Unit 4: Cooperation

READINGS AND RESOURCES

- Axelrod R, Hamilton WD (1981) The evolution of cooperation. *Science* 211: 1390-1396.
- Nowak MA (2006) Five rules for the evolution of cooperation. *Science* 314: 1560-1563.
- Cronk L (2015) Human cooperation: Evolutionary approaches to a complex phenomenon. *Handbook on evolution and society: Toward an evolutionary social science*, 441-459.

The problem of cooperation

There are obvious benefits to helping one another and working together. All the major transitions in the evolution of life on earth have involved new cooperative structures (e.g., multicellularity, social animals, symbolic communication). Through cooperation, humans can accomplish amazing things. It's clear that our success as a biological and cultural species comes from the things we can do together.

However, it's not obvious how cooperation can be sustainable. If everyone works to benefit others, those that free ride on others get the benefit without paying the cost. How can cooperation emerge in a population? Once present, how can it be maintained?

A simple model for this principle problem of cooperation is a two-player game called **the prisoner's dilemma**. Here's the *payoff matrix*, with the payoffs shown going to the row player. Each player can either cooperate or defect. Cooperation incurs a cost, c on the cooperator, and confers a benefit, b, on the recipient, where $b > c$. This requirement makes the game positive sum: mutual cooperation creates a positive benefit for both parties. Here is the dilemma: mutual cooperation is better than mutual defection, but the best move is always to defect in a single game.

	Cooperate	Defect
Cooperate	$b - c$	$-c$
Defect	b	0

We can reframe the problem of cooperation in terms of the prisoner's dilemma game: How do we get the emergence and maintenance of cooperation in a population of individuals playing the prisoner's dilemma? To answer this question, we need to talk a little bit about evolution.

Evolutionary dynamics

Our analysis will be based on the logic of natural selection. Evolution by natural selection requires three components:

1. There must be variation.
2. That variation must have consequences for the survival and/or reproduction of individuals (selection).
3. The variation must be heritable.

Here we will assume that individuals use behavioral strategies related to how they play the prisoner's dilemma game (meaning they have fitness consequences), and that those strategies both vary between individuals and are heritable. I think it's fairly clear that if there are multiple strategies, there is variation. However, it's worth saying a little more about selection and heritability.

Transmission of traits from one organism to another can occur via genes or through *social* transmission – that is, by learning. (There are other ways, including inheritance of environmental constraints on development, but I won't discuss these here). In the case of human cultural traits, we are often talking about social transmission. Luckily, several transmission mechanisms are roughly equivalent in our model. If individuals copy their parents, then the individuals who have the most offspring will propagate their traits *as if* they were genetically determined. This sort of **vertical transmission** of social information can include active teaching as well as less conscious processes. Alternatively, consider the case where individuals actively assess which individuals to learn from. If certain behaviors lead to the accumulation of desirable resources and individuals can assess differences in resources and imitate the strategies of successful individuals, then again, the most successful traits will propagate in the population. This is **success-biased transmission**. Either way, the dynamics of the model work out the same: successful individuals will preferentially transmit their strategies.

We'll use modeling to explore how two or more strategies behave when they interact in a population. In particular, it will be interesting to look at how a strategy behaves both when it is already common in the population, as well as when it is rare. The latter question is particularly important if we are interested in social change: can a strategy spread when only a few individuals employ it. This is the fact of *frequency dependence*: the fitness of a strategy can depend on the frequency of it and other strategies in the population (the rate at which it plays itself vs. other strategies). In our case, this means that the utility of being a cooperator depends on the prevalence of other cooperators. After all, mutual cooperation can outperform mutual defection, but defectors can succeed by exploiting cooperators. A mechanism which makes cooperators more likely to interact with one another could help to make cooperation sustainable. Let's explore this idea with a model.

A simple model with assortment

CODE: PD_simple.nlogo

Just like in the previous unit, we'll use a square lattice structure, in which agents interact with their four nearest neighbors. We'll start by assuming that agents play **pure strategies.** That is, an agent is either a cooperator who always cooperates, or a defector who always defects. Each time step, every agent plays a prisoner's dilemma game with each of their four neighbors, and accumulates a payoff. Then, we have evolution. Every agent will consider their own payoff and the payoff of their neighbors. If any of them have a higher payoff than they do, it adopts the strategy of the neighbor with the highest payoff.

