VICTORIA UNIVERSITY OF WELLINGTON
Te Whare Wananga o te Upoko o te Ika a Maui

Computer Science

PO Box 600
Wellington
New Zealand

Tel: +64 4 463 5341
Fax: +64 4 463 5045
Internet: office@mcs.vuw.ac.nz

Reciprocal Altruism for Dummies

Emanuel Rabina

Supervisor: Dr Marcus Frean

Submitted in partial fulfilment of the requirements for:
Bachelor of Information Technology

Abstract

The biological world is filled with examples of creatures helping each other out. However, in a world where the benefits of rational self-interest far outweigh the benefits of cooperation, it's a big mystery as to why animals continue to help each other out. One possible explanation is the notion of reciprocal altruism, where a creature helps another out in hopes of the other creature paying back the favour in the future. Because of the requirements of this interaction, the standard view is that it only occurs in animals with cognitive abilities.

This project seeks to dispel this view by investigating a form of reciprocal altruism in which the creatures involved have no cognitive abilities. By developing a simulation focussed on the symbiosis between a squid and its light-emitting bacteria, we show that organisms with none of the cognitive abilities normally associated with altruistic creatures are nonetheless able to exhibit a form of reciprocal altruism.

Acknowledgements

First and foremost, I'd like to thank my supervisor, Marcus Frean, for introducing me to this strange and wonderful hybrid of biology, game theory, and computers (hmm, it's all just science isn't it!). Cheers for the patience he has shown, and the assistance he has provided, as I desperately tried to grasp the integration of all these new ideas.

Next, all the people that regularly inhabit the level 4 BIT/Comp labs; only they could ever create an environment where I can feel both good and bad about myself, and make me keep coming back for more.

Gratitude aside, I would like to say that this project has been a huge learning experience. It was even kind of fun: the biological aspect got me interested, the game theory got me curious, and the programming of a simulation got me energised.

What more could you ask for in a project? (cookies I guess… mmm, cookies)

- Emanuel Rabina

Contents

List of Figures/Graphs

Figures Page No.

Graphs Page No.

1. Background

1.1. Introduction

The living world is filled with inter & intra-species relationships of cooperation, exploitation, and everything in-between. Associated with cooperation is the notion of altruism: a trait where the well-being of others takes precedence over the well-being of oneself. A creature with altruistic behaviour will then go out of its own way to do something useful for another. Such behaviour has an associated cost with it, and being cooperative 100% of the time has very little benefits for the creature in question.

However, if the other creature cooperates back, then both can benefit from this mutual cooperation (normally called mutualism in biology). This is called reciprocal altruism:

> *Reciprocal altruism is a form of altruism in which one organism provides a benefit to another in the expectation of future reciprocation (Wikipedia 2005).*

Reciprocal altruism is one of several theories used to explain the maintenance of cooperation.

An example of reciprocal altruism can be found amongst vampire bats. Vampire bats need blood every few days to survive. If they can't get blood, they'll approach another bat whilst roosting, asking for a blood 'transfusion'. The blood is exchanged mouth-to-mouth in a motion that looks very much like kissing. It is hoped that after doing this, the bat receiving the blood will do the same for another bat in future. There is however, the chance that this bat won't reciprocate, and this further complicates such relationships with the advent of 'cheaters' (Okasha 2003).

Now suppose both parties make their cooperation contingent on the cooperation of the other party. If the other party cooperates, then so will it. If they don't, then it won't either. This eliminates the option of 'free-riding' if any party is hoping to receive the greater benefits of mutual cooperation. This behaviour is another form of reciprocal altruism, loosely called 'tit-for-tat' (TFT).

In general, this type of cooperation requires good recognition and memory (for the prior identification of previous reciprocal efforts), and the standard view is that it can only occur between creatures with advanced cognitive abilities (Dugatkin 1997, Stevens & Cushman 2004).

However, recognition and memory may not be strict requirements. It may be sufficient for an agent to generate an environment in which the other is directly rewarded for cooperating shortly after it does so. In such a place, playing TFT has not only the best benefits but the only way to maintain such an environment.

This project will explore such an alternative.

We examine a particular situation in which reciprocal altruism would appear to fail – the symbiosis between the Hawaiian bobtail squid and its light emitting bacteria. The light produced by the bacteria is of benefit to the squid when it looks for food, and the living conditions within the squid appear ideal for the bacteria.

In this scenario, because of the cognitive requirements of reciprocal altruism, you wouldn't think that a single-celled organism would know how to persist in this relationship of mutualism, let alone millions of them. Yet they do, and probably have been for what could be millions of years.

So we believe that, despite popular belief, a form of reciprocal altruism can occur amongst these basic organisms (Frean & Abraham 2004).

1.2. Biology

The 2 players in our simulation are the Hawaiian bobtail squid, *Euprymna Scolopes*, and light-emitting bacteria, *Vibrio Fischeri*. This section introduces them and some of their more interesting aspects.

1.2.1. Squid – Euprymna scolopes

E. scolopes is a small bobtail squid endemic to the Hawaiian Islands. Most aspects of this squid are actually very small: the average length for a full grown adult is 3.5cm, with a life span of a mere 3-10 months. *E. scolopes* are also found in very shallow waters of 2-4cm deep. Like most other cephalopods, *E. scolopes* is a nocturnal creature, spending its days buried in the sand. When it comes out at night, it likes to eat small shrimp (Wood 1999).

Being a night-time feeder, as well as living in the shallow water ranges, *E. scolopes* faces the challenge of alerting both predators and prey. When the moon is shining, the squid casts an easily visible shadow that can alert other creatures to its presence. To counter this, it makes use of the light created by light-emitting bacteria that live in a specialised light organ of this squid. Through the use of a light-diffusing behaviour known as counterillumination, the squid can disguise itself in its own moonlight glow (MBL 2005).

Figure 1a: The tiny Hawaiian bobtail squid, *Euprymna Scolopes*. Makes for a good pet too for you cuttlefish fans.

The light organ, located in the mantle of the squid, is home to several of these light-emitting bacteria. The squid isn't born with these bacteria, but rather harvests them from the surrounding seawater after hatching (Frean 2004, McFall-Ngai 1999).

Due to the relative ease at which *E. scolopes* can be raised in captivity, as well as the symbiotic relationship it shares with its bacteria, the *scolopes-fischeri* relationship is the only experimental model available to biologists (McFall-Ngai 2005).

The squid in its entirety is not modelled in the simulation. Instead, only the light organ, the environment within the squid in which the bacteria live, is modelled.

1.2.2. Bacteria – Vibrio fischeri

Vibrio fischeri is a rod-shaped bacterium found globally in marine environments, albeit in very sparse numbers. The only time you'd ever find several of these bacteria in the same place is in symbiosis with another marine creature (taking advantage of its light-emitting properties for camouflage, finding food, or both), or in a lab culture for the purpose of some experiment.

Free-living *V. fischeri* have flagella to propel themselves within the water, but undergo several morphological changes, including losing the flagellum, when in symbiosis with another creature.

V. fischeri don't live within the squid forever, but rather are recycled daily. The squid expels a large portion of it's *V. fischeri* population every morning (roughly 95%), either as a means of population control, or to aid the colonisation of the light organ of juvenile squid (Sachs et al 2004, McFall-Ngai 1999).

To be of any use to our squid, the bacteria would need to produce levels of light at least equal to that of the moon on a clear night. However, in free-living conditions, *V. fischeri* actually produce very little light at all. In the light organ of the squid though, bacteria are reported to emit 1000 times more light compared to when they're cultured outside of the squid (MBL 2005).

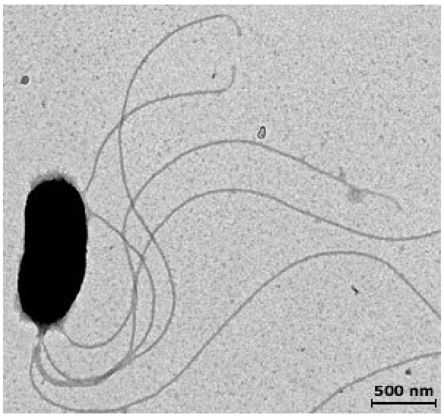


Figure 1b: *Vibrio fischeri* still with its flagellum under an electron microscope.

The bacteria play the most visible role in the simulation. Several hundred bacteria can be present on the simulation area at any given time, and while they won't be given any real brains (ie: an advanced cognitive suite and recognition abilities), they have been given the ability to evolve their various parameters as normal bacteria do.

1.2.3. The Light Organ

Conditions in the light organ of the squid are considered to be ideal for the bacteria, conditions that can hardly be improved upon in lab cultures (Sachs et al 2004). It's still a mystery to scientists for the real reasons why the bacteria produce light for the squid, but one of the theories as to why this occurs is that the bacteria do it as a means of self-preservation, as follows.

The light organ is an environment high in oxygen. This oxygen is needed for 'host respiratory burst activity' (an immune response to potential pathogens). When required, a series of chemical processes occur that lead to the creation of hypochloric acid (HOCl).

Hypochloric acid is a type of hypohalous acid: a strong disinfectant and bacteria killer that is easily able to pass through the cell membrane of micro-organisms, oxidise the cell contents, and kill them (Healthy Buildings 2002).

One way the bacteria can defend themselves against this immune response is to remove the oxygen from the surrounding environment. The light reaction undertaken by the bacteria requires and consumes oxygen. So by producing light, the bacteria can starve the chemical processes within the local area of this much needed ingredient for the bacteria-killing hypochloric acid.

For the squid, the times it needs the light of the bacteria the most is at night when the shadow it casts by the moonlight can give away its position. Observations have shown this to be the case: that light levels are highest at night when the squid is out and about. However, bacteria outside of the light organ produce a constant amount of light both at day and night times. Because of these facts, the most conservative explanation for this fluctuation in light levels is that the squid controls the levels of oxygen in the light organ (McFall-Ngai 1999).

This is one theory for the production of light, and because the interactions and the maintenance of cooperation that occur between squid and bacteria are not yet completely known, it is this theory that this project has been based on.

1.3. Game Theory

1.3.1. Prisoner’s Dilemma

Stemming from game theory, the Prisoner’s Dilemma describes a 2x2 ‘payoff matrix’. That is, a matrix describing the size of the payoffs (consequences) to each player based on their own actions and the actions of the other player. An example of the Prisoner’s Dilemma can be described with the following example (involving prisoners no less), adapted from (Kuhn 2003):

> *Ethan and Casey have just been arrested for robbing a bank and have been placed in separate interrogation rooms. Both of them care much more for their own personal well-being and freedom than their accomplice.*
>
> *The detective makes the following offer to each prisoner: “You may choose to confess or stay silent. If you confess and your accomplice remains silent, I will drop all charges against you and use your testimony to ensure that your accomplice does serious time. Likewise, if your accomplice confesses while you remain silent, they will go free and you will do the time. If you both confess, I get two convictions, but I’ll ensure that you get early parole. If you both stay silent, I’ll have to settle for token sentences on possession of a firearm charges.*

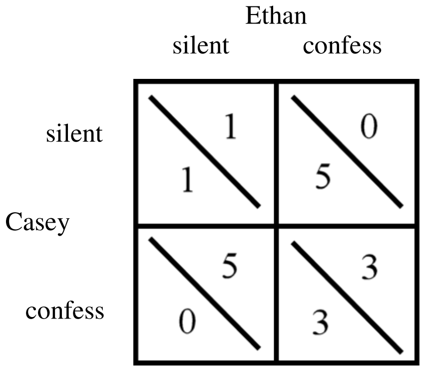


Figure 1c: The translation into prison years for each of the prisoners in our example. The upper portion of the \ is Ethan's jail term, while the lower portion of the \ is Casey's jail term.

If we translate these sentences into (relative) jail terms, we get the matrix shown in figure 1c on the left.

The dilemma faced here by both prisoners is that whatever the action of the other, each of them is better-off confessing. Yet the jail term for both confessing is much worse than that if they both chose to remain silent.

More generally, a group where the members are of an 'every one for themselves' mindset would end-up much worse than a group where the members were altruistic. Alternatively, a group whose members pursue a goal with only rational self-interest will meet less success than a group whose members worked together (Kuhn 2003). The game can also be played between members of different groups, and this is where inter-species cooperation fits in.

So what does this all have to do with biology and our project? Prison sentences don't mean much to animals, and I don't think they know what it means to confess either. The idea behind this story (one of many forms of making the original Prisoner's Dilemma more accessible) is this conflict between individual and group rationality: selfishness and selflessness, especially in the presence of selfish incentives.

