Faculty of Science

An exploration of harvesting strategies in a spatial dynamic game

Dominic Hunt

Degree project for Erasmus Mundus in Complex Systems Science
30 hec

Department of Physics
University of Gothenburg

UNIVERSITY OF GOTHENBURG

Abstract

Strategies are proposed and studied for finding the optimal action each timestep in a dynamic game, mimicking the harvesting of a natural resource. The performance of these strategies was studied for the cases where one or two players were competing for a resource. These results were then used to try and understand a toroidal grid of interconnected two player games, with each player taking part in four games. Each player could change strategies based on the performance of their neighbours. One strategy was found that was able to consistently reach the optimal harvest and dominate in the lattice, while still being dominated by another strategy in single-pool games.

Contents

Introduction

In the real world, different situations may appear to be identical. The same action would in such situations produce different responses despite there being no discernible reason for this. That is to say, the state of the system is not fully described by what can be observed. Therefore, for predictable responses to actions, recent interactions will also have to be taken into account, "learning" from past examples of similar local histories.

This is especially true when harvesting a renewable resource, where natural regrowth, along with the actions of any actors, will vary the state of the system. Harvesters aiming to maximise their long term harvests would wish to identify the most stable way of reaching their goal. Whilst in general this is not possible, for systems whose regrowth is deterministic, strategies can be developed to identify optimal harvesting levels or cycles. Harvesting wood, for example, requires many years of waiting before cutting a tree down, whereas parsley or algae can be harvested all year round without significant seasonal effects. When multiple harvesters compete for the same resource they will not only be aiming to stabilise the resource levels with their harvesting intensity, but also aim to do so in a way that maximises their own harvests with respect to the other harvesters.

The identification of the optimal harvesting levels is made all the more difficult by a harvester's inability to fully estimate the current state of a resource. At any stage a harvester does not know what the capacity of a resource container is or the growth rate. As harvesting tends to take a much shorter amount of time than the identification of equilibria, there will be a tendency to over harvest what appears at first to be an inexhaustible supply of the resource. Only by comparing current states to those in the past can changes be clearly seen. For this to be possible, harvesters would have to be able to identify how the resource quantity has changed over time. This will allow the harvester to reach the optimal harvesting levels without an explicit knowledge of whether a given harvest intensity will increase or decrease the harvest in the long term.

This thesis looks into simple strategies, for two-player dynamic games, that reach optimal harvests with other players when the payoffs vary in a way similar to a harvested renewable resource. Players will always have the same goal, to maximize their long term profit by optimising their harvesting intensity. We will then describe how those strategies compare and how well they perform against each other.

To allow the impact of the strategies to be clearly seen, the model has been designed to have the most simple base components possible, both in terms the natural growth function, and the deterministic nature of the harvests. Likewise, when studying how the strategies compete against each other the connections between players should be as uniform as possible, while still constraining each harvester's neighbourhood to only a few other harvesters. This will allow a strategies success to be independent of the players they start with.

To this end, a square lattice with period boundary conditions is used, with a player at each node and a resource pool on each edge. Each player will harvest from four resource pools and each resource pool will only have two players harvesting from it. Players will be able to compare results with their four neighbours and identify whose strategy is performing the best.

For the players to be able to compare how they are performing, they will have access to the harvests of all players harvesting in the same pools as them. Players are aware of their neighbour's harvest quantities from their shared pool, as well as their weighted time-average of the overall harvests. From this information they will respectively attempt to adjust their harvest intensities in each resource pool, as well as choose which strategy they think adjusts the harvest intensity to produce the highest harvest.

The only information a player has on the strategy used by a neighbour to update its harvest intensity is the quantity they harvest each timestep. With only this as a guide the best a player could hope for is to assess the probability that their neighbour is using a specific strategy. As this will add another layer of complexity onto the study, we will assume that players can find out the strategy used by their neighbour when they need to, along with the values of any parameters used for that strategy.

This report is split into three main sections. First, an overview of the literature is provided and this current investigation is placed into that context. Next, the model is described, and the necessary considerations are outlined. Player strategies are then described and their characteristics are explored for different simplified versions of the model, notably for one player and one pool, two players and one pool, before finally looking at the dynamics of the whole grid system.

Background

A significant interest of game theory has been the identifications of strategies that maximise a player's profit. Some games aim to show a conflict between wanting to maximise a player's possible profits and minimise their possible losses. The classic example of this is the *prisoners' dilemma*, (PD), whose payoffs are shown in table 1, for $T > 1$ and $0 > S$. When a player takes part in a *one-shot* PD game, where there will be no repercussions for them once it is over, a players benefits most from *defecting*. Defecting allows them to maximise their maximum and minimum profits, T and 0 respectively. As both players have identical profits they will both want to defect, giving them each a profit of 0. However, if they were both to *co-operate* they would end up with a higher profit, 1. Given this additional knowledge it is still more profitable to defect, since $T > 1$, resulting in the maximum payoff for the player who does so.

For *one-shot* games the preferred strategy is therefore to defect. If instead the players have to accumulate the highest profits over many repeats of the game then cooperating would appear to be even more attractive. The classic work of Axelrod and Hamilton[3]

Player 1	Player 2	
	Co-operate	Defect
Co-operate	$(1, 1)$	(S, T)
Defect	(T, S)	$(0, 0)$

Table 1: The payoffs for a prisoner's dilemma game, where $T > 1$ and $0 > S$

Player 1	Player 2	
	A	B
A	(a, b)	$(0, 0)$
B	$(0, 0)$	(b, a)

Table 2: The payoffs for a coordination game, where $a \geq b$

found that while pure defection is an *evolutionary stable strategy* while the majority of players are defecting, when a broader range of strategies are available *Tit-for-Tat*, (TFT), will produce higher or equal profits. TFT has the player only defecting when the other player had done so in the previous turn, acting as a *punishment* for the other player and as a protection for the player should the opponent have chosen to continue defecting.

The term *co-operate* in repeated games has been used to refer to conditions under which a player would consent to not maximise short term profits in favour of long term gains by putting themselves in a position where they could loose significantly if the other player does not also co-operate. It is this form of cooperation that will be studied throughout this thesis.

In expanding the PD game to many players, one approach is the N-person prisoner-dilemma game, where the payoffs depend only on the fraction who choose to co-operate. For example for a game with n players the payoff to each of the k cooperating players would be $\frac{k}{n-1}$. Eriksson and Lindgren[5] used this for a repeated N-person PD game, where the strategies of the players themselves do not evolve, but instead the players are partially replaced after each timestep. Their replacements are chosen such that the proportion of replacements coming from a specific strategy is dependent on the success of that strategy in the previous timestep. While this method allows for a clear understanding of the robustness of different strategies we are interested in the survival of strategies for persistent players.

