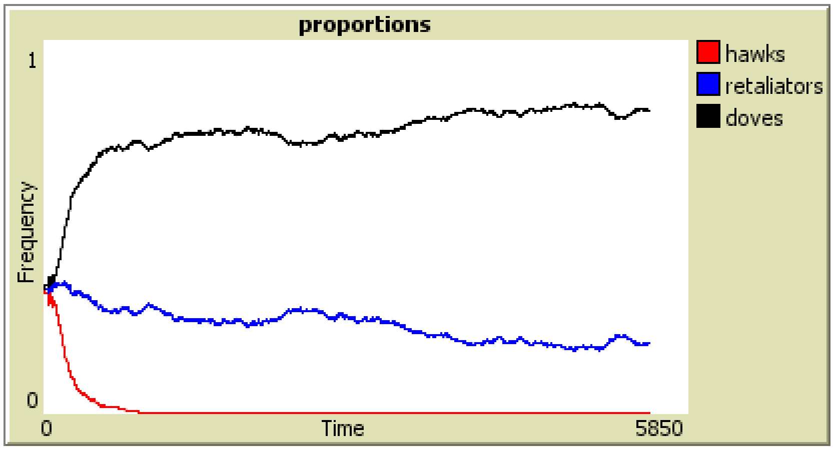


Retaliators are introduced, in the following initial populations the rewards are always 5, and the cost of fighting always 2.

Setting: 33 percent doves, 33 percent hawks, 33 percent retaliators.

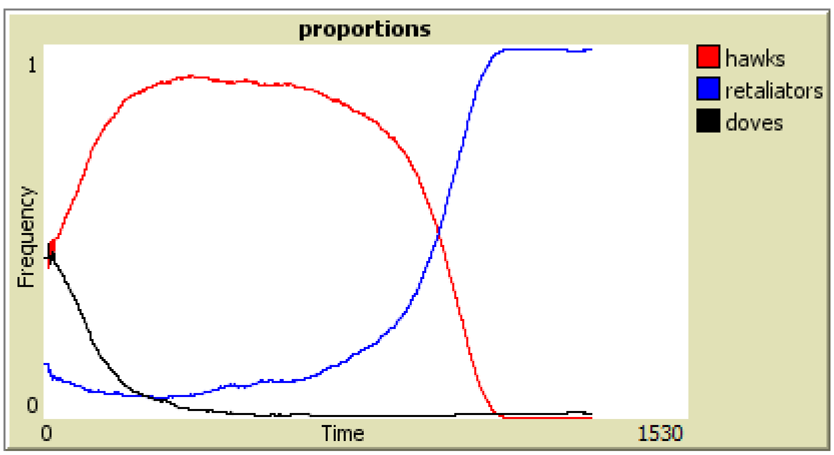
Things get more interesting here! The retaliators seem to kill off the hawks quite rapidly. This must be because they adapt to the competing creature, and do not have the disadvantage of fighting with their own species that the hawks have.

The doves and retaliators later co-exist, as they share the same strategy among eachother.



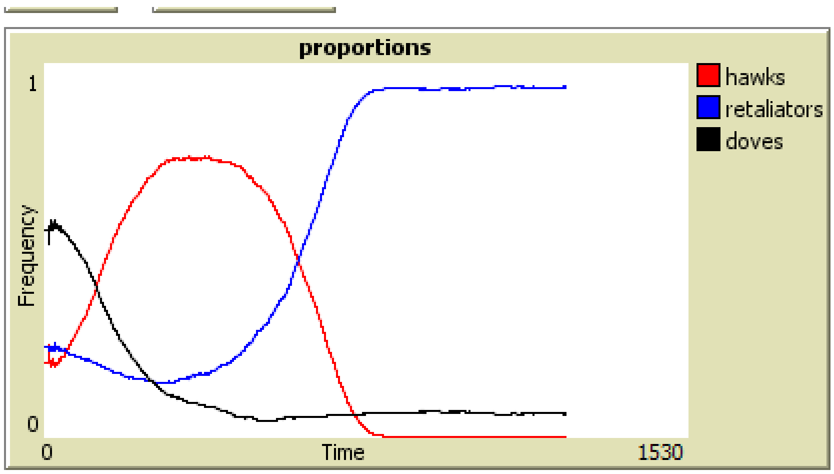
Setting: 33 doves, 33 hawks, 11 retaliators shares in population.

This is again very interesting! Depending on chance the retaliators are able to "kill off" the hawks before all the doves are gone. Sometimes some doves are left, sometimes none are left.



Setting: 25 doves, 9 hawks, 11 retaliators shares in population.

The same as in the previous settings occured, however with these settings the retaliators usually kill off the hawks in time for some doves to remain.



Setting: 25 doves, 0 hawks, 25 retaliators shares in population.

They simply co-exist, as they share the same strategy.

b The pro's and cons of using a program like NetLogo.

Pro's:

- It is already built. Instead of having to build a system for simulations, NetLogo can be used for a multitude of multi-agent problems, as long as it can be described in a n2logo script.
- Easy to get started, tutorials exist. Can maybe even be used by those who are afraid of programming.
- Simulating a model can form empirical proof for one's calculations of, for instance, ESS's.
- NetLogo is built upon the JVM, which makes it possible to run on many platforms (cross-platform).

Con's:

- Requires one to specify the problem in a netlogo specific script, which requires some learning.
- Performance may be lower than an implementation that is closer to the metal (such as in a language compiling to machine code).