So abstracting away from jail-terms and confessions, if we return to the original form of the Prisoner's Dilemma, what we get is 2 parties, each given the ability to perform 1 of 2 actions: cooperate with the other party, or defect: mapping to 'remain silent' and 'confess' from our prison example. The payoffs also map directly: each is better-off defecting no matter the action of the other party, but the payoff for cooperation from both parties is much greater than the payoff for both defecting. This general case is shown in figure 1d.

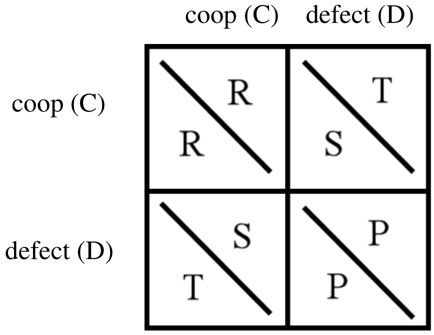


Figure 1d: Instead of numbers, we have these letters representing the different consequences based upon the available actions.

R is the **r**eward received by both parties should they cooperate, while P is the **p**unishment for both defecting. T is the **t**emptation received for defecting alone, where S is the **s**ucker payoff received for cooperating alone. Now we have a payoff matrix satisfying this chain of inequalities:

$$T > R > P > S \quad \text{and} \quad 2R > T + S$$

The first ensures defection is favoured over cooperation, while the second gives it that paradoxical quality: the combined total payoff is highest for mutual cooperation (Frean 1996).

As with the prison example, defection is the best option no matter the actions of the other player. If they cooperate, then you get the temptation award, and if they defect, then you're safe from being the sucker. Defection truly holds its own in a one-shot situation. However, if a species exhibited this defection behaviour in the long run, it would not be expected to survive the natural selection process for very long. In other words, a strategy of complete defection is not evolutionary stable.

Imagine that when shopping, if a shop decides to rip you off over and over, then you'll remember them hatefully, and will stay away from that shop in future. And if the shop continues to play the defection card with all customers all the time, then they won't be in business very long. When it comes to several iterations of the prisoner's dilemma game, memory and cognition start to play a vital role as past interactions serve as a factor to your next decision with the other player.

Ignoring the whole 'bargaining power' aspect of the previous example (economics, not our field), what we are now dealing with is the evolution of strategies of cooperation. The prisoner's dilemma has been applied in many evolutionary scenarios, and ever since it came to be, the strategies that came forward to explain the evolution of cooperation have been based around an iterative prisoner's dilemma game. One of the best known of these is called 'Tit for tat'.

1.3.2. The Evolution of Cooperation

Tit for Tat (TFT) is a strategy that emerged from 2 computer tournaments held by university professor Robert Axelrod in 1981. Axelrod invited contestants to submit their strategies to see how they'd pan-out in a computer simulation. It was a round robin competition in which the various strategies were paired against another, and over several iterations (a median of 200) several qualities were recorded, such as the total payoff populations using a strategy would receive, as well as population size, to decide a winner.

The winner of both of these tournaments was TFT, submitted by an Anatol Rapoport (Axelrod 1984, Maynard Smith 1982, Meredith 1998).

An agent using the TFT strategy will initially cooperate, and then respond to the other player's previous action.

Against all the other strategies, including against those where the other strategy attempted to cheat/defect in a variety of cunning ways, TFT always came out on top. By using the simple strategy mentioned above, it is able to harness the reward payoff against other populations that choose cooperation (by reciprocating cooperation), and is able to resist becoming the sucker against defectors (by reciprocating defection). And against itself, it reaps the reward payoff due to the initial cooperation move. Also, it has proved to be evolutionary stable, and able to infect populations of other strategies with its own (Maynard Smith 1982).

After the second competition, Axelrod identified several features of TFT in an effort to explain its success. In short, Axelrod stated that TFT is a successful evolutionary stable strategy because it is 'nice', 'provokable' and 'forgiving'. A nice strategy is one where it is never the first to defect. A provokable strategy is one where it responds to defection by immediately defecting against it. And a forgiving strategy is one where it will readily return to cooperation only if its opponent does the same.

Because of the inherent similarities between reciprocal altruism and TFT, this project will prove TFT amongst the bacteria, which in turn will show reciprocal altruism.

2. The Simulation

Most of the work involved in this project went into developing a plausible simulation of the squid and bacteria described in section 2. The simulation covers the conditions within the light organ of the squid and the interactions therein.

The programming language of choice was Java, simply because of my familiarity with it.

2.1. Diffusion

Because of the bacteria's ability to remove the local amount of oxygen around them, I needed first to create a surface area where such a reaction could take place. With the light organ also being a viscous environment, the removal of local oxygen would leave a sort of 'dip' that would be filled-in by the surrounding oxygen and the natural diffusion process.

At its heart, the simulation is a large 2D surface area for watching and simulating diffusion. The first steps I took when creating the simulation into what it is today, was to actually get a model of diffusion working properly, one where large amounts of the representative values used to measure the diffusive process wouldn't be lost after some time.

Start:

0	0	10	0	0

1st Step:

0	3.33	3.33	3.33	0

End:

2	2	2	2	2

Figure 2a: Given the starting point in the top picture, and enough time to continue, the result should be the bottom picture.

This started off first by getting a large number (which could represent any arbitrary measurable substance, although at the initial stages of the project it didn't really matter what it was) to spread itself evenly from 1 point anywhere in a 1D array into all the available spots. That is, if the array had 5 slots and one slot had a value of 10 in it, given enough time, all the slots would end-up with the value of 2 (figure 2a).

The formula for calculating the value at a given slot at any point in time in 1D is:

$$v_i = v_{i-1} - ((v_{i-1} / (n + 1)) * n) + (n * \alpha)$$

Where:
- v is the value
- i (or the subscript of v) is the current iteration at which this formula is run
- n is the number of neighbouring slots
- α is the average value received from all of this slots' neighbours

The breakdown of this formula is as follows:

$$v_i = v_{i-1} - ((v_{i-1} / (n + 1)) * n) + \ldots$$

The current value is calculated by taking the previous value, and then subtracting the amount sent to the 2 neighbouring cells. A single amount is the previous value divided by the number of neighbours plus itself. For example, a cell with a value of 9 would send to each neighbour 9 / 3 = 3. This leaves 3 for itself, thus equalising the value to itself and its neighbours. This is then multiplied by the number of neighbours.

$$v_i = \ldots + (n * \alpha)$$

Then, the values received from every neighbour are added to what is left. Because the value received from each neighbour could potentially be different, I have just used the above to represent the average value received from all neighbours.

Edge slots are dealt with by having an invisible neighbour which just throws back the value that it received from the edge slot. This way no values are lost at edges, and edges don't act as strong borders to the simulation.

In a 2D space, diffusion is slightly more complicated as there are now 2 types of neighbouring cell: immediate neighbours and corner neighbours. The immediate neighbours were those directly above, below, left and right of the cell in question. Corner neighbours are the other 4 that are connected at the corners.

Accordingly, diffusion to the corners should not occur at the same rate as it does to immediate neighbours because they are √2 further away. Using an equal rate would lead to anisotropic diffusion (eg: rigid diamond shapes) rather than smooth circles. Therefore, values sent to corner neighbours are only √2 of the normal amount.

Start:

0	0	0
0	10	0
0	0	0

1st Step:

0.71	1.11	0.71
1.11	2.72	1.11
0.71	1.11	0.71

End:

1.11	1.11	1.11
1.11	1.11	1.11
1.11	1.11	1.11

Figure 2b: 2D diffusion in the intended smooth circular fashion.

The formula for calculating the value at a given slot in 2D then becomes:

$$v_i = v_{i-1} - ((v_{i-1} / (n_{total} + 1)) * n_{imm}) - (1/\sqrt{2} * (v_{i-1} / (n_{total} + 1) * n_{cor})) + (n_{total} * \alpha)$$

Where the algebra is the same as before except for:
- n_{total} is the total number of neighbours
- n_{imm} is the number of immediate neighbours
- n_{cor} is the number of corner neighbours

The breakdown of this formula is as follows:

$$v_i = v_{i-1} - ((v_{i-1} / (n_{total} + 1)) * n_{imm}) - \ldots$$

This part is the same as the 1D formula and deals exclusively with immediate neighbours. Instead of 2 neighbours, there are now 4 immediate neighbours.

$$v_i = \ldots - (1/\sqrt{2} * (v_{i-1} / (n_{total} + 1) * n_{cor})) + \ldots$$

This part is new to the formula and deals exclusively with corner neighbours. Notice the introduction of the 1/√2 * part which reduces the amount to send to corner neighbours by √2 accordingly.

$$v_i = \ldots + (n_{total} * \alpha)$$

This part is the same as the 1D formula, once again summarising the unknown values received from all neighbours as 'the average of all neighbours'.

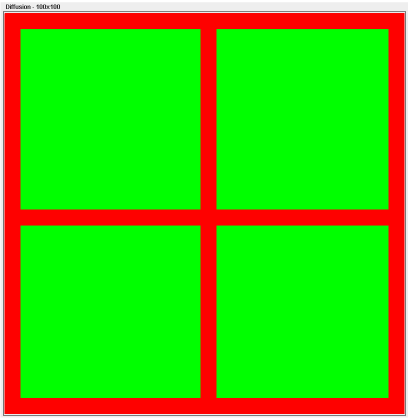


Figure 2c: An initial starting position used to demonstrate the diffusion (the actual simulation just starts-off all green).

Now that the diffusion was working, the value was chosen to represent the levels of the bacteria-toxic hypochloric acid in the area.

A visual representation of the diffusion process was generated with a large section of the simulation's GUI dedicated to the light organ's surface area of 100x100 cells. The screenshot to the right and on the next page show diffusion occurring in the completed simulation.

To the right here in figure 2c we have an image of the area of the overall simulation GUI that shows diffusion taking place (later on this area becomes home to the bacteria). Normally the area just starts-off all green to represent a lack of toxin, but for this demonstration I have forced some areas to begin with high levels of toxin (red). Depending on the level present in each cell, the colour will vary between red and green.

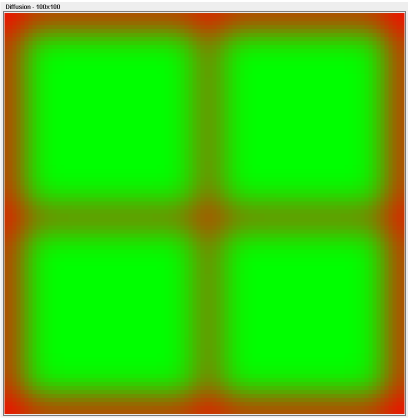


Figure 2d: After several more iterations of the diffusion algorithm, there is what we get.

The overall area is 100x100 cells, and each cell is a Java class of its own that extends the Button classes from the in-built Java libraries. While it could have been any sort of GUI component, I chose a Button so that I could attach click behaviour to the cells later on.

Each cell implements the 2D diffusion algorithm from the previous page which is run on each cell before applying the updated toxin value (synchronous update).

After several iterations, we get something like figure 2d where the green and red areas are starting to blend with one another.

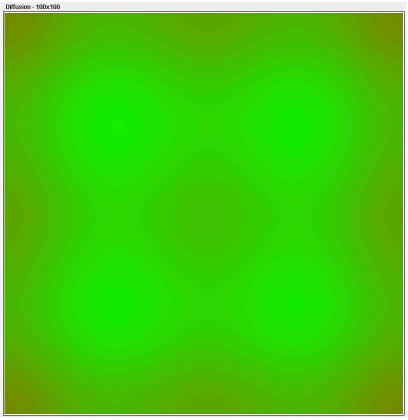


Figure 2e: Getting the diffusion to this stage is akin to watching the paint dry.

And after letting it run for a lot longer, we finally get something like that shown in figure 2e. The areas of high toxicity are all but gone, and the colour across the entire simulation area is becoming uniform.

2.2. Squid

While the squid itself is not visually represented in the simulation, the variables that affect the levels of toxin can be thought of as being under the squid's control. While these variables remain static throughout the life of the simulation, I make several runs altering the values to these variables.

An example of one is the speed at which the diffusion occurs within the 100x100 area. Changes to this variable could be similar to the 'viscosity' (for lack of a better word) in the light organ of an actual squid, and there are some results where I attempt to alter this value (results, section 5).