Gradual diffusion of strategies across a set of players can be achieved by restricting the neighbourhood of the players. This produces a *network* of links between players through which we can gain an understanding on how strategies will propagate through a social group, with networks such as Erdős-Rényi random graphs or and Barabási-Albert preferential attachment networks, or how strategies will propagate through space, with lattice-like networks.

One study[4] focused on how the structure of a network affected the propagation of an action, or *opinion*. It had players choose which of two actions to perform based either on a 'generalised simple majority' from neighbours, or on the cumulated results of playing the co-ordination game, shown in table 2. In this game the players will maximise their payoffs by both playing the same action as each other. Both players are 'punished' for a lack of coordination by receiving a much smaller payoff than they would otherwise. An extra tension is added by making the payoffs asymmetric; the payoffs for playing the same action will not be the same for both players, with each profiting most from different actions. This repeated game was used to study the effects of small world and scale-free networks on the density of cooperation and identifying apparent phase transitions.

The latter of the two propagation methods, comparison of average success with that of neighbours, is a common one in network diffusion studies, and a variation on this will be used in this thesis. PD games are frequently used as the interaction between the two players across a link. One such example[6] looked at the densities of stable and unstable cooperators when varying between Erdős-Rényi and Barabási-Albert networks. It found variations in the number of stable cooperators and stable defectors for different variations on the two network structures.

Another approach[11] to studying the impact of the network structure is to not only allow players to modify their strategy, but also modify who their neighbours are when playing a PD game. Defectors were found to try and surround themselves with cooperators

The models discussed so far have all dealt with the success of fixed strategies over repeated plays of a game. Mechanisms for the evolution of strategies have been studied, such as by Lindgren[8], who looked at finite memory strategies in the N-player iterated PD. The strategies had a response for every combination of an opponent's past actions. These then evolved through point mutations, duplications (extending the history evaluated by one timestep), or splitting in half, (reducing the memory evaluated by one timestep). The number of offspring a player has going into the next timestep was based on the relative success of the strategy. This style of strategy was later applied to a lattice-based network[9]. The style allowed for a broad strategy space to be explored without much being hard-wired, which is ideally what we should aim for in strategy analyses.

Broadening the number of actions a player can choose necessitates taking a different approach in constructing strategies, as the decisions a strategy must take are no longer binary. In this thesis the strategies used will have to contend with time-varying payoffs. To act in response to the latest state of the system players will have to choose from a continuous range of possible actions. Such *dynamic games* have been studied with varied games and group structures. For example, Akiyama and Kaneko[1] looked at different strategies for the *lumberjack game*[2] and how it works in the context of a dynamical system. This game requires the players to choose when to harvest wood as the trees grow. Different payoff functions were used and their impact on the level of resources was studied as the players tried to determine the optimal harvest time.

The difficulty in constructing strategies for such games is that they must balance correctly identifying the long-term impact of an action with trying to maximise the harvest. Ideally a player would wish to identify the optimal actions as fast as possible, but the faster they choose to change their actions the more the impact of individual changes will go unnoticed. There is therefore a balance to be maintained between testing potentially better actions and identifying the actual impact of those changes.

An example of how a small change can have a big impact is the *tragedy of the commons*, described by Hardin in his 1968 paper as follows[7]:

> Picture a pasture open to all. It is to be expected that each herdsman will try to keep as many cattle as possible on the commons... As a rational being, each herdsman seeks to maximize his gain. Explicitly or implicitly, more or less consciously, he asks, "What is the utility to me of adding one more animal to my herd?" This utility has one negative and one positive component...The positive component is a function of the increment of one animal...The negative component is a function of the additional overgrazing created by one more animal. Since, however, the effects of overgrazing are shared by all the herdsmen, the negative utility for any particular decision-making herdsman is only a fraction of -1. Adding together the component partial utilities, the rational herdsman concludes that the only sensible course for him to pursue is to add another animal to his herd. And another; and

> another.... But this is the conclusion reached by each and every rational herdsman sharing a commons. Therein is the tragedy.

What appears to be a small change for an individual participant leads to a large impact when all participants choose to maximise their gains through this small change. The impact of these changes are small enough that they will only be felt after some time. For participants not to collectively over harvest they must choose to co-operate with each other and not fully maximise their short term profits.

For the dynamic game we will be studying in this thesis, the strategies will have to identify the optimal harvesting intensity and resource pool level without the explicit knowledge of whether a given harvest intensity will cause a long-term increase or decrease in their harvest.

Model

The general model is structured as a toroidal square grid, with players on the nodes and resource pools on the edges. Players aim to harvest the most resources possible from the adjacent resource pools over an infinite number of timesteps. The quantity of resource in each resource pool will grow each timestep based on the resource it had at the end of the previous timestep. Each player can see the harvests of all players in each pool it harvests from, as well as the average total harvest of their neighbours. At the end of each timestep the player decides on how to update its harvest intensity for each resource pool. One of a number of specified strategies is used for this update, with the strategy used depending on the player.

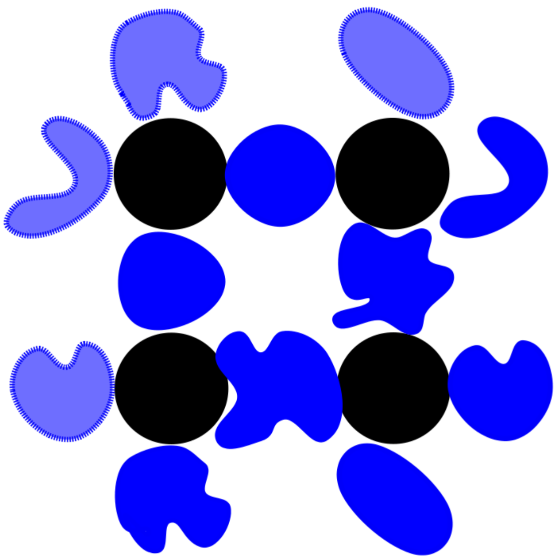


Figure 1: A representation of the toroidal 2×2 grid, with 4 players as the black circles and 8 pools. The faded pools are used to represent the lattice's periodic boundary conditions.

Model basis

A resource pool is an edge p on the network containing a quantity of resource $r(p)$, where $r(p) \in \mathbb{R}_{\geq 0}$. A player, $n \in \mathbb{N}_0$, may interact with resource pools adjacent to their position and choose to harvest a fraction, $h(\tau, n, p) \in]0, 1[$, from the pools each timestep τ.