This is an extremely simplistic view of both social behavior, structure, and evolution (be it cultural or genetic). Starting with simple models is a good thing. It gives us a place to start. In general, it's good to remember this about modeling: the baseline model will often be extremely, stupidly simplistic. But simple models can give us insight, including insight into the sort of additional complexity we might or might not need to make sense of our systems, as long as we don't forget that we've made those critical simplifications. And as I hope this course continues to demonstrate, it's not always often how an apparently simple system will behave.

SETTING UP THE MODEL

- *init-coop-freq* slider
- *payoff-benefit slider* (fix at b = 1), since important thing is ration b/c.
- *payoff-cost slider* (between 0 and 1)
- turtles-own [*strategy, payoff*]
    - *strategy* = 0 for defector, 1 for cooperator.
    - We can use integers rather than a Boolean switch to allow for more than two strategies in future versions.

INITIALIZATION

- Each patch of the grid will sprout a turtle. With prob *init-coop-freq*, make a cooperator, otherwise a defector. Cooperators are blue, defectors are red.

DYNAMICS

- Stop if one strategy completely disappears from the population.
- ASK TURTLES:
    - Consider my own strategy and that of my neighbors.
    - Calculate my payoff from all four games
- ASK TURTLES:
    - Compare my payoff to those of my neighbors

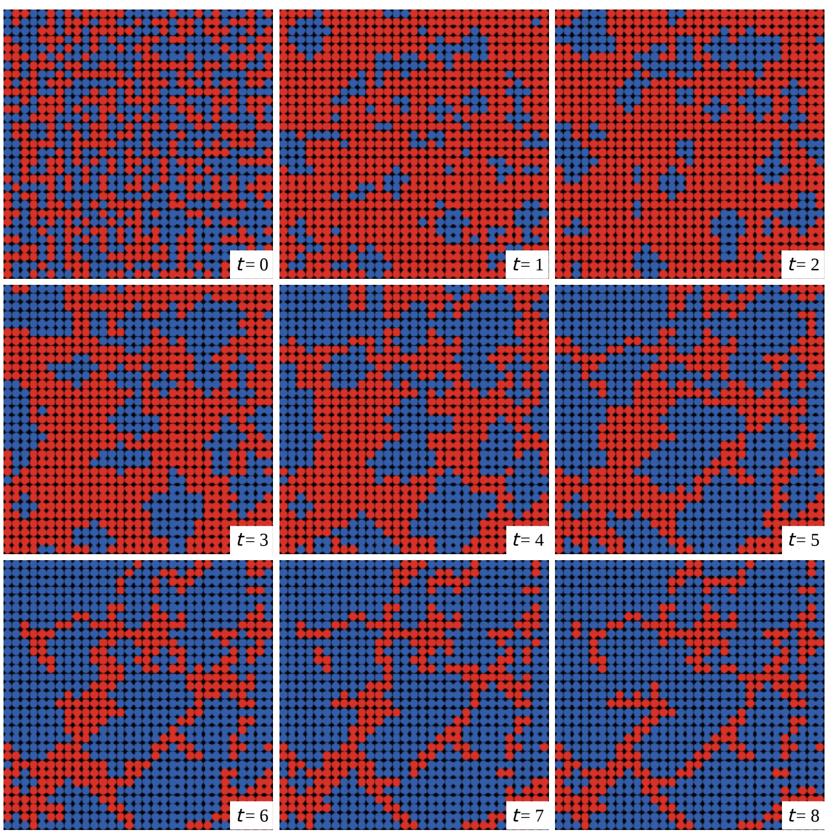
- Copy the strategy of the neighbor with the highest payoff if its higher than mine.

PLOTTING:

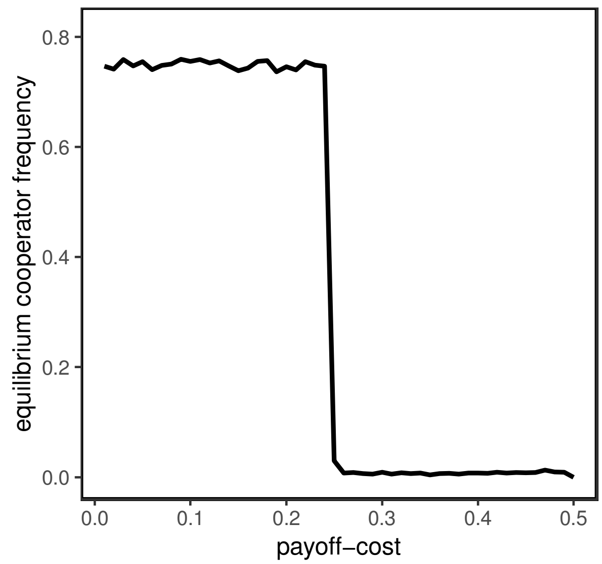
- Agents' colors will represent their strategies, and we can see the change happen on that level. Also plot the frequency of cooperative strategies in the population over time.