So in the simulation, the squid can do the following:
- leave the light organ alone
- flood the light organ with toxin

In the context of the prisoner's dilemma game, these actions correspond to cooperation and defection.

Because the light from the bacteria is needed by the squid at night time, the simulation will alternate between periods of not producing toxins (day) and producing toxins to stir-up some light activity (night).

Another thing to consider with a day and night cycle is the flushing of bacteria from the light organ of the squid. When first starting to get bacteria to breed in the simulation, overpopulation did become a problem. The action of flushing bacteria from the light organ has been proposed as a method of population control (section 2), and the simulation does have this capability.

This did have the tendency to slightly disrupt long-term results, so was dropped in favour of implementing a carrying capacity instead. This carrying capacity doesn't just stop the seeding or breeding of bacteria once it hits a certain point. Instead, as numbers increase, the chances of adding breeding just become reduced.

2.3. Bacteria

Built atop the diffusion model is where the light-emitting bacteria reside. As mentioned in sections 1 and 2, the bacteria emit light to reduce the oxygen available that leads to the production of hypochloric acid. This of course means less hypochloric acid. So in the simulation, the bacteria will be reducing the toxin levels on the cell of the diffusion model they reside in (and because of diffusion, this will reduce the toxin around them as well).

While the bacteria I described in section 2 have many interesting attributes, most of them were left-out both to keep the simulation simple and to maintain focus on the original goals.

The bacteria can do the following:
- emit light
- replicate (binary fission)

In the context of the prisoner's dilemma game, these correspond to cooperation and defection.

The bacteria attributes are:

- Health
- Regeneration
- Switch level
- Light detox
- Breeding/Fission rate

Health of the bacteria gets reduced if the local toxin exceeds some preset 'tolerance' level. If health reaches 0, then the bacteria dies and is removed from the simulation. Regeneration is a characteristic used to keep bacteria in the simulation for longer. It is inherently a part of the health equation as follows:

$$h_i = h_{i-1} - \gamma + r$$

Where:

- h is the health of the bacteria
- i (or the subscript of h) is the current iteration at which this formula is run
- γ is the level of toxin in the cell in which the bacteria resides
- r is the level of health to regenerate

The switch level is the attribute that sets what level of exposure to toxin will start the bacteria emitting light. When the levels of toxin in the local cell equal or exceed this value, then the bacteria turns on its light to reduce the toxin.

Light detox is the amount of local toxin that will be reduced when emitting light. When emitting light, the squares around the bacteria are made brighter in the simulation to visualize it. Light detox is dependent on the switch level, so operates by the following:

$$\textit{if } (\gamma > s) \textit{ then [light on] } \gamma_i = y_{i-1} - d$$

Where:

- s is the switch level
- i (or the subscript of γ) is the iteration at which this formula is run
- d is the light detox level

Fission rate (actually the interval between the next binary fission) is how often the bacteria perform binary fission. While influenced by health and the carrying capacity, any new bacteria that get passed these measures are created adjacent to the existing bacteria with the exact same set of attributes as the original.

$$\textit{if } (f \textit{ modular } i = 0) \textit{ then [perform binary fission]}$$

Where:

- f is the interval between attempts at fission (lower = replicate more often)
- i is the current iteration at which this formula is run

At first, the last 3 attributes were randomly generated upon the creation of the bacteria, and then those values remained constant throughout their lifetime. Bacteria were also assigned a colour in the simulation:
- Red = defectors (high switch level, low light detox, shorter fission interval)
- Blue = co-operators (low switch level, high light detox, longer fission interval)
- Black = in-between

Finally, the last thing to do with the bacteria was to make the bacteria attributes of switch level, light detox, and fission rate, evolvable.

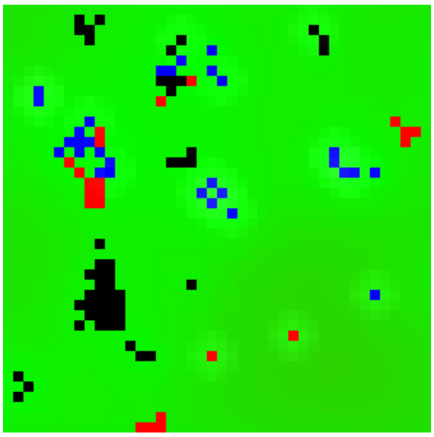


Figure 2f: Visual representation of the bacteria in the simulation. As you can see, some have established mini-colonies through binary fission, and some are emitting light.

2.4. Evolution

Evolution amongst the bacteria was done very simply: depending upon the average 'feeling' of several previous iterations since the last evolution, alter the parameters accordingly. What I mean by 'feeling' is whether the case was more often or not one where the level of toxin in the cell that the bacteria resides was above or below the bacteria's switch level.

If it was more often that the level was below the switch level, then that means the toxin levels are too low for the bacteria to bother creating light. The bacteria will instead adjust its parameters slightly towards defection: lower light detox level, increase switch level, shorten fission interval.

If it was more often that the level was above the switch level, then that means the toxin levels are getting the better of the bacteria, causing it to create light quite often. The bacteria will then adjust its parameters slightly towards cooperation to remove the toxins: higher light detox level, decrease switch level, lengthen fission interval.

Over time, the diversity of the bacteria would decrease, but move towards something more suited for the environment. That result, and all the other combinations of attributes along the way, will tell us more about the squid-bacteria relationship.

2.5. Other Modelling Decisions & Assumptions

Other things about the operation of the simulation that didn't fit into the previous subsections:

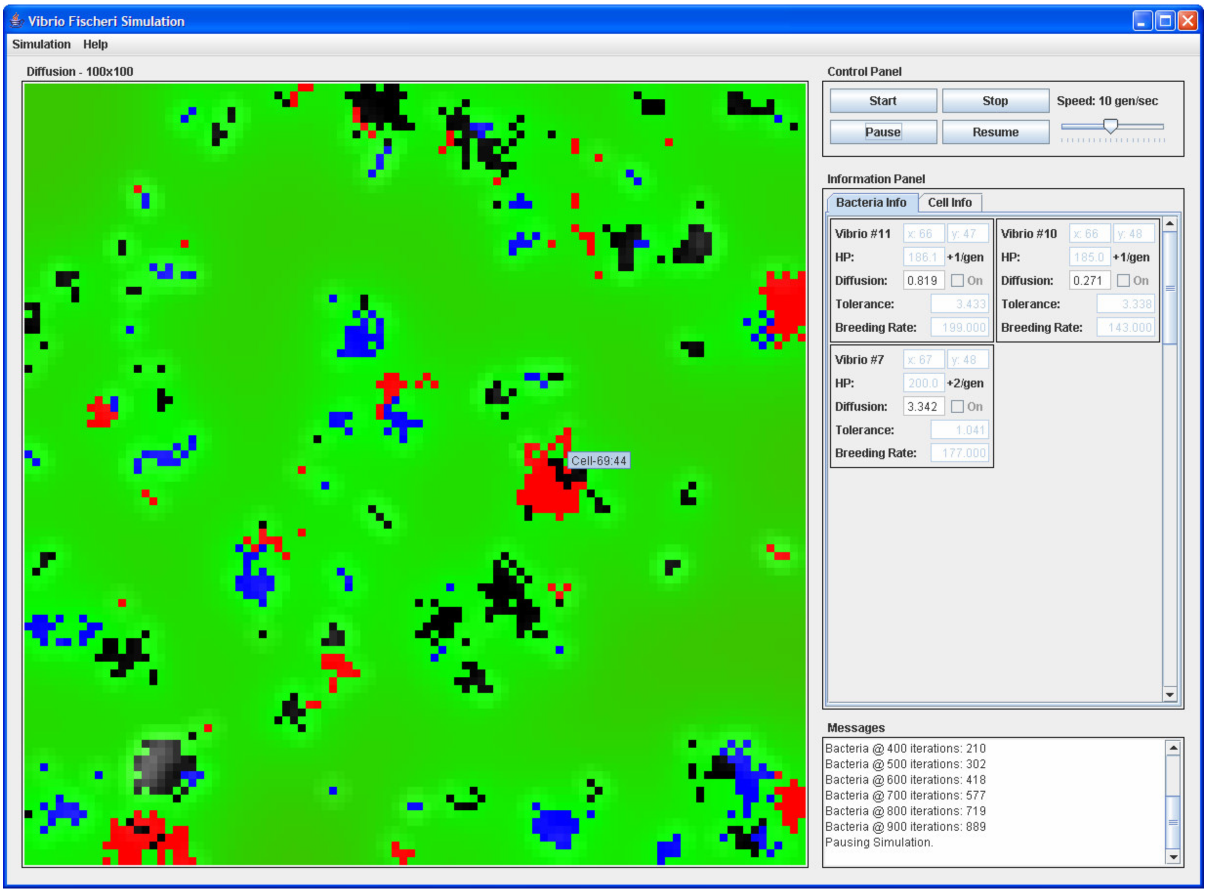
- When starting, the simulation area will be randomly seeded with a small amount of bacteria. This was originally just to give the simulation a kick-start, but can be considered to be the juvenile bacteria when they hatch, harvesting bacteria from the surrounding seawater.

- Throughout the running of the simulation, an even smaller amount of bacteria will be added to the simulation over set intervals. This function is affected by the carrying capacity, so the continual running of this function will not overload the population.

- The way in which the toxin is introduced to the simulation is by increasing the amount in each cell at set intervals during the defined night time period. Initially it was seeping-in from the edges by setting the edge cells to the maximum toxin levels and locking them at those levels, where the diffusion process would then bring the toxin to the centre. Observations have shown that the current method of seeping-in from the floor of the simulation is a lot fairer as the central areas aren't a favoured spot for bacteria.

2.6. Extras

Apart from modelling the interactions between the squid and bacteria, the simulation also incorporated several extra features, mainly for my benefit of tracking and debugging. The following screenshot is the simulation in its entirety, and the caption explains these extras.



Vibrio #11 x: 66 y: 47
HP: 186.1 +1/gen
Diffusion: 0.819 ☐ On
Tolerance: 3.433
Breeding Rate: 199.000

Vibrio #10 x: 66 y: 48
HP: 185.0 +1/gen
Diffusion: 0.271 ☐ On
Tolerance: 3.338
Breeding Rate: 143.000

Vibrio #7 x: 67 y: 48
HP: 200.0 +2/gen
Diffusion: 3.342 ☐ On
Tolerance: 1.041
Breeding Rate: 177.000

```
Bacteria @ 400 iterations: 210
Bacteria @ 500 iterations: 302
Bacteria @ 600 iterations: 418
Bacteria @ 700 iterations: 577
Bacteria @ 800 iterations: 719
Bacteria @ 900 iterations: 889
Pausing Simulation.
```

Figure 2g: The simulation in its entirety. On the right is where most of the extra features reside. The top-right has controls for starting/stopping/pausing/resuming the simulation, and the slider controls the speed at which the simulation runs using its own metric called 'generations'. Intervals between actions are based on these generations, as is the rate at which the diffusion algorithm is applied. Centre-right is a tabbed display panel showing extra information for cells/bacteria that live in them. When on the bacteria panel, it displays all of the 5 attributes of bacteria. When on the cells panel, the location and local toxin value is shown. Finally, the bottom-right is a message output area, mainly denoting how many generations have run, and how many bacteria are present in the simulation.

3. Experiments and Results

3.1. The Simulation as Normal

Here is the simulation left running using the standard parameters that closely mimic the day-to-day interaction between the squid and its bacteria.

When performing a test run of the simulation, the following attributes are recorded:
- breeding/fission intervals (or breeding/fission rate)
- light detox levels
- switch levels
- number of bacteria in the simulation
- percentage of those bacteria emitting light

In the simulation, 1 night/day cycle is 1000 iterations: 500 day, 500 night. Each test run consists of 5000 iterations, 5 days.