At the end of each timestep the quantity remaining in the resource pool, $r(\tau, p)$, increases by a quantity

$$\Delta r(\tau, p) = g\, r(\tau, p) \left(1 - \frac{r(\tau, p)}{C}\right) - \sum_{n \in N_p} h(\tau, n, p)\, r(\tau, p)$$

where $g \in \mathbb{R}_{\geq 0}$ is the growth rate of the resource and $C \in \mathbb{R}_{\geq 0}$ is the resource pool capacity. Both of these parameters are constant throughout the simulation. This can be written in a reduced form by defining $r' = \frac{r}{C}$

$$\Delta r'(\tau, p) = g\, r'(\tau, p) \left(1 - r'(\tau, p)\right) - \sum_{n \in N_p} h(\tau, n, p)\, r'(\tau, p) \tag{1}$$

During a given timestep the players adjacent to a resource pool will interact simultaneously. However, if

$$\sum_{n \in N_p} h(\tau, n, p) > 1$$

it will be normalized

$$h(\tau, n, p) = \frac{h(\tau, n, p)}{\sum_{n \in N_p} h(\tau, n, p)}$$

This will keep the ratio of the harvesting intentions of each player the same.

The players consider that they are trying to maximise their average total harvest over an infinite number of timesteps, and so have no termination point.

System equilibria

From this characterisation we can also define values for the optimal harvest intensities h^* and pool resource quantities r'^*.

We know that when the system reaches equilibrium $\Delta r' = 0$, so from equation 1 we can write this as a Lagrangian problem, maximising $\sum_{n \in N_p} k(n) = \sum_{n \in N_p} h(n)\, r'$:

$$\mathcal{L} = \max \sum_{n \in N_p} h(n)\, r' - \lambda \left(\sum_{n \in N_p} h(n) - g\left(1 - r'\right) \right)$$

$$\frac{\partial \mathcal{L}}{\partial h(n)} = r' - \lambda$$

$$\frac{\partial \mathcal{L}}{\partial r'} = \sum_{n \in N_p} h(n) - \lambda\, g$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = \sum_{n \in N_p} h(n) - g\left(1 - r'\right)$$

$$\sum_{n \in N_p} h(n) = g\left(1 - r'\right)$$

Setting the partial differentials to zero and combining their results appropriately we can see:

$$g\left(1 - r'\right) = g r'$$

So $r'^* = 0.5$, which implies:

$$\sum_{n \in N_p} h(n) = \frac{g}{2}$$

As all the harvest intensities will be the same at the optimum

$$N_p \, h(n) = \frac{g}{2}$$

We therefore have $r'^* = 0.5$ and $h^*(n,p) = \frac{g}{2N_p}$.

Players

A player interacts with resource pools with an aim to maximise its total payoff. The strategy chosen to achieve this will potentially vary from player to player. Their choice of harvesting intensity, for each pool, at each timestep, is based on the effect of their past actions in that pool as well as those of their neighbour using that pool.

For all strategies studied, if a player finds that one of the resource pools they harvest is producing no harvest, then they do not change the harvest intensity for that pool. This is unlikely to occur as it would require the total harvest intensity of all the players harvesting a pool to be greater or equal to one.

Each player has a series of different memories stored. These memories are stored for each timestep even if a player's strategy does not use them. The memories can be broken down into pool-based memories and global memories, that is memories associated with the state of an individual pool and memories associated with the player's overall state.

The memories for each pool are:

- The intended harvest per timestep, $h(\tau, n, p)$.
- The quantity of resource harvested, for each player per timestep $k(\tau, n, p)$.
- A *fitness value* of each player $k_{\text{av}}(\tau, n, p)$, calculated based on $k(\tau, n, p)$. The specific form of the calculation is described in the following section. It is also referred to as a *memory of success.*

From these memories a player can update its own harvest intensity for that pool at the end of the timestep. Global memories are used to evaluate any strategy changes. These memories are:

- The strategy the player used in each timestep
- A fitness value for the player overall $k_{\text{av wealth}}(\tau, n)$, taken from the harvests of all pools a player harvests in and updated in a simmilar way to k_{av}. This is also referred to as the *wealth* of a player.

Modifiable parts

The description of the model so far has dealt with the parts that will always remain the same for all players in all instances of the model. However, other aspects will vary across individual players in each instance of the model. They are:

- The distribution of initial harvest intensities for each player $h(0, n, p)$.
- The method of calculating their own and their neighbour's success, $k_{\text{av}}(\tau, n, p)$. The value of which will be calculated based on the quantity harvested by each player $k(\tau, n, p)$.
- The strategy for choosing the harvest intensity. This can be changed potentially for each user and at each timestep.
- The chosen harvesting intensity per pool, per user, per timestep $h(\tau, n, p)$. This is calculated based on the strategy the player is currently following. If an individual player harvests in multiple pools the harvest intensity chosen is unlikely to be kept the same for all pools as it is calculated separately for each one based on the memories associated with a pool.

Although we are only considering one growth function, the strategies considered should be able to deal with a more general class of growth functions. The general class of growth functions is considered to have some restrictions imposed upon it. The functions must be continuous for all values, with no growth outside of a user-specified window. This implies that there is a saturation point to the resource pool, which has the advantage of implying that there exists an optimal harvesting intensity h^* and an associated optimal resource pool saturation point r^*. We will also impose the restriction that the optimum is unique.

Fitness of a player and its neighbours

The *fitness* of a player in a given pool is based on their harvesting success along with the assumption that players will remember more clearly recent events. The player fitness, or memory of success, per pool has been implemented as a weighted sum of the quantity harvested in the current timestep $k(\tau, n, p)$ and the previous result from the weighted sum, $k_{\text{av}}(\tau - 1, n, p)$. A weighting factor, $i \in [0, 1]$ is introduced to replicate the impact the latest harvest has on the 'perceptions' of a player.

$$k_{\text{av}}(\tau, n, p) = (1 - i)\, k_{\text{av}}(\tau - 1, n, p) + ik(\tau, n, p)$$

The parameter i is specified at the start of each simulation and is kept constant for all players.

An almost identical function is used to calculate the current wealth of each player, using a weighting factor $i_w \in [0, 1]$.

$$k_{\text{av wealth}}(\tau, n) = (1 - i_w)\, k_{\text{av wealth}}(\tau - 1, n) + i_w \sum_p k(\tau, n, p)$$

The parameter i_w is also specified at the start of the simulation and is kept a constant for all players. For all simulations shown in this report it will be kept the same as i, but we will continue to make the distinction for clarity.