RESULTS

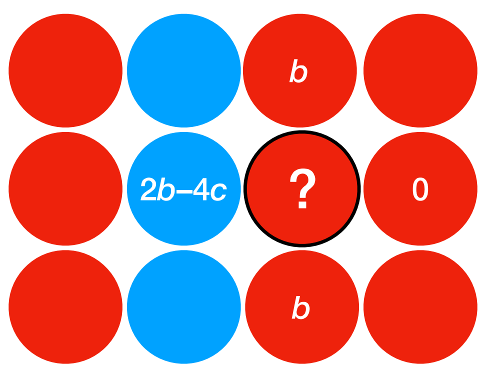
- Let's start with c = .2 (b = 1), and an initial population of 50% cooperators and 50% defectors in random locations. Cooperators dominate, with a few defectors remaining! Perhaps this isn't such a dilemma after all? What's going on?
- Consider the case of *c* = .2 a bit more carefully. Notice in a time course of a single simulation, the frequency of cooperation first goes *down* and then up. Let's look at this more slowly by clicking go-once. Defectors can increase rapidly when cooperators are scattered. A lone cooperator receives no benefits and pays large costs. However, by chance, some cooperators will next to a few other cooperators. And in these cases, the benefits they receive may outweigh the costs they pay. And so places where cooperators assort initially by chance can survive and grow.



- As we start increasing the cost of cooperation, we reach a threshold, above which cooperation collapses. If we start with a mixed population, cooperators increase in frequency if $c < b/4$. If $c \geq b/4$, defectors increase in frequency. To illustrate how stark this result is, I ran ten simulations for 100 time steps each for values of c between .01 and .5 in increments of .01. In this plot, the dots are the individual runs, and the line is their average. We see there's a sudden chance in outcomes right at $c = .25$. Why does this happen?

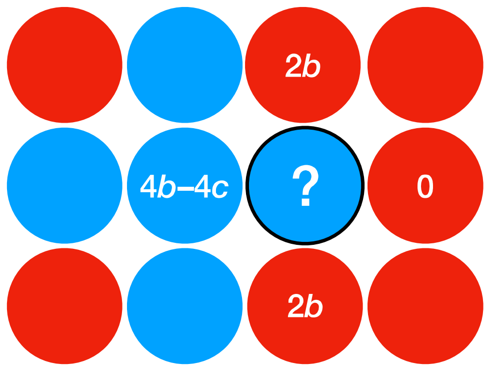


- In this particular model, each agent plays with four neighbors. We already know that one act of mutual cooperation cannot outperform one act of exploitation. What about two acts of mutual cooperation? Consider a cooperator with just two other cooperators as neighbors. Its payoff is $2b - 4c$. A nearby defector who interacts with one cooperator has a payoff of b. Cooperators will therefore spread when $2b - 4c > b$, or when $c < b/4$, and our simulations reflect this.



- When the cost c is above this threshold, cooperation declines, but does not vanish entirely. Some cooperators usually stick around as long as c

< *b*/2, but above that threshold, cooperators go to zero always. Why? In our model, a cooperator can receive at most four acts of mutual cooperation, if all its neighbors are cooperators. When is even this not enough? Consider a single unit of four cooperators. The center cooperator gets a payoff of $4b - 4c$. A nearby defector can interact with two of those cooperators, and get a payoff of $2b$. So, the cooperator formation is stable as long as $4b - 4c > 2b$, or as long as $c < b/2$.



So, cooperation can do quite well if the cost isn't too high relative to the benefit, and there's sufficient assortment. We can see that there are limits to this if we consider invasion of just a few scattered cooperators (say, 5%). In this case, cooperation rarely increases, because there is not enough initial assortment.

In general, our assumption of assortment is very strong. We have assumed that social networks are very rigid and never change. Let's challenge this assumption with a simple extension.