The results are as follows:

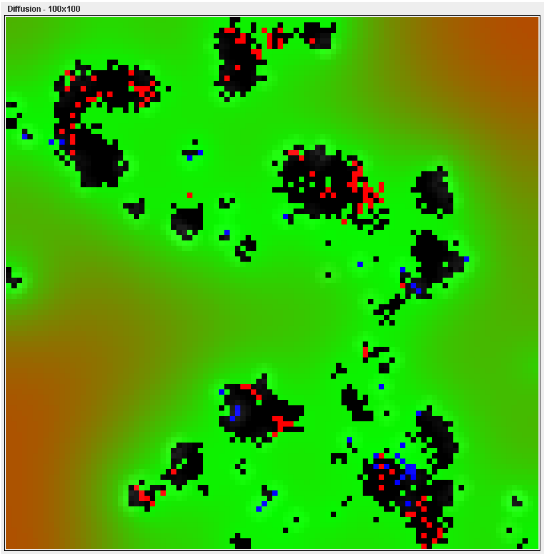


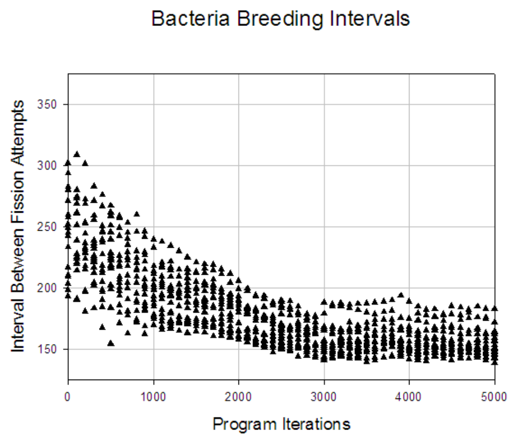
Figure 3a: The simulation after a normal run - the bacteria are mostly of the in-between sort.

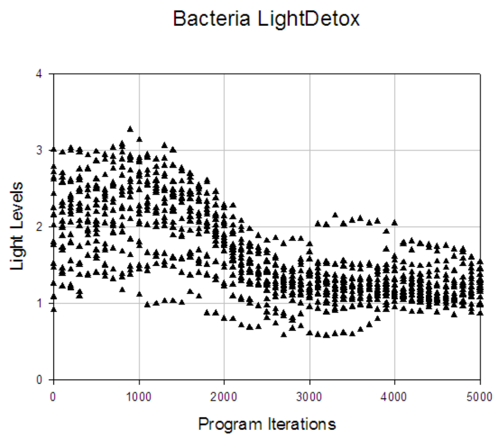
The picture to the left shows mostly in-between bacteria: neither fully cooperating of fully defecting. Upon closer inspection, it shows slightly more defectors than co-operators. The graphs on the next page show that the evolution of the bacteria has headed mainly in the direction closer to defection, but with a fluctuating population and total light emission, giving the squid the benefits it seeks from the bacteria at night.

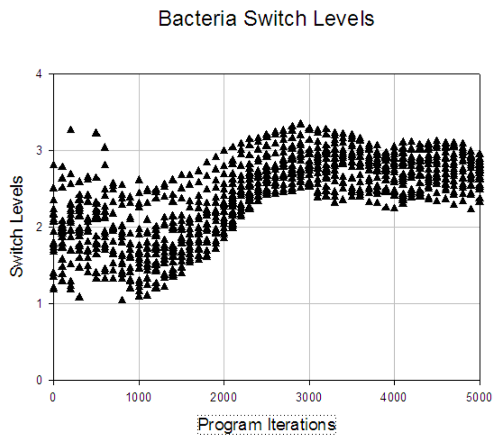
From these, we could at least deduce that the bacteria both cooperates and defects, but as mentioned before, is just a reactionary action to the toxin.

Are the bacteria playing TFT though? The picture and the graphs we have at the moment won't tell us the whole picture, so a few more runs must be made, particularly ones where we can directly compare the range from a squid's cooperation (no toxin) to defection (generate toxin), compared to the bacteria's range from cooperation (high light, slower breeding) to defection (low light, faster breeding).

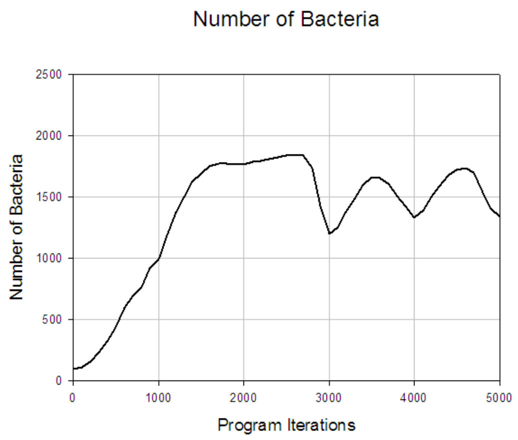
Before answering the question of whether the bacteria play TFT, I first take a look at each quadrant of the prisoner's dilemma payoff matrix, to see if the benefits received by both bacteria and squid properly correspond to the tested quadrant.

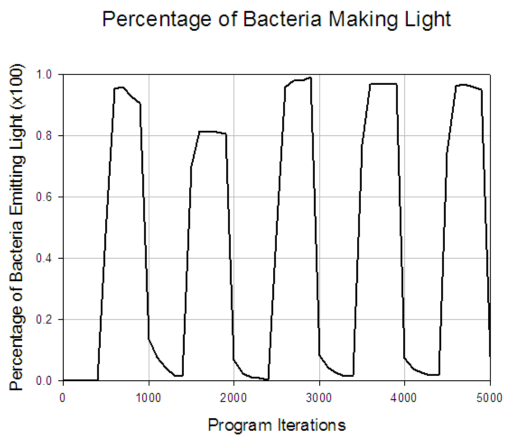






Graphs 3a-e: The normal run doesn't tell us too much about the bacteria. The first 3 tell a story of mainly defection, but not to the same level as we'll see in future total-defection runs. The number of bacteria emitting light, as well as the population of bacteria, coincides with the day-night cycle of the squid.





3.2 A Prisoner's Dilemma

Ensuring that what takes place in the simulation was a prisoner's dilemma as described in section 3, turns were taken between the squid and bacteria, setting their strategies to be one of only cooperation or only defection.

3.2.1 Cooperating Bacteria, Defecting Squid

A cooperating bacteria is one that always has its light on and evolves towards cooperation (meaning more light, less breeding). A defecting squid means that it will not provide the wonderful environment that the light organ normally is, and instead generate toxins all the time. One worry which I have is that the constant generation of toxins will prevent any bacteria from gaining a foothold in the light organ. Hopefully their tendency to evolve for cooperation (which means more light and increased defence against toxins) will give them a fighting chance.

The results are as follows:

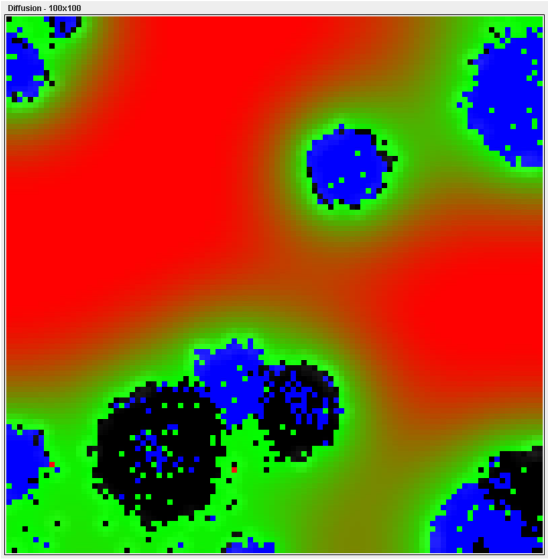
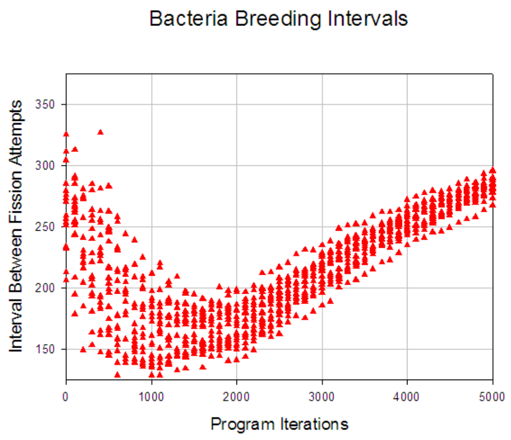


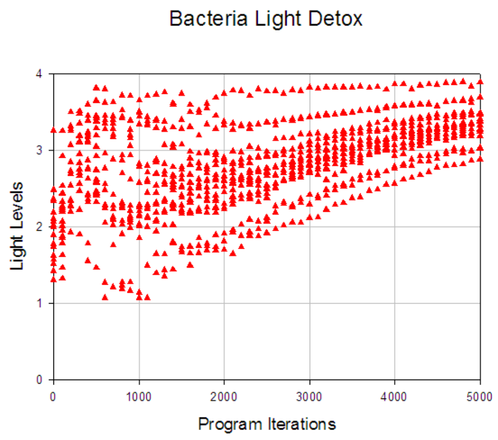
Figure 3b: The final picture after the simulation was run. Luckily, the bacteria that opted for cooperation were able to survive the onslaught of constant toxin generation. Even those that didn't (the few red dots and the large colonies of black dots) were able to survive because they lived next to areas of low toxin reduced by nearby co-operators.

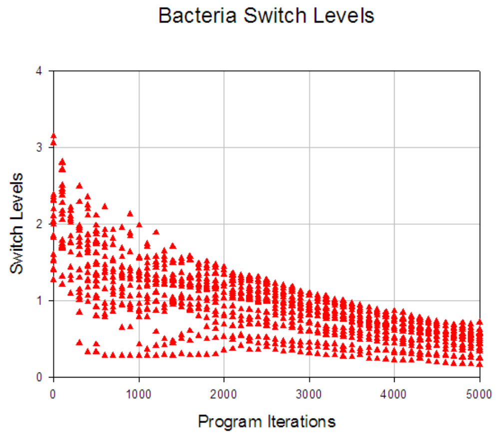
Looking at the picture to the left and the graphs on the next page, we can see the majority of bacteria are cooperators. Because of the reduction in breeding, it took the bacteria just over 2000 iterations to reach the carrying capacity of the simulation. There is also this visible flat line of the population at the beginning: this was where the bacteria where struggling to gain a foothold in the form of a colony.

It's also easy to see the squid getting the better of the bacteria. Almost from the beginning, 100% of the bacteria are producing light at all times. And at the high level of light the bacteria have evolved to produce, the squid could probably even emulate 2 moonlight glows!

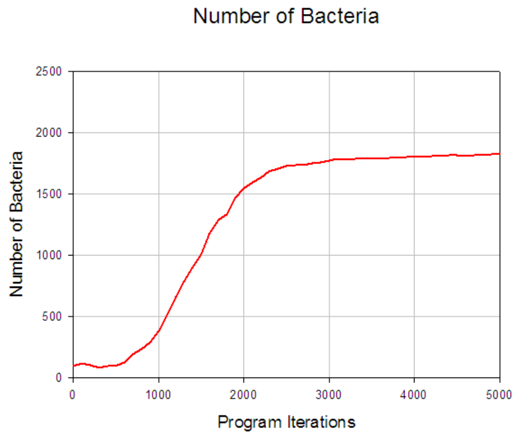
The bacteria gains very little benefits from this interaction, and is being exploited by the squid, just like in the appropriate quadrant of the prisoner's dilemma.

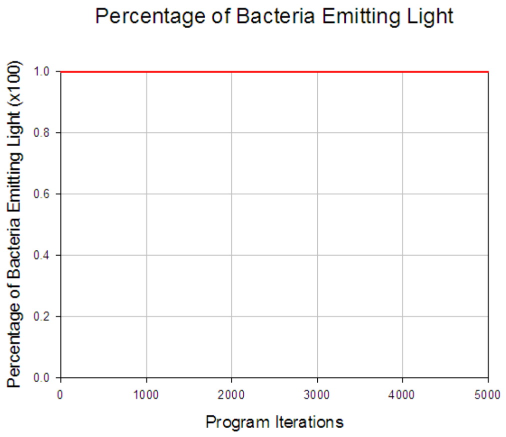






Graphs 3f-j: Results from the simulation run where squid make a sucker out of the bacteria. From the first 3 graphs we can see the bacteria evolving its attributes to concentrate on light production (longer interval between fission attempts, increased light levels, and decreased switch levels), even though the switch level should be ignored because a light 'always on' scheme was used.





3.2.2. Defecting Bacteria, Cooperating Squid

This time, the squid will be the sucker to a defecting bacteria population, revealing what goes on in the opposite quadrant of the prisoner's dilemma to the one before.

If the squid's end of the bargain in the standard symbiosis is to provide an ideal living space for the bacteria, then a forever cooperating squid is one who will do this all the time: no toxins are ever generated in the light organ. The evolution of the bacteria will be one to favour defection to take advantage of this situation.

What I expect to happen this time around is for the squid to gain virtually nil benefits while the bacteria live the good life.