Strategy modification

In the networked system players are able to change their strategy depending on which they perceive to be performing the best. They will do so by evaluating which of those around them has the current maximum wealth.

$$\text{player ID}\left(\max\left(k_{\text{av wealth}}\right)_{n\cup\text{neighbours of n}}\right)$$

If that person is not using the same strategy, then the player will use this new strategy.

This evaluation is performed randomly following a poisson distribution when

$$\text{poisson}\left(w\right)\geq w_{\text{thresh}}$$

with w as the mean of the poisson distribution and w_{thresh} as a threshold for the random selection.

Having now outlined the implementation of the model we can now look at the various strategies employed by the players and their successes.

Strategies and their performance

Simulation methodology

For ease of comparison between strategies and simulations a few of the simulation parameters have been kept the same throughout. The resource pool capacity, C has been arbitrarily set to 40 as the actual value has no impact on the results. The growth factor, g, has been set to 0.5. Lower growth factors were studied, but were found to have no effect on the dynamics of the simulations, other than to lengthen the time to stability for the strategies using k_{av}. The initial pool resource quantity has been set to $r'(0) = 0.5$ for simplicity. In the cases where this initial value has an apparent impact for more than a few timesteps its effect has been discussed.

These parameters, along with i, i_w, the maximum initial harvest intensity and the number of timesteps are initialised at the start of a simulation. In the case of one harvester or two harvester simulations the strategies used by each harvester are defined at the start, along with any necessary parameters. The method of assigning strategies to harvesters in the network simulation is described in the associated section.

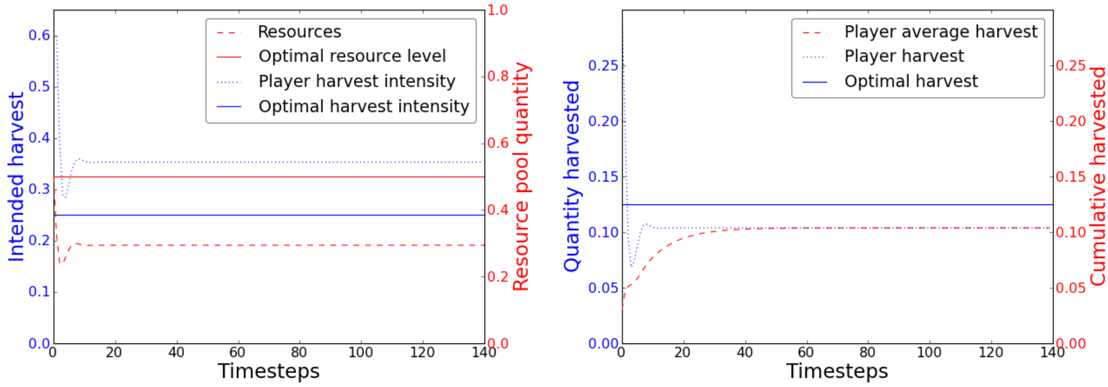
The simulations themselves have been written in *Python* using the *Scipy* modules and plotted using *Matplotlib*. The plots shown are from individual runs of the simulations and have been chosen as representative of the general dynamics of the model instance, while still clearly showing the important aspects to be discussed.

One harvester

In these simulations we look at the interactions between one player and one resource pool. The strategies studied are therefore purely aiming to maximise the harvest of the player. Each strategy will be looked at in turn and the results analysed before moving on to the next one.

Reaction to change in resource level

The initial one-harvester strategy studied estimates the changes in the resource pool quantity and tries and stabilize it. This is achieved by looking at the relative changes in the harvest intensity and the harvest from one timestep to the next. This change is



(a) Harvest intensity and pool resource quantity (b) Quantity and average quantity harvested

Figure 2: The time evolution for a player using the reaction strategy and a resource pool with a growth rate of 0.5. The discount rate for the average is taken to be 0.1.

used as the correction to the harvest intensity for the following timestep:

$$\begin{aligned} h(\tau+1) &= h(\tau)\frac{\text{Change in harvest}}{\text{Change in harvest intensity}} \\ &= h(\tau)\frac{k(\tau)}{k(\tau-1)}\frac{h(\tau-1)}{h(\tau)} \\ &= h(\tau)\frac{r(\tau)}{r(\tau-1)} \end{aligned}$$

In the case where $\tau = 0$ this is strategy is:

$$h(\tau+1) = h(\tau)$$

We can see figure 2 that the strategy leads to the system stabilising quickly to a harvest intensity and a resource pool quantity that are not at the optimum. In fact, the strategy will, depending on the initial harvest intensity and initial resource pool quantity, converge to a point on a line of harvest intensity and resource pool quantity pairs. This can be seen in figure 3, which shows the dynamical system produced by a player using this strategy.