Reducing assortment

CODE: PD_randomized.nlogo

In our first model, agents interacted with neighbors in a fixed network structure. Your neighbors at the start of a simulation are your neighbors forever (or their offspring are your offsprings' neighbors, if we are thinking about genetic evolution – the phenomenon of offspring remaining close to home is called *limited dispersal*). Let's relax that assumption. We'll do this by introducing probabilistic randomization. That is, each time step, every agent has a probability of switching its spatial position with a randomly selected agent, thereby disrupting the spatial assortment that can emerge as we saw before.

SETTING UP THE MODEL

- Add *randomization-prob* slider

INITIALIZATION

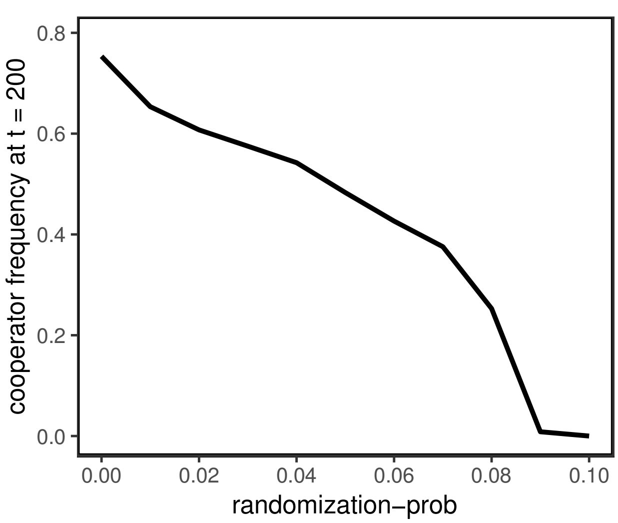
- Each patch of the grid will sprout a turtle. With probability *init-coop-freq*, make a cooperator, otherwise a defector. Cooperators are blue, defectors are red.

DYNAMICS

- Stop if one strategy completely dominates.
- **ASK TURTLES:**
  - **With probability *randomization-prob*, switch spatial location with another randomly selected agent.**
- ASK TURTLES:
  - Consider my own strategy and that of my neighbors.
  - Calculate my payoff from all four games
- ASK TURTLES:
  - Compare my payoff to those of my neighbors
  - Copy the strategy of the neighbor with the highest payoff if its higher than mine.

RESULTS

- Adding random assortment decreases the amount of cooperation that can be maintained in the population, and also makes it more variable over time – it sort of oscillates. Adding just a bit of randomness is enough to drive cooperation to extinction. Here is a plot with 20 simulations for 200 time steps each for values of *randomization-prob* between 0 and .1 in increments of .01 (*c* = .2, *init-coop-freq* = .5). We can see that with only a bit of randomization, cooperation suffers greatly, and past a point can’t be maintained at all.



- This is a problem, because in many real world cases, cooperative partnerships cannot be fully exclusive, especially in the case of the gregarious human. We interact with lots of people, many of whom could easily take advantage of us if we are just naively altruistic, giving our time and resources to anyone we meet.
- To recap, naïve cooperation can invade and be maintained in the population is the costs of cooperation aren't too high and there is very strong assortment across generations. However, these conditions are not likely to be met often, especially for humans who interact with a wide range of others. So next, let's consider a scenario where there is an opportunity for cooperators to be a little more savvy.

Reciprocity

CODE: PD_reciprocity.nlogo

In our previous models, we assumed that cooperative interactions occurred once per time step. This is fine if we are only considering pure strategies, since their performance in one round of game play will be the same as in many. But what if interactions between players don't occur just once, but several times, and we consider *contingent strategies***,** in which individuals have the opportunity to update their behavior based on what happened in previous rounds?

Now our game has shifted to the **iterated prisoner's dilemma (IPD) game**, which just means that pairs of players play multiple rounds of the game. We will introduce a new contingent strategy: Tit-for-tat.

Tit-for-tat (TFT) agents are cooperative but responsive. They don't like being exploited. TFT always starts out cooperating, but thereafter copies its coplayers last move. So it will happily continue to cooperate against another cooperator, but will only cooperate with a defector once (unless the defector switches tactics and starts to cooperate). Here's where the importance of the iterated game comes into play. If we only play once, then TFT does no better than a pure cooperator, since by the time it's ready to respond, the game is over. When the same pairing has multiple opportunities to cooperate, however, TFT is only exploited once, thereafter preferring mutual defection to being played for a sucker. We will need to update our model to account for multiple rounds of game play, but this is easily accomplished. Let's see how TFT fares.