The results are as follows:

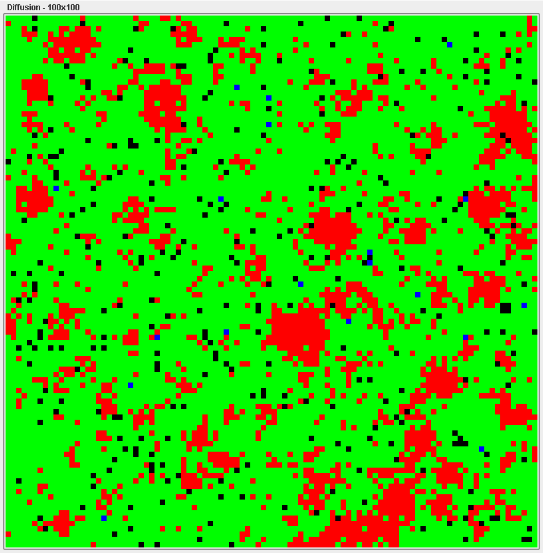


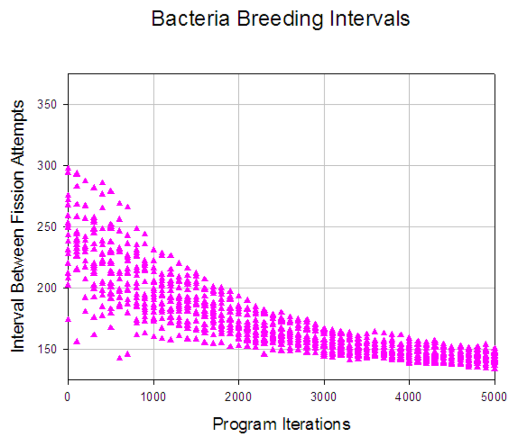
Figure 3c: The majority of bacteria in the light organ are defectors taking advantage of the squid's 'hospitality'.

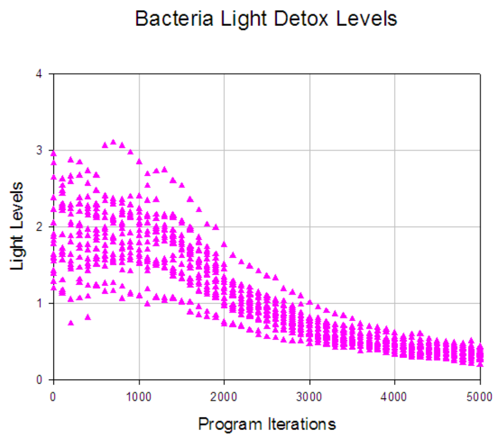
Now we see a sprawl of bacteria like miners to a gold rush. With no toxin to regulate bacteria, they have the ability to live anywhere, and have done just that. The previous run (coop bacteria, defect squid) saw bacteria only able to survive by sticking close to one another.

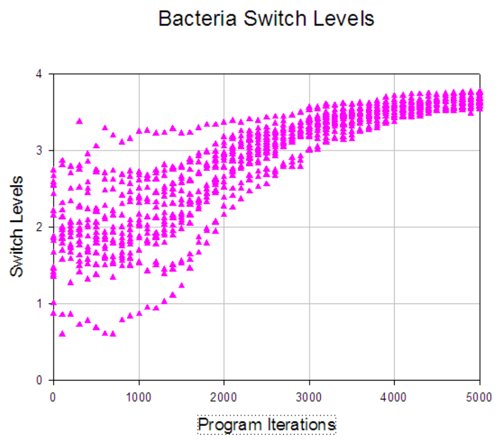
Looking at the graphed results on the next page, we can see the speed at which the bacteria achieved carrying capacity, just over 1000 iterations (1 day). The previous run took over 2000 iterations and had a moment at the beginning where bacteria had to achieve a foothold by establishing a colony before any growth could begin. This time, no 'foothold lag' is present.

Looking at the final graph of % of bacteria actually emitting light, it remains 0 throughout the run. With no motivation to produce light and a strategy aimed for defection, this is no surprise.

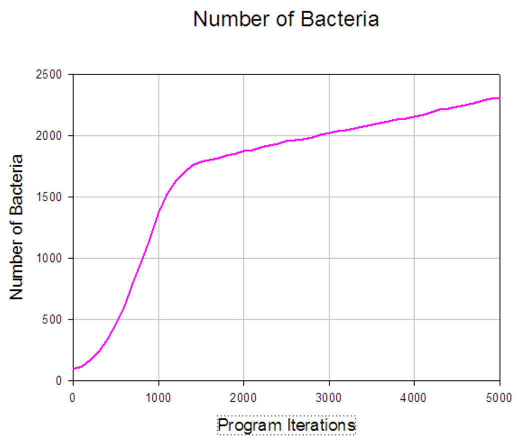
The squid gains no benefits from this interaction.

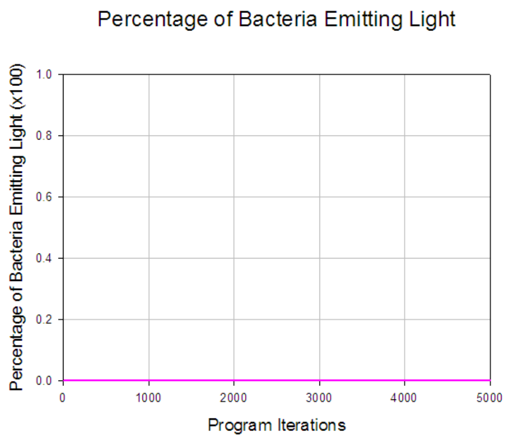






Graphs 3k-o: Results from the simulation run where bacteria make a sucker out of the squid. From the first 3 graphs we can see the bacteria evolving its attributes to concentrate on breeding. Dropping its light production isn't a huge consequence as the squid is not making any toxin to fight against.





3.2.3. Defecting Bacteria, Defecting Squid

Here, both squid and bacteria take the role of defection, and so receive some benefits against the actions of the other, thus exploring the defect/defect quadrant of the prisoner's dilemma.

In terms of benefits, it will be minimal at best. If the bacteria keep their switch/tolerance levels high and the squid just keeps generating toxins, then I expect to see very little bacteria on the simulation at any given time. The constant toxin would eventually just make the environment too potent to live in, killing any new bacteria very quickly. There may be some light, but only a small reaction as the bacteria make a desperate attempt to save themselves.

The results are as follows:

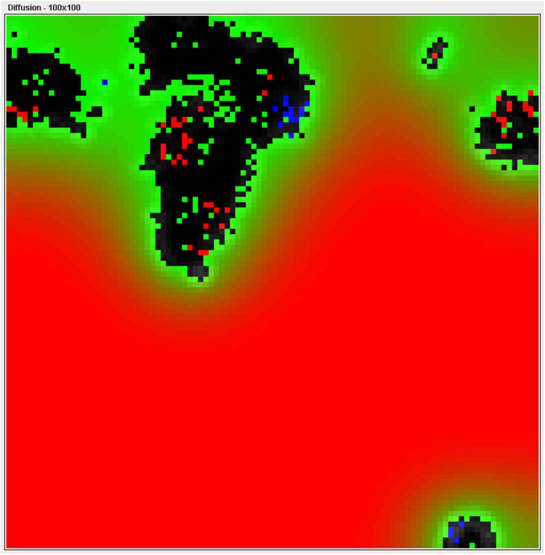


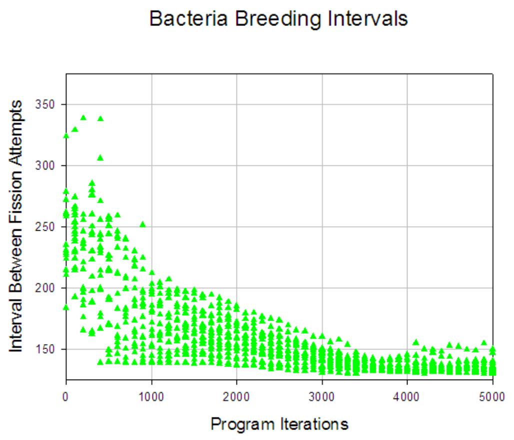
Figure 3d: The squid is obviously defecting, but the population consists of more normal bacteria than defectors. Where are the defectors?

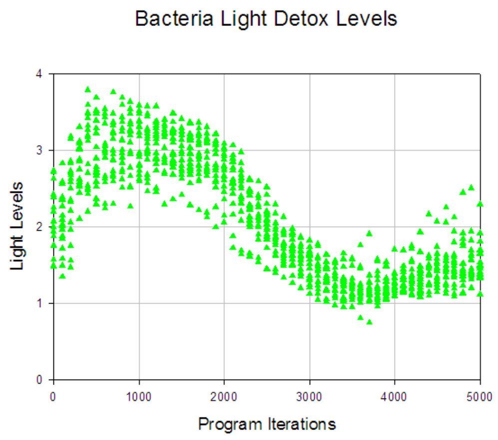
Now things are starting to get interesting. From the picture on the left, bacteria with a defection strategy should be all over the place displaying their defecting red colour prominently. But all we get is the red of extremely high toxin levels. Where are all the defectors?

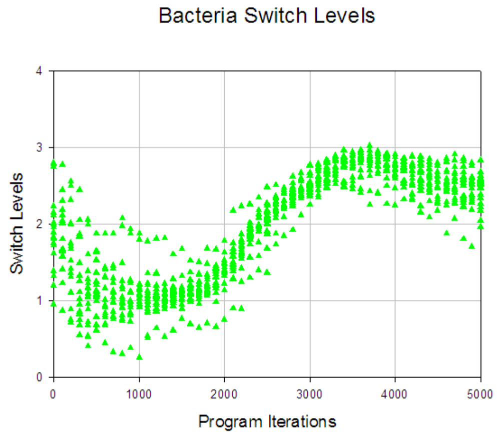
As I watched this run unfold, what I witnessed was the evolution of defectors, followed by the swift removal of them by being drowned in toxin levels that they were unable to defend against. A strategy of defection entails low light production, thus low defence. The squid's toxin has weeded-out defectors, reinforcing the notion I made earlier that always-defect isn't evolutionary stable (not for the bacteria in our simulation anyway).

A look at the graphs on the next page starts to show several stories: the breeding intervals of bacteria have moved towards the end depicting faster attempts at binary fission, whereas the light and switch levels make very late strides towards defection. The explanation for this would be that in an all-tox environment, cooperating bacteria survive best and longest. Because of the removal of defectors (as described in the previous paragraph), the sample points gather data from a majority of the cooperating survivors. Once evolution kicks in, then the graphs start to follow.

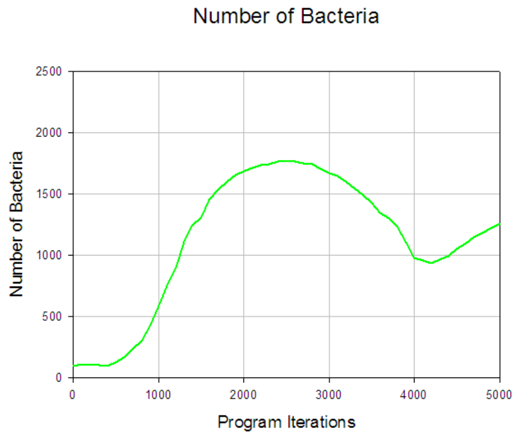
The number of bacteria emitting light is still very high (80%+), but 80% of the less-than-capacity population (and a fluctuating population at that) isn't too much.

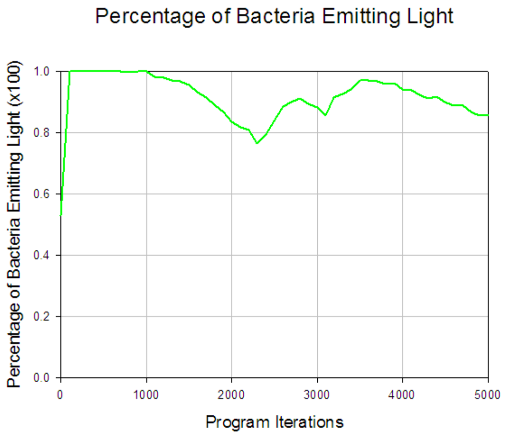

Bacteria Breeding Intervals


Bacteria Light Detox Levels


Bacteria Switch Levels

Graphs 3p-t: A strange picture is being painted here, with the graphs of light levels, switch levels, and population showing no definitive direction.