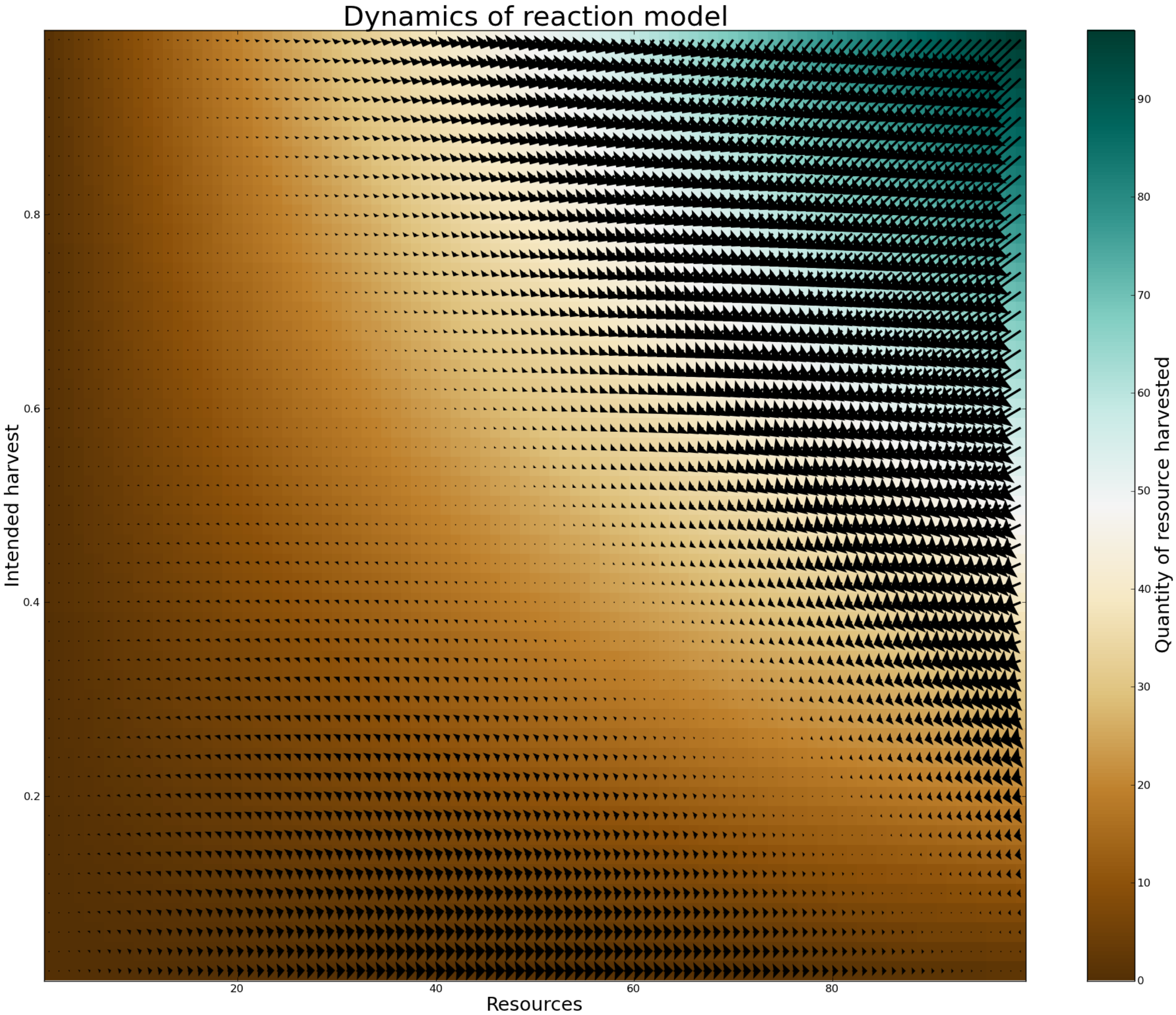


Figure 3: The dynamics of a system where the player is using the reaction to change strategy and where the growth rate is set to 0.9. The arrows denote the magnitude and direction of any change. A series of stable points can therefore be seen crossing the plot

Reaction to change in harvest

To simplify the strategy looked at above we looked at stabilising the changes in harvest. This is achieved by looking at the relative changes in the harvest from one timestep to the next and using this as the correction to the harvest intensity for the following timestep:

$$h(\tau+1) = h(\tau)\frac{k(\tau)}{k(\tau-1)}$$

Once again, when $\tau = 0$:

$$h(\tau+1) = h(\tau)$$

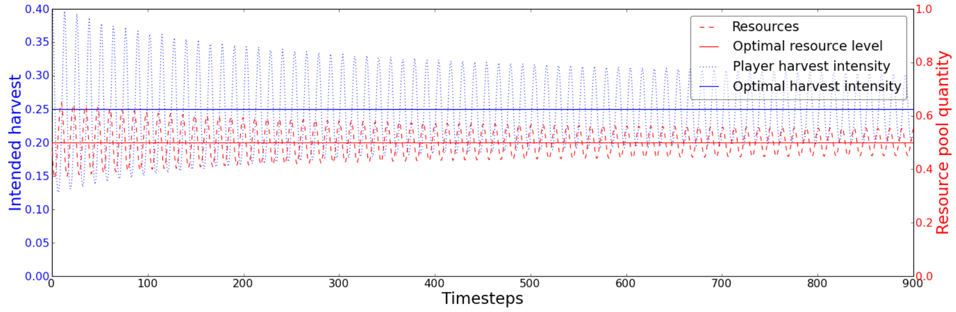


Figure 4: The time evolution of the harvest intensity and pool resource quantity for a player using the harvest reaction strategy and a resource pool with a growth rate of 0.5.

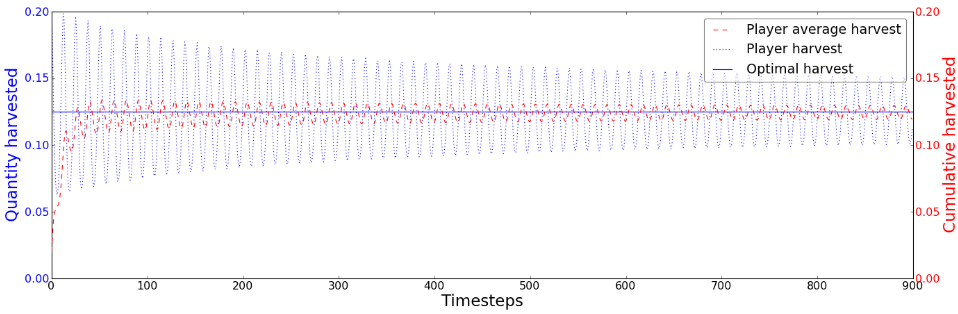


Figure 5: The time evolution of the quantity harvested and the average quantity harvested for a player using the harvest reaction strategy and a resource pool with a growth rate of 0.5. The discount rate for the average is taken to be 0.1.

The dynamics of the convergence of this strategy appears to hold much more promise than the previous strategy. We can see from figures 4 and 5 that the player converges on the optimal harvest intensity and lets the resource pool reach the optimal quantity. This is found to work for all initial harvest intensities, with sharp changes in the intial harvest intensity, followed by decaying fluctuations. However, if the initial resource pool quantity is changed from $r' = 0.5$ to another value, the system will reach an equilibrium near, but not quite the optimum. This can be seen in figures 6 and 8, showing when $r' = \frac{3}{8}$ and $r' = \frac{1}{8}$ respectively.