CODING THE MODEL

- Add switch *TFT?* to convert our pure cooperators into TFT agents

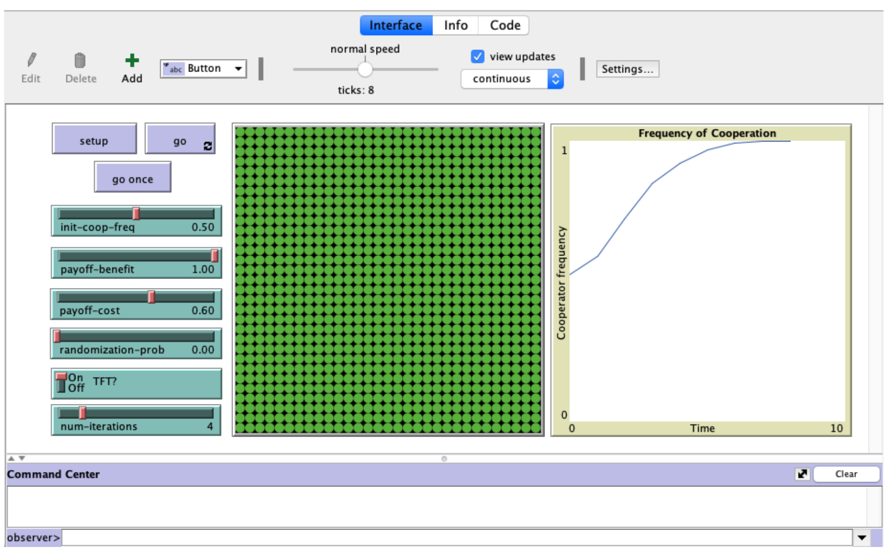
- Add *num-iterations* slider, controlling for how many rounds each game is played.

INITIALIZATION

- Each patch of the grid will sprout a turtle. With probability *init-coop-freq*, make a cooperator, otherwise a defector. Pure cooperators are blue, **TFT agents are green**, defectors are red.

DYNAMICS

- Stop if one strategy completely dominates.
- ASK TURTLES:
  - With probability randomization-prob, switch spatial location with another randomly selected agent.
- ASK TURTLES:
  - Consider my own strategy and that of my neighbors.
  - Calculate my payoff from all four games
  - **Account for number of iterations when TFT agents are involved**
- ASK TURTLES:
  - Compare my payoff to those of my neighbors
  - Copy the strategy of the neighbor with the highest payoff if its higher than mine.



RESULTS

- First, we note that if the number of iterations is one, the model is exactly equivalent to the one with pure strategies, since TFT doesn't get the opportunity to respond to its opponent.
- Now, let's increase the number of iterations. Let's say four. Start with spatial assortment, and c = .25. This is where pure cooperators got into trouble. Yet TFT not only outperforms defectors, it totally

dominates. Now let's crank up the cost, all the way to c = .6. Remember, at this point, pure cooperation is toast. Not TFT! It dominates again.

- But so far we've assumed very strong assortment, so cooperative clusters can persist. For illustration purposes, let's turn the randomization-prob all the way up to 1, so there is no longer ANY persistent spatial assortment across generations. No problem! TFT still wins, because when it interacts with cooperators it can take advantage of the situation, but still avoids getting exploited too badly by defectors. TFT can handle much higher amounts of randomization, because it can capitalize when it gets paired with cooperators and pays only minimally when it gets paired with a defector.
- If we continue to increase the costs, we do get to a point where TFT does better with less randomization – there are still benefits to cooperation. Consider *c* = .7. TFT persists in a mixed equilibrium for num-iterations = 4 and no randomization, but loses out as the randomization is turned up a bit.
- TFT can also invade with less initial assortment than pure cooperators, because it doesn't get exploited. Let's consider the case where num-iterations = 4, c = .3, init-coop-freq = 0.05, and randomization-prob = 0.1. Under these conditions, ALLC would get creamed. However, even though TFT agents rarely find each other, as long as they do occasionally, the benefits of their repeated interactions still outweigh the costs of being exploited occasionally.
- In general, though, TFT is a lot more robust than ALLC, and can permit the persistence of cooperation under greater costs and lower assortment, as long as interactions persist long enough so that there is ample opportunity for reciprocal cooperation. It's OK to occasionally be exploited if you can subsequently ignore exploiters but continue to interact with other savvy cooperators.