Number of Bacteria


Percentage of Bacteria Emitting Light

3.2.4. Cooperating Bacteria, Cooperating Squid

Lastly, both squid and bacteria take the role of cooperation in an attempt to receive maximum total benefits, thus exploring the coop/coop quadrant of the prisoner's dilemma.

The ideal mutual cooperation scenario would be one where the bacteria emit light with very little motivation at all. The squid will then not generate any toxins to provide an ideal environment for the bacteria 24/7. This strong focus on emitting light should put the bacteria's fission rate very low, so less binary fission will occur.

Once again, because of the 'always on' light of the bacteria, the switch level graph has been made redundant.

The results are as follows:

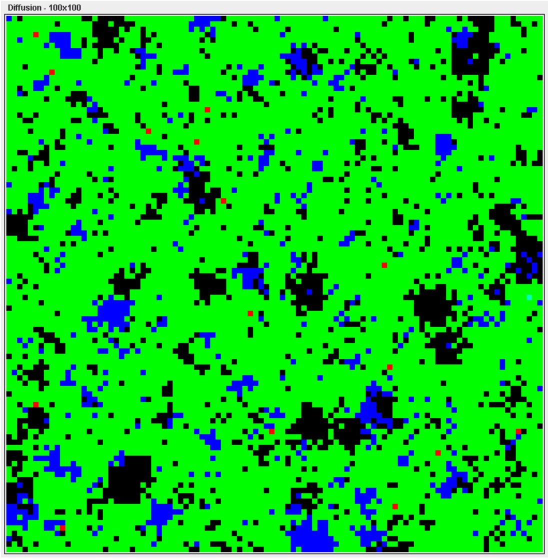
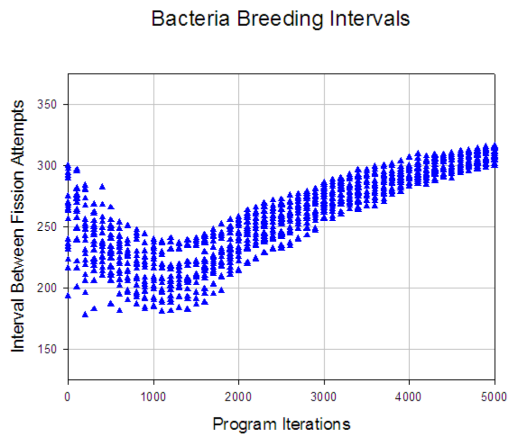


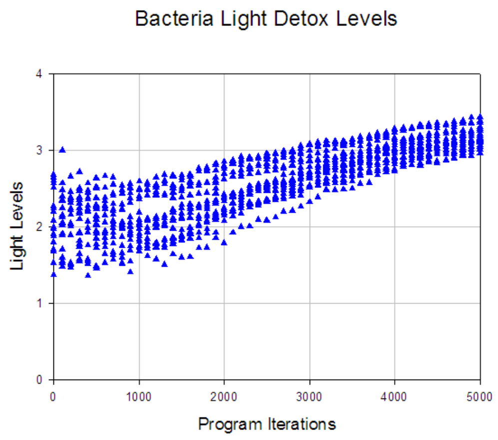
Figure 3e: Another bacteria sprawl, except this time the occupants are co-operators.

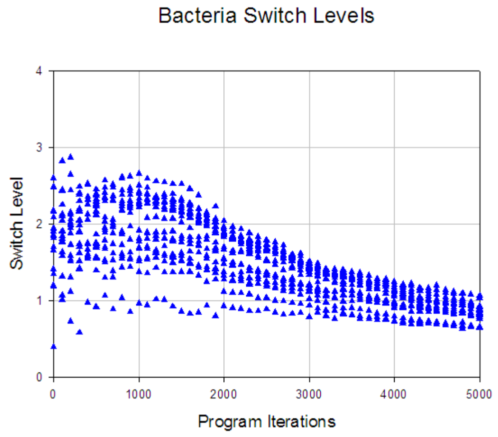
Looking at both the picture on the left and the graphs on the next page, we see relatively high benefits for both parties.

For the bacteria, even with their reduced focus on breeding, we still see a quick increase in the bacteria population, almost as fast as what we've seen when the bacteria opted for a defect strategy.

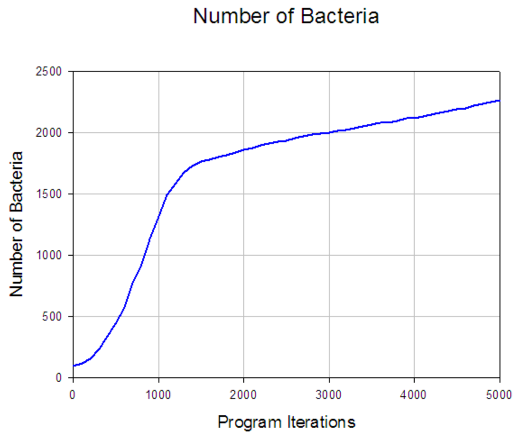
For the squid, because of the always on nature of the bacteria's light, and the bacteria's evolution towards more light production, then coupled with the fact that the population isn't stunted with toxins, and the squid is getting all the light it could ever want (maybe enough for 3 moonlights?).

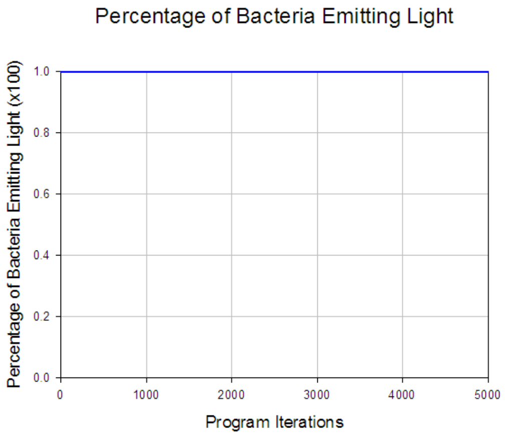






Graphs 3u-y: The first 3 graphs show what is to be expected from a cooperating squid (even though the switch level graph is once again redundant for this situation). The last 2 show a steady growth of bacteria only seen in the defect bacteria / coop squid, while the 100% light is close to what we've seen with the coop bacteria / defect squid situation.





3.2.5 A Prisoner's Dilemma - Summary

Just to recap what has been presented, the following payoff matrix summarises the benefits each party received in each of the interactions, just to ensure that it is a prisoner's dilemma game. Remember, the prisoner's dilemma is about payoffs for a certain 'duet' of actions, satisfying this chain of inequalities:

$$T > R > P > S \quad \textit{and} \quad 2R > T + S$$

For every benefit, 1 will be added to that creature's 'score' for that interaction. For every cost, 1 will be deducted. And for something not perceived to be either, nothing will be added or subtracted.

Figure 3 on the next page shows the scores as I interpreted them from the graphs and figures on the previous pages, and the table below explains how these numbers were obtained.

Bac coop / Squid coop, R / R:
- B-1 bacteria making light at all times
- B+1 mediocre population growth
- B+1 bacteria can live anywhere
- S+1 obtaining light at all times…
- S+1 …from a large population of bacteria
- S-1 maintenance of the light organ

Bac defect / Squid coop, T / S:
- B+1 bacteria not making light
- B+1 fast population growth
- B+1 bacteria can live anywhere
- S-1 no light at all…
- S+0 …from a large population of bacteria
- S-1 maintenance of the light organ

Bac coop / Squid defect, S / T:
- B-1 bacteria making light at all times
- B-1 slow population growth
- B+0 bacteria confined to living in colonies
- S+1 obtaining light at all times…
- S+1 …from a large population of bacteria
- S+1 no maintenance of the light organ

Bac defect / Squid defect, P / P:
- B-1 bacteria forced to make light
- B+1 mediocre population growth
- B+0 bacteria confined to living in colonies
- S+1 obtaining light at most times…
- S-1 …from a fluctuating/low population of bacteria
- S+1 no maintenance of the light organ

Assigning values to the variables, T = 3, R = 1, P = 0, S = -2, satisfying the chain equation above.

Surprisingly, the values for bacteria/squid temptation/sucker seem to be identical. This was mere coincidence of my part of the adding, but does challenge a view I developed where the bacteria seem to be getting the most out of each of the interactions. Even when they are receiving sucker payoffs, the bacteria still (after a long time) manage to reach the carrying capacity of the simulation.

Another thing to consider is what was perceived to be the costs and benefits in the table above was only what was most visible from the pictures and graphs. Coupled with a whole-number scoring system as opposed to a fraction based one, and this could account for both the 'same T/S value' coincidence, as well as the discrepancy in the tallying of the benefits/costs from the defect/defect interaction.

		Squid	
		coop (C)	defect (D)
Bacteria	coop (C)	1 1	3 -2
	defect (D)	-2 3	0 0

Figure 3f: Payoff matrix for the squid and bacteria in each of the prisoner's dilemma test runs.

When comparing it to the coop/coop quadrant; the 'obtaining light at most times' benefit definitely isn't as rewarding as the 'obtaining light at all times' benefit, especially when coupled with the fluctuating/low bacteria population.

Alternatively, a run of 5000 generations may not have been enough for that interaction.

4. The Emergence of Tit-for-Tat

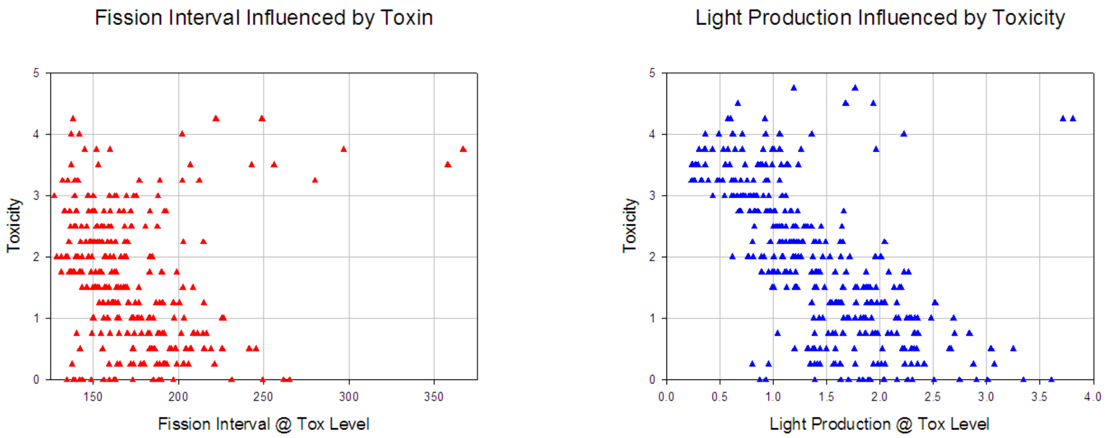
So now that it can be shown that the prisoner's dilemma can describe the sorts of payoffs that each party is receiving for their actions, it's still to be shown that some form of tit-for-tat is going on here.

Trying to track exactly which quadrant of the prisoner's dilemma is occurring at each phase can be quite difficult as there are several bacteria on the simulation at any given time. What we can look at instead is what happens to each bacterium under the influence of the squid. What I mean by this is whether or not the cause for bacteria favouring the emission of light versus a higher breeding rate is a direct result of the level of toxicity the squid can generate. Is the basic payoff between light and breeding rate being controlled by the toxins generated by the squid?

To find the answer to this question, I will do a normal simulation run, this time gathering light levels and breeding rates at the varying levels of toxicity that occur in the simulation. While the obvious answer is yes – these attributes are of course altered by toxins - the less obvious answer is what shape plotting toxicity against light/breeding will have.

I did have my doubts as to whether or not TFT will be shown because I know that high toxicity pushes for lower breeding rates and higher light production. This is obviously a defect/cooperate interaction, instead of a defect/defect interaction that should occur if TFT is the sort of game being played by the bacteria. The opposite is also true: low toxicity encourages high breeding and low light production, a cooperate/defect interaction.

Anyway, the results of this run are as follows:



Graphs 4a-b: Plots of the levels of light produced and the rate at which bacteria attempt to perform binary fission, versus the levels of toxicity that most commonly exist in the simulation (the plots at extreme distances away from the main group can be safely ignored). For each graph, the toxin level of 0 would best represent cooperation, and each step up the scale is an increasing level of defection from the squid. For the bacteria, left represents highest defection, and each step along the scale is towards cooperation, up until the rightmost part which is maximum cooperation.

It seems my initial doubts were unnecessary: the simulation has been able to show something closer to TFT than a reactionary response as I originally feared.