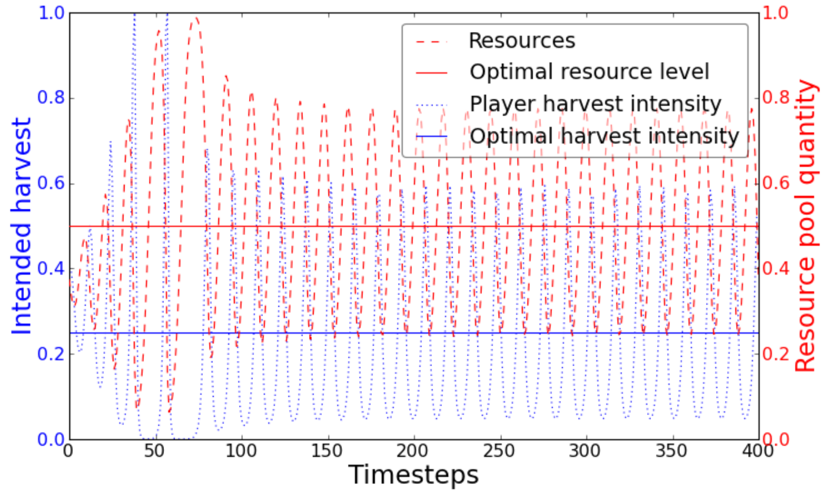


Figure 6: The time evolution of the harvest intensity and pool resource quantity for a player using the harvest reaction strategy and a resource pool with a growth rate of 0.5. The initial resource pool quantity has in this case been set to $r'(0) = \frac{3}{8}$

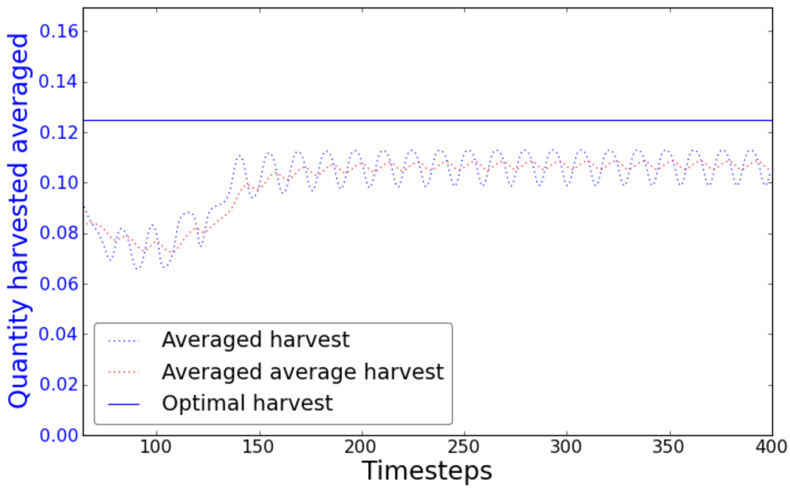


Figure 7: The time evolution of the quantity harvested and the average quantity harvested for a player using the harvest reaction strategy and a resource pool with a growth rate of 0.5. Both have been averaged using a moving average. The discount rate for the average is taken to be 0.1.The initial resource pool quantity has in this case been set to $r'(0) = \frac{3}{8}$

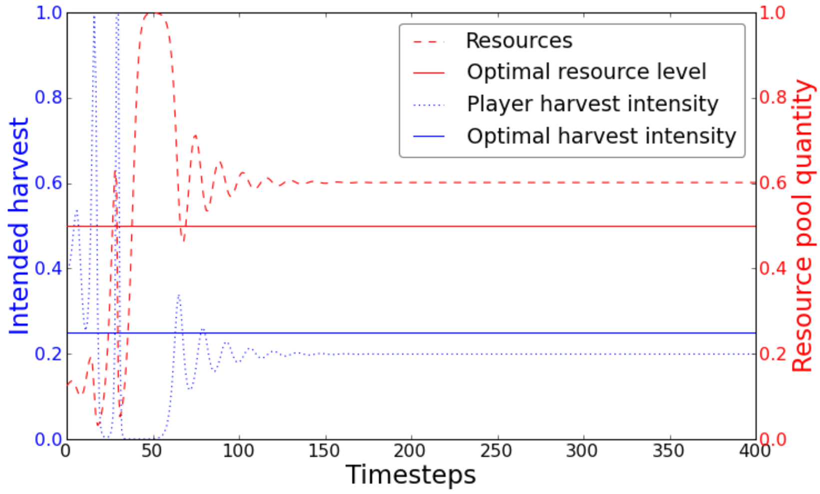


Figure 8: The time evolution of the harvest intensity and pool resource quantity for a player using the harvest reaction strategy and a resource pool with a growth rate of 0.5. The initial resource pool quantity has in this case been set to $r'(0) = \frac{1}{8}$

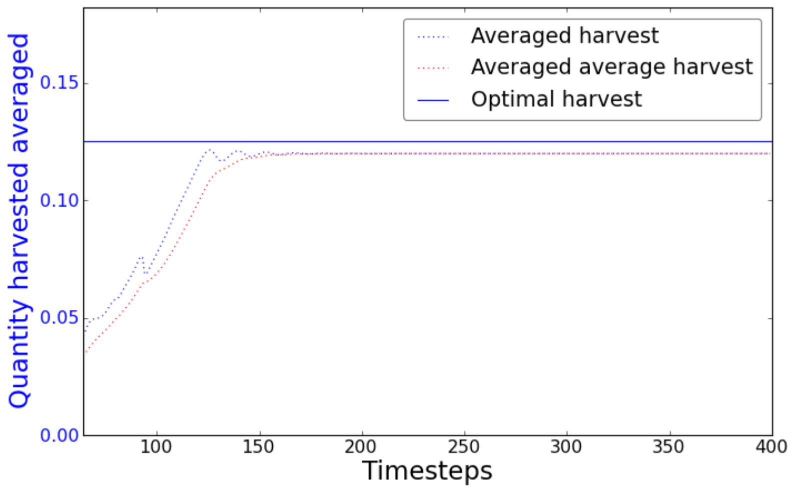


Figure 9: The time evolution of the quantity harvested and the average quantity harvested for a player using the harvest reaction strategy and a resource pool with a growth rate of 0.5. Both have been averaged using a moving average. The discount rate for the average is taken to be 0.1.The initial resource pool quantity has in this case been set to $r'(0) = \frac{1}{8}$

Probabilistic jumping

The strategies shown up till now have all led to a series of stable points being found. We would like to find strategies that get the system to jump out of these stable points and reach the optimal stable point. To achieve this a strategy must be able to have a random component to its motion. This is achieved by having the harvest intensity change value each timestep with a given probability. The probability is dependent on the relative size of the change in quantity harvested with respect to the average quantity harvested so far.