Further directions

- Diving deeper. There is a vast, vast amount of work modeling cooperation, much of it focusing on solving the prisoner's dilemma. There is also a lot of empirical work, both in humans and non-human animals. It's a very rich literature. The research on cooperation and assortment that the first part of this lesson is based on originally dates to W. D. Hamilton's work in the 1960s (Hamilton 1964), who discussed genetic relatedness as an important mechanism for generating assortment. The work on reciprocity stems from two seminal papers, one by Trivers (1971) and the other by Axelrod and Hamilton (1981). A lot of research on cooperation explores various mechanisms for assortment, which also includes work on group structure, network structure, movement strategies, environmental harshness, and reputational management (cooperating with those who have a reputation for cooperating, sometimes called indirect reciprocity).

 - Trivers RL (1971) The evolution of reciprocal altruism
 - Axelrod R (1997) *The complexity of cooperation: Agent-based models of competition and collaboration*. Princeton University Press.
 - Nowak MA, Sigmund K (2005) Evolution of indirect reciprocity. *Nature* 437: 1291–1298.
 - Smaldino PE, Schank JC, McElreath R (2013) Increased costs of cooperation help cooperators in the long run. *American Naturalist 1*81: 451–463.

- <u>Cooperation in larger groups</u>. Our model focused on cooperation between just two individuals. However, cooperation sometimes involves larger groups. A related game, called the public goods game, is a sort of *N*-person prisoner's dilemma game for arbitrarily large groups. When cooperation involves groups of more than two people, it turns out that reciprocity doesn't work that well as a strategy, partly because defectors can free ride on the efforts of the majority. This may be one of the reason that altruistic behavior is rarely observed in large, unrelated groups in non-human animals. To get cooperation in large groups, something more is needed. Analyses of this game have considered the importance of psychological and institutional mechanisms like conformity, reputation, punishment, and policing.
  - Simon HA (1990) A mechanism for social selection and successful altruism. *Science* 250: 1665–1668.
  - Boyd R, Richerson PJ (1992) Punishment allows the evolution of cooperation (or anything else) in sizable groups. *Ethology and Sociobiology* 13: 171–195.
  - Henrich J, Boyd R (2001) Why people punish defectors: Weak conformist transmission can stabilize costly enforcement of norms in cooperative dilemmas. *Journal of Theoretical Biology* 208: 79-89.
  - Hooper PL, Kaplan HS, Boone JL (2010) A theory of leadership in human cooperative groups. *Journal of Theoretical Biology* 265: 633–646.
  - Smaldino PE, Lubell M (2014) Institutions and cooperation in an ecology of games. *Artificial Life* 20: 207–221.
- <u>Cooperation and competition</u>. There are many other issues relevant to cooperation other than simply avoiding or punishing free riders. Group boundaries and markers can help identify individuals identify certain individuals to cooperate with or avoid, for better or worse. Cooperation is also part and parcel with intergroup conflict, and conflict may actually drive cooperation, as more cooperative groups will be better able to compete. Because cooperation is such a core part of human

societies, there are many important avenues for research, including many untapped or barely explored ones.

- Hammond RA, Axelrod R (2006) The evolution of ethnocentrism. *Journal of Conflict Resolution* 50: 926–936.
- Choi J-K, Bowles S (2007) The coevolution of parochial altruism and war. *Science* 318: 636.
- Makowsky MD, Smaldino PE (2016) The evolution of power and the divergence of cooperative norms. *Journal of Economic Behavior & Organization* 126: 75–88.
- Waring TM, Goff SH, Smaldino PE (2017) The coevolution of economic institutions and sustainable consumption via cultural group selection. *Ecological Economics* 131: 524–532.

Exercises

- *There's a million games they haven't run.* Using simulations, can you establish a relationship between *payoff-cost* and *num-interations* for the evolution of TFT cooperators? Assume that TFT agents are initially 10% of the population, and there is no network assortment (randomization-prob = 1). What is your general conclusion? (If you want a hint, check out the mathematical proofs in Axelrod and Hamilton 1981, which explore this question).
- *Making mistakes.* Sometimes sometime might intend to cooperate, but nevertheless fail to do so. For example, you might intend to drive your friend to the airport, but you mark the wrong date on your calendar and oversleep. Introduce implementation error into the model. Add a variable called *error-prob* that affects TFT agents: with some probability, any given act of cooperation can change to a defection. Recall that TFT starts out cooperating, and thereafter copies its co-player's previous move. This will involve a careful consideration of the code. Explore how error disrupts the success of TFT considerably.