By looking at the graphs on the previous page, when the toxin levels are high (ie: the squid is defecting) around the 4 area, the bacteria that are most common in that group are ones that are also defecting: ones that have shorter intervals between binary fission attempts, and lower light production. Then, if we look at where the toxin levels are low (ie: the squid is cooperating), around the 0 area, the bacteria are closer towards cooperation (than they were when they were defecting anyway): longer intervals between fission attempts, higher light production.

So, from all the data and graphs we have observed so far, I think it is safe to say that some form of TFT is being played by these bacteria: simple organisms without recognition or an advanced suite of cognitive abilities.

This distinction is more apparent in the light levels per toxicity graph than the breeding one, although both graphs bring-up another interesting aspect of reciprocal altruism. Something that was briefly mentioned in section 1: cheaters.

4.1. The Emergence of Cheaters

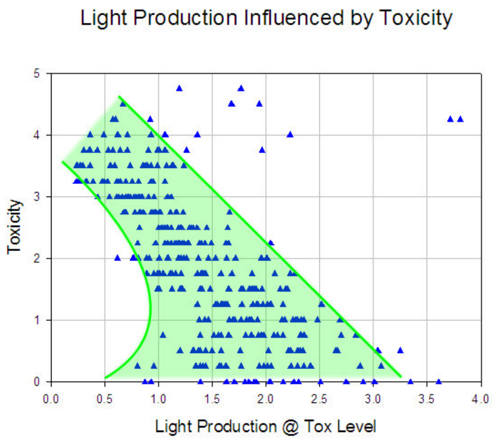


Figure 4c: The graph from the previous subsection with the areas housing the main plots shaded in green. The green shaded area represents the areas where most of the plots reside. Apart from the few extreme plots at the high toxin levels, the majority of the light production stays low. As the toxin levels die down, the plots actually take a range instead of just a single main area as it did in the high toxin level.

Upon closer inspection of the graphs we've just seen, they resemble more of a lambda shape than a basic line going from coop/coop to defect/defect. What I mean can be seen in the picture on the left.

At high toxin levels, the light production is clumped together at low. But at low toxin levels, the light production actually takes a range from low to high. Originally I was wondering where these low light / low toxin plots were coming from, but I was reminded of something mentioned about reciprocal altruism and cheaters.

Those low light / low toxin plots are actually coming from defecting bacteria that are living-off the work done by cooperating bacteria. The more cooperating bacteria there are around to reduce the toxin, the higher chance of getting a cheater into the equation. That's why we see more of these deviating values as the toxin levels approach lower levels.

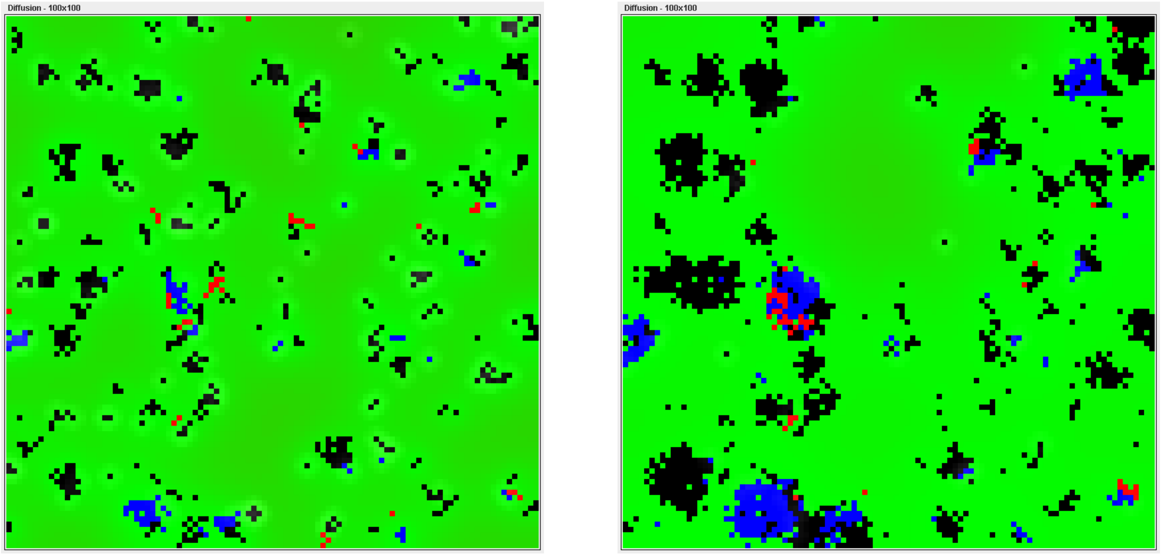
With several co-operators present in the simulation, these cheaters have several places to hide from the onslaught of incoming toxins. Especially with the 2D nature of my simulation, the best places to hide are in amongst a group of co-operators, and this has actually been the case with my observations of the simulation.

5. Patterns

It has been both interesting and tiresome to watch my simulation in action, but from it I've been able to observe some interesting patterns that arise as the bacteria fight-off the toxin that threatens to kill them.

The patterns mainly focus on the places defecting bacteria can hide, and the sorts of bacteria that are left to fight-off encroaching toxins.

5.1. Survivors



Figures 5a-b: Before and after. On the left is the layout of the simulation just as the first wave of toxin starts to hit. On the right is the shot after all those waves have passed over.

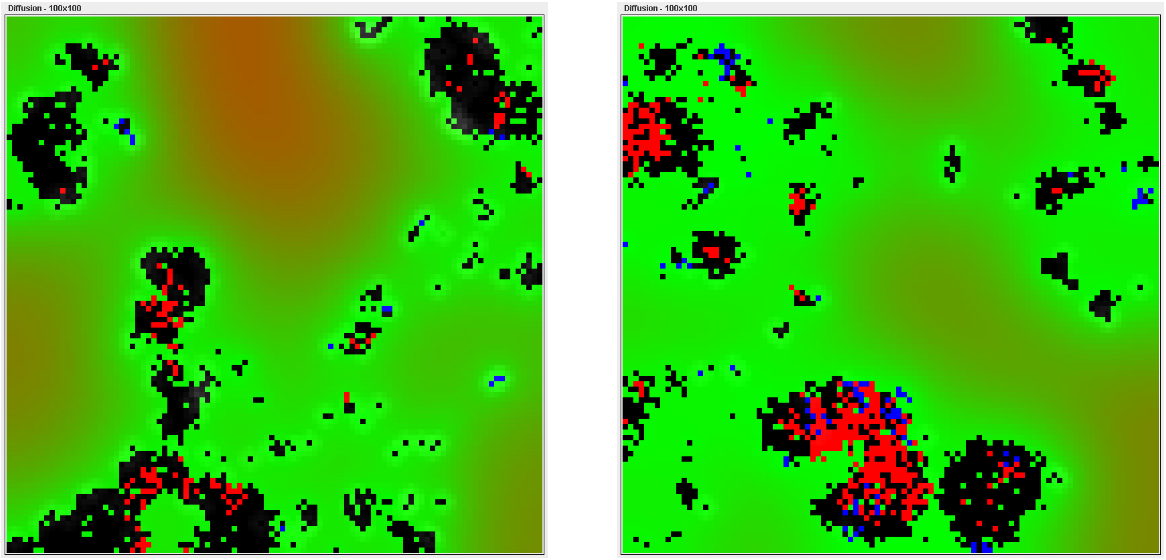
Before toxins are added, there are lots of bacteria spread randomly throughout the space. In the after shot, we can see several of the lone bacteria have now been eliminated, and the biggest survivors are the colonies of bacteria which have weathered the storm. Not only that, but these colonies have also grown.

Whether you're a co-operator or not, the lesson here is that there is safety in numbers.

5.2. Hiding Places

If it is the toxins generated by the squid that cause the bacteria to make their light, and its seen that the ones that survive heavy amounts of toxin are the co-operators (the ones the squid obviously wants), then the act of generating toxins can be thought of as a deterrent to cheating, or as a means of weeding-out cheaters. From the defect/defect run in section 5.1.3, I observed that the bacteria that have evolved to defect as much as possible were killed in the onslaught of incoming toxins, even in large groups.

After colonies of bacteria have been established, the ones in the middle are usually safe from incoming toxins. As I watched my simulation, I saw large colonies of regular bacteria look as if they were being infested from the inside-out by cheaters. In-fact, it was actually just the normal bacteria on the inside evolving into cheaters, living-off the work done by the others.



Figures 5c-d: 2 screenshots of bacteria from the same run, going from left-to-right in chronological order. The focus is on the cheaters that have managed to sneak into colonies of non-defectors.

The 2 images above illustrate it best. On the left, we see the large colony in the bottom left start to evolve some cheaters in the middles (or close to it) where bacteria are relatively safe from toxins. On the right, some thousand iterations later, the cheaters have almost taken-over the entire colony. The few in-between bacteria left have evolved to co-operators to pick-up the extra workload of toxin removal. The large area that the colony used to occupy in the bottom left corner have actually died-off because they all became cheaters and the toxins that came along were too much for them.

The situation I have described also looks like it will take place to the colony in the top-left, with more cheaters evolving from the bacteria in the middle of the colony.

5.3. Fighters

The bacteria that are at the front lines defending their colonies from toxins coming-in from the diffusion process, I call the 'fighters'. Quite often these are the normal and co-operators (non-defectors), although the latter is a bit rare to find (I had trouble obtaining a good screenshot of co-operators at the front lines).

The picture on the next page shows the places where these fighters are, reducing toxins at the frontiers of their colonies to make the living space in the middle a safer place.

Through evolution, it looks as all they ever end-up doing is harbour cheaters. At one of the earlier stages of the project, I observed what looked like a double barrier, with co-operators at the front and normal bacteria directly behind them. Behind that layer were the cheaters.

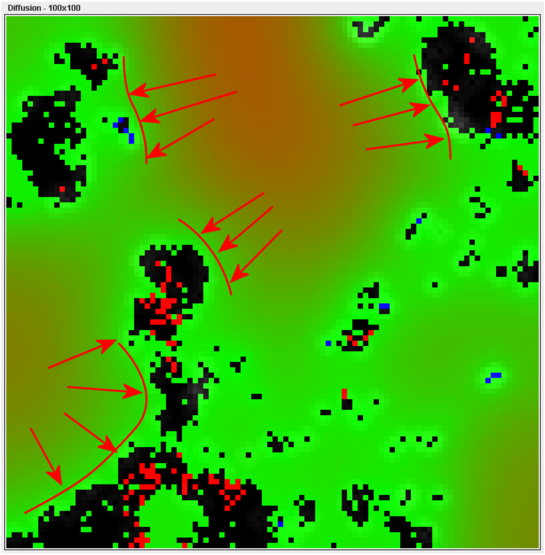


Figure 5e: The same image as 5i previously, except that I've added arrows and lines to mark the approach of toxins from diffusion and the barriers made by the fighters to reduce the toxins as they come.

From the perspective of the cooperating bacteria, they could be insanely jealous (if bacteria could feel the emotion) of the cheaters behind them taking advantage of the situation. But instead though, the act of cooperating could be seen as a selfless act: providing a haven for others of their species where they can breed and increase the population faster.

If all of the bacteria evolved towards defection, then they are easily overcome by the toxins (as seen in the defect/defect run in 3.2.3). And if they all evolve towards cooperation, the population doesn't rise as quickly (as seen in the coop/defect run in 3.2.1). Against an opponent that seeks to regulate the population, a mixed strategy produces the best results.

I wouldn't be too surprised to see similar behaviour in other creatures, not only in symbioses, but also those who fight against other opponents such as environmental factors or predators. The examples that come to mind are usually of creatures that have a lot more intelligence than our bacteria (eg: ants, meerkats), but it seems this ability to adapt rather quickly to the environment is making-up for this lack of a dedicated brain.

6. Squid Tit-For-Tat

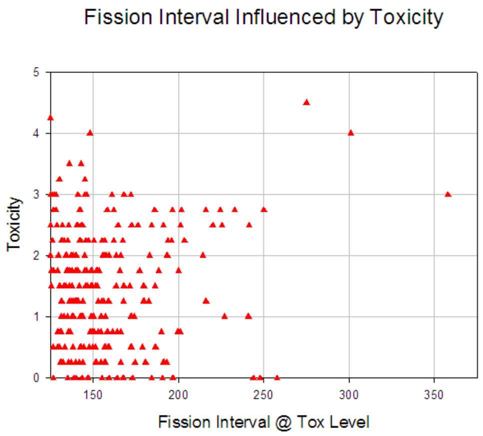
The previous results have focused heavily on TFT from the bacteria perspective (after all, it is the bacteria without the cognitive abilities). The next thing to attempt with the simulation was to see TFT from the squid's point-of-view. Since the squid effectively in-charge of the symbiosis (even though what we've seen so far points to the bacteria gaining the most benefit from the symbiosis), the squid itself must have tweaked with the conditions in its light organ ever so slightly over the centuries to create a good environment. So the next phase of the project was to play around with these values.