We define a fixed increment for changes in the harvest intensity, Δh, constant for the whole simulation. We are interested in looking at the likelihood that the harvest will improve if the harvesting intensity is changed to $h' = h \pm \Delta h$. The sign of the change will vary and the choice of sign is defined below. We can estimate the change in harvest based on the last timestep by considering:

$$\begin{aligned}\Delta k &= \Delta h \, r \\ &= \Delta h \, \frac{k}{h}\end{aligned}$$

In this way we can construct a relative impact of the change by comparing it to the current average success, k_{av}. The probability of changing to this new harvest intensity is therefore defined as:

$$\frac{p(h')}{1 - p(h')} = \exp^{-\frac{\Delta k}{k_{\text{av}}}}$$

resulting in a probability of:

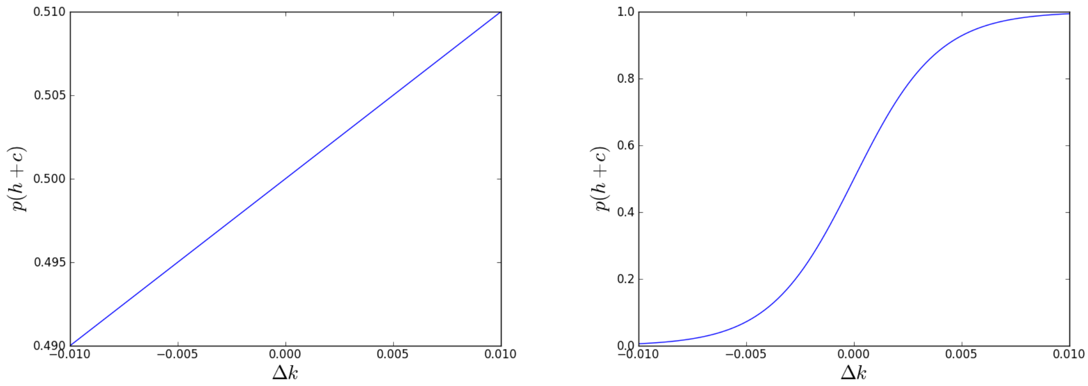
$$p(h') = \frac{1}{1 + \exp^{-\frac{\Delta k}{k_{\text{av}}}}}$$

After d timesteps the success of the new harvest intensity is evaluated by comparing the current harvest to what it was before the harvest intensity changed. If the new harvest is found to be greater or equal then it is kept, otherwise the change in harvest intensity is reversed. If the change is reversed then the direction of future changes will also be reversed. If the new harvest intensity is upheld then the sign of future changes will stay the same. It has been chosen that the initial change will always be negative.

As can be seen in figure 10, the probability distribution depends heavily on k_{av}. When the size of the average harvest is large with respect to the proposed change in harvest the probability of making a jump is roughly 0.5, regardless of the current saturation of the resource pool. This means the number of jumps in a given direction will predominantly be determined by the evaluation of the harvest after d timesteps. Since the average harvest is produced by a weighted average of the current average and the current harvest, the average harvest will always initially be small. Therefore initially the jumps in harvest intensity will be controlled by the probability distribution, and so the saturation of the resource pool.

As we have set $g = 0.5$ for these simulations it makes sense to set $\Delta h = 0.01$, as this gives a reasonable resolution to the harvest intensity, while still allowing the simulation to progress at a reasonable rate.

Since Δh is a constant and $h(0)$ can have any value, it may not be possible to reach exactly the optimal h^*. Instead, the system will oscillate around the optimum. As will be seen, the fluctuations found with this strategy are much larger than Δh, so this is not a problem.



(a) $k_{\text{av}} = \frac{1}{4}$

(b) $k_{\text{av}} = \frac{1}{512}$

Figure 10: The resultant probabilities for a system with a $\Delta h = 0.01$ and two different k_{av}

The size of d is important as it will affect the ability of the player to correctly assess the success of a harvesting intensity. Once a harvest intensity has been chosen the change in the harvest will be dependent on the change in the resource level, $\Delta k = h\,\Delta r$. From equation 1 we can see that:

$$\Delta k > 0 : g\left(1 - r'\right) > h \tag{2a}$$
$$\Delta k = 0 : g\left(1 - r'\right) = h \tag{2b}$$
$$\Delta k < 0 : g\left(1 - r'\right) < h \tag{2c}$$

From this we can see that if there is an equilibrium between the current harvest intensity and the effective growth level, case 2b, then the system will be stable. If the effective growth level is initially greater than the harvest intensity, case 2a, then $\Delta^2 k < 0$ and will eventually stabilise. Finally, if the effective growth level is initially smaller than the harvest intensity, case 2c, then $\Delta^2 k > 0$ until $g\,(1 - r') = h$, or if $g < h$ then until $r = 1$.

We can see the effect of this in figure 11, where for a fixed resource pool growth rate of $g = 0.9$ and for a given initial pool resource quantity, the resource pool stabilises. The resource quantities at stability can be expressed by a function by using equation 1 and noting that the resource quantity will be stable when $\Delta r = 0$, at $\tilde{r}$:

$$g\,\tilde{r}\,(1 - \tilde{r}) - h\,\tilde{r} = 0$$

Which re-arranges to give:

$$\tilde{r} = 1 - \frac{h}{g} \tag{3}$$

as seen in figure 11.

As mentioned in the caption to figure 11, the plot shows the state of the resource after 500 timesteps. This large number of timesteps was necessary as the system tends to stabilise very slowly when both the harvest intensity and initial resource pool quantities are low. It is therefore important to get an estimate of how long the system will take to stabilise. For this we can look at the rate of convergence. Taking

$$r(t) = \tilde{r} + \delta(t)$$

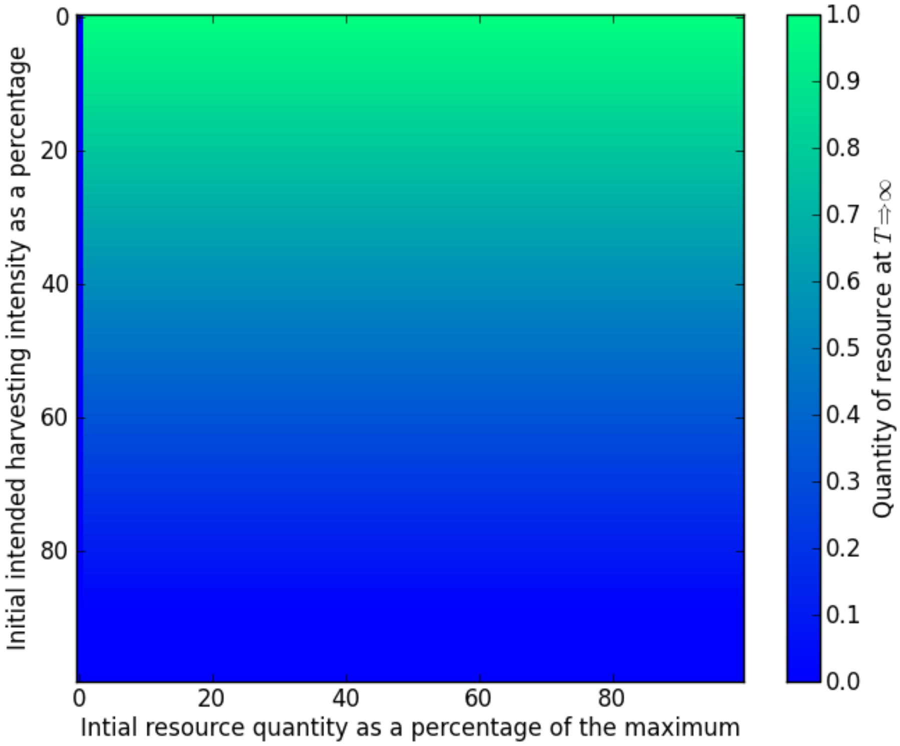


Figure 11: The stable resource levels for a range of harvest intensities and initial pool resource quantities after 500 timesteps. The growth rate for the pools has been kept constant at $g = 0.9$