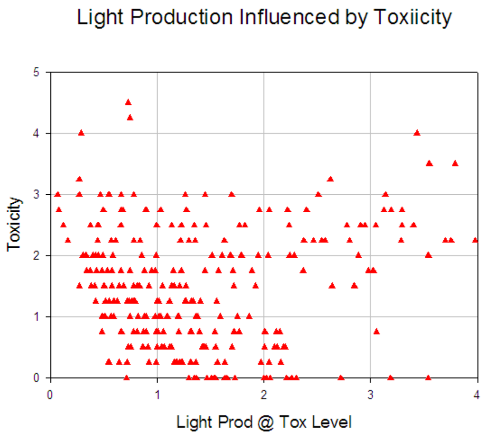
Playing around with the level of toxins that are generated, as well as the intervals at which toxin levels are raised, obviously has a great affect on the outcome. My experiences with tweaking the simulation so that I could get bacteria without killing them quickly taught me this, so I aimed at changing some of the more subtle features. The suggestion given to me by my supervisor was to alter the viscosity of the light organ. That is, the speed at which the diffusion process can take place within the light organ.

For all the test runs done so far, the viscosity has simply been 1:1 – the diffusion algorithm is calculated once for every generation. For example, after bacteria have reduced the toxins in the cell they occupy, the diffusion algorithm is then calculated for every cell, applied, thus completing a program generation. What was proposed was to alter it to other values such as 1:5 or 1:10 to see whether this has an impact on the bacteria's ability to play TFT with the squid.

The rationale behind this was that if the bacteria reduce the toxin in their area, and then the algorithm is applied several times such that it brings in lots more toxins, then the bacteria are forced to keep producing light much more often. This could severely impact their evolution strategy, and even (as an afterthought) make them less-effective fighters than those seen in the previous subsection 5.4.3.

The results on 1:2, 1:3, 1:4, and 1:5 viscosities are as follows:


Fission Interval Influenced by Toxicity


Light Production Influenced by Toxiicity

Graphs 6a-h: The TFT graphs for each of the test runs of varying viscosity. 1:2 (red, previous page), 1:3 (magenta), 1:4 (green), 1:5 (blue).

What the results tell us is exactly as my supervisor suspected: altering the viscosity of the light organ environment does have an impact on whether the bacteria can play TFT. With each successive increase in viscosity, the graph becomes more and more distorted. The light production one starts to become more random and the plots are spread all over the show, whereas the plots in the breeding ones hug the left side of the spectrum much more closely.

Once we reach the 1:5 test, there are actually very few values beyond toxin levels of 3. My observations of each one from 1:2 to 1:5 showed less and less bacteria being able to populate the simulation. At the extreme of 1:5, the population actually crashed and all died shortly before the end of 3000 generations. One reason behind this could be that when the toxins can slide below all of the bacteria, it becomes more easily able to remove all the toxins. With much less toxin residing in the simulation at these early stages, the bacteria evolves towards defection. Once this happens, a majority of the bacteria are now defectors, with very low light production levels. When the next wave of toxins hits, the current bacteria aren't able to remove it quickly with their low light production. With the effectiveness of fighter/barrier bacteria significantly decreased (toxins just sliding under from the high viscosity), not even the centre areas of colonies are safe.

So TFT has effectively been destroyed by increasing viscosity. Can it also be destroyed by reducing viscosity? To do this, I will have to be able to cut viscosity into fractions below 1:1, a capability that the simulation doesn't currently have.

7. Conclusion

This report has detailed all of the work involved with the project ‘Reciprocal Altruism for Dummies’.

Reciprocal altruism was defined and presented as a ‘you scratch my back, I’ll scratch yours’ notion, especially prevalent in the animal kingdom when dealing with both inter-species cooperation (more commonly known as symbiosis) and intra-species cooperation. It was also presented as a series of actions that required some decent processing power. Something easily found in insects, mammals, or any creature with a certain degree of intelligence.

As for the ‘dummies’ part of the project title, apart from taking a stab at a popular series of DIY-learning books in an attempt to make a subtle joke, the project was also about finding out if reciprocal altruism is possible in creatures without the smarts for reciprocal altruism. ie: dummies. eg: bacteria.

One of the first sections introduced 2 unlikely partners in crime: a tiny bobtail squid, and their even tinier light-emitting bacteria. These light emitting bacteria would be the ‘dummies’ we would test to see if they can perform a type of reciprocal altruism after all.

The next section then introduced something which seemed completely from left field: game theory and the Prisoner’s Dilemma. After a bit of explanation, the prisoner’s dilemma fit in quite well with biology as a means for explaining the evolution of cooperation. In our case, it was the maintenance of cooperation that was important, and we would use the prisoner’s dilemma and a strategy called tit-for-tat (TFT, the game theory parallel to reciprocal altruism) to look for altruistic behaviour in out bacteria.

After that, an introduction to the simulation I built to showcase the squid-bacteria relationship and to answer the questions we aimed to solve right at the beginning.

So did the results that came afterwards give us an answer? Our initial question, stated in section 1:

> *The aim of this project is to build a simulation of a biologically plausible system of very simple agents. This is to demonstrate and explore the possibility that the current standard view of reciprocal altruism is incorrect.*

So could a form of reciprocal altruism be performed by single-celled organisms?

The results showed that bacteria were able to perform TFT, and without requiring the smarts of other animals – their ability to evolve and adapt to the environment proved just as useful as recognition.

7.1. Future Work

Proposals for further work would be to explore several of the subtleties that arose along the way to getting to TFT.

One of these would be with regards to the results. Not only did the results show that the bacteria were playing TFT, they were doing a whole lot more. The TFT we saw wasn't as clear-cut as the way TFT is actually described:

> *Defect if the other party defects, cooperate if the other party cooperates*

The TFT we saw was more like this:

> *Defect if the other party defects, cooperate if the other party cooperates, or defect if there's enough co-operators such that they can pick-up my share of the work!*

In fact, it looked more like reciprocal altruism than just plain-vanilla TFT (the in-depth definition of reciprocal altruism does cater for cheaters: individuals who don't choose to cooperate).

Another of the subtleties was how viscosity of the light organ played a role in the ability for bacteria to play TFT. We saw how increasing the viscosity of the light organ slowly destroyed the bacteria's ability to play TFT, and when it became too extreme, destroyed the population of bacteria altogether. The work that could take place here would be to explore the consequences of reducing the viscosity of the light organ: an ability the simulation currently doesn't have.

Exploring the effect of patterns in the simulation could also be an avenue for future work. We saw how there was safety in numbers: a prime concept seen throughout the animal kingdom. We also saw how the formation of a bacterial colony was designed to provide optimal growth conditions for the population as a whole: selfless behaviour in a creature that doesn't even know the meaning of the word.

Lastly, when this project began, not too much was known about the *scolopes-fischeri* relationship. Much of the maintenance of cooperation between the 2 still remains a mystery. Future work here would include using and applying newer research and findings to further delve into this very mysterious symbiosis.

8. References

Axelrod, R. (1984), *The Evolution of Cooperation*, Basic Books, New York.

Dugatkin, L.A. (1997), *Cooperation Among Animals: An Evolutionary Perspective*, Oxford University Press.

Foster, K.R. (2004), *Diminishing returns in social evolution: the not-so-tragic commons*, Available: http://www.ruf.rice.edu/~evolve/pdf/Kevin/Dimreturns.pdf.

Frean, M.R. (1996), The Evolution of Degrees of Cooperation, *Journal of theoretical Biology*, Vol. 182, Issue 4, Pp. 549-559, Academic Press Ltd.

Frean, M.R. & Abraham, E.R. (2004), Adaptation and enslavement in endosymbiont-host associations, *Physical Review E*, Vol. 69, Article# 0501913, Journals of the American Physical Society.

Graf, J. (2003), *Vibrio fischeri – Euprymna scolopes symbiosis*, Available: http://www.sp.uconn.edu/~mcbstaff/graf/VfEs/VfEssym.htm.

Healthy Buildings International Inc (2002), *Health Issues: Chlorinated Swimming Pools*, Available: http://www.hbiamerica.com/s2/SwimmingPools.pdf.

Hu, Y.F. & Blake, R.J. (2000), *Load Balancing for Unstructured Mesh Applications,* Available: http://www.dl.ac.uk/TCSC/Staff/Hu_Y_F/PROJECT/pdcp_siam/pdcp_siam.htm.

Kimbell, J.R. & McFall-Ngai, M.J. (2003), The squid-vibrio symbioses: From demes to genes, *The Society for Integrative and Comparitive Biology Journal*, Available: http://www.findarticles.com/p/articles/mi_qa4054/is_200304/ai_n9170373.

Kuhn, S. (2003), Prisoner's Dilemma, *Stanford Encyclopedia of Philosophy (Fall 2003 Edition)*, Available: http://plato.stanford.edu/entries/prisoner-dilemma/.

Marine Biology Laboratory (MBL), (2005), Marine Organisms: Marine Organisms Database, *Marine Organisms Database*, Available: http://www.mbl.edu/marine_org/marine_org.php?func=detail&myID=Cep-140.

Maynard Smith, J. (1982), *Evolution and the Theory of Games*, Cambridge University Press, Cambridge.

McFall-Ngai, M.J. (1999), Consequences Of Evolving With Bacterial Symbionts: Insights from the Squid-Vibrio Associations, *Annual Review of Ecology, Evolution, and Systematics*, Vol. 30, Pp. 235-256, Annual Reviews.

McFall-Ngai, M.J. (2005), *Kewalo Marine Lab Website*, Pacific Biomedical Research Center, University of Hawaii, Available: http://www.kewalo.hawaii.edu/labs/mcfall-ngai/.

Meredith, C. (1998), Tit for Tat, *The Lab – Australian Broadcasting Corporation's Gateway to Science*, Available: http://www.abc.net.au/science/slab/tittat/story.htm.

Okasha, S. (2003), Biological Altruism, *Stanford Encyclopedia of Philosophy (Summer 2005 Edition),* Available: http://plato.stanford.edu/entries/altruism-biological/.

Paustian, T. (2004), Mutualism under the sea, *Microbiology Textbook: The World of Microbes*, Available: http://www.bact.wisc.edu/Microtextbook/modules.php?op=modload&name=Sections&file=index&req=viewarticle&artid=120&page=1.

Ruder, K. (2004), Hawaiian Bobtail Squid Glows from Below, *Genome News Network*, Available: http://www.genomenewsnetwork.org/articles/2004/01/09/squid.php.

Sachs, J.L., Meuller, U.G., Wilcox, T.P., Bull. J.J (2004), *The Evolution of Cooperations*, Available: http://www.journals.uchicago.edu/QRB/journal/issues/v79n2/790201/790201.html.

Stevens, J.R. & Cushman, F.A (2004), *Cognitive constraints on reciprocity and tolerated scrounging*, Available: http://www.wjh.harvard.edu/~jstevens/cv/pubs/stevens.cushman.2004.pdf.

The BBC (Mar 2005), *In-depth aricles – Behaviour - Reciprocal Altruism*, Available: http://www.bbc.co.uk/nature/animals/mammals/explore/altruism.shtml.

Wikipedia, (Aug 2005), Bacterial growth, *Wikipedia: The Free Encyclopedia*, Available: http://en.wikipedia.org/wiki/Log_phase.

Wikipedia (Jan 2005), Reciprocal altruism, *Wikipedia: The Free Encyclopedia*, Available: http://en.wikipedia.org/wiki/Reciprocal_altruism.

Wikipedia (April 2005), Tit for Tat, *Wikipedia: The Free Encyclopedia*, Available: http://en.wikipedia.org/wiki/Tit_for_Tat.

Wikipedia (Feb 2005), Vampire Bats, *Wikipedia: The Free Encyclopedia*, Available: http://en.wikipedia.org/wiki/Vampire_bat.

Wood, J.B. (1999), Euprymna scolopes, *The Cephalopod Page*, Available: http://www.nhm.ac.uk/hosted_sites/tcp/Escolopes.html.