as the perturbed point of stability and inserting it into equation 1

$$\begin{aligned}\Delta\delta &= g\ (\tilde{r} + \delta)\,(1 - \tilde{r} - \delta) - h\ (\tilde{r}\delta) \\ &= (g\,\tilde{r}\,(1 - \tilde{r}) - h\,\tilde{r}) + g\,\delta - 2g\,\tilde{r}\,\delta - g\,\delta^2 - h\delta \\ &\approx (g - h - 2g\,\tilde{r})\,\delta\end{aligned}$$

$$\delta \approx \exp^{(g-h-2g\,\tilde{r})t} \tag{4}$$

By taking $(g - h - 2g\,\tilde{r})$ as the inverse of the convergence rate we can look at how the convergence varies for different harvest intensities. We can apply the result previously found for $\tilde{r}$ in equation 3

$$\begin{aligned}g - h - 2g\,\tilde{r} &= g - h - 2g\left(1 - \frac{h}{g}\right) \\ &= g - h - 2g + 2h \\ &= h - g\end{aligned} \tag{5}$$

Since the pool resource growth rate used throughout this study is $g = 0.5$ we can say that $\left|\frac{1}{h-g}\right| > 2$. While we are not able to set an absolute upper bound on the number timesteps the system will take to converge, we can set an effective upper bound. When the change between levels in the harvested quantity $\Delta k = \Delta h\,r$ is greater than the difference between the current and stable pool resource quantities, then we can consider that the resource quantity has stabilised. With that in mind we can say using the results in equations 4 and 5 that

$$\begin{aligned} \Delta k &> \delta \\ &> \exp^{(h-g)t} \end{aligned}$$

$$\ln(\Delta k) > (h-g)\,t$$

$$t < \frac{\ln(\Delta h\, r')}{h-g}$$

Therefore, as $h \to g$ from below, $t \to \infty$. We can ignore the cases where $h > g$ as there $r \to 0$.

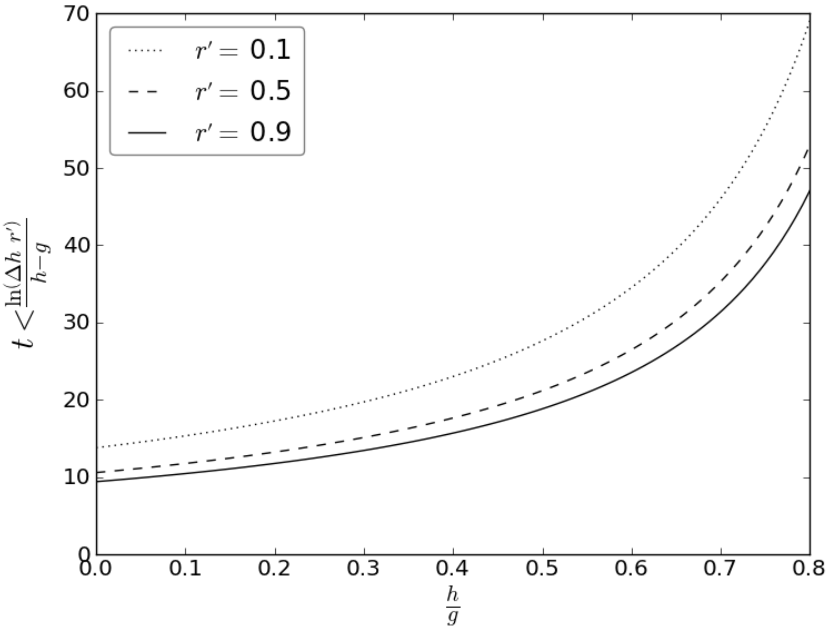


Figure 12: The maximum time to convergence for a system with $\Delta h = 0.01$ for different resource pool saturations.

The maximum number of timesteps to effective convergence for different resource pool levels is shown in figure 12, showing that convergence takes at least a maximum of ten timesteps.

This convergence time is important as we know that the player will get no more information by waiting longer than this number of timesteps before evaluating their current harvest intensity. The system does not have to have converged for the player to get the necessary information, only for the harvest intensity to have changed sufficiently. Experimentally the lowest effective value of d was found to be ten, tallying the results above. The value of $d = 10$ has been used in all simulations unless otherwise specified.

Some results from the probability jump strategy are shown in figures 13 and 14. These show that the system reaches the optimal equilibrium and continues to fluctuate with fluctuations of the order of $5\Delta h$ over about 200 timesteps. Despite the resource pool saturation decreasing as the player's harvest intensity rises above the optimum, the player's

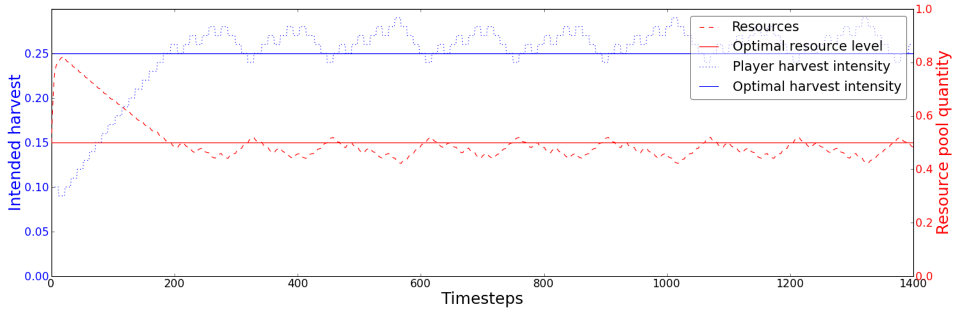


Figure 13: The time evolution of the harvest intensity and pool resource quantity for a player using the probability jumping strategy and a resource pool with a growth rate of 0.5. The discount rate for the average is taken to be 0.1 and the strategy uses the parameters $d = 10$, $\Delta h = 0.01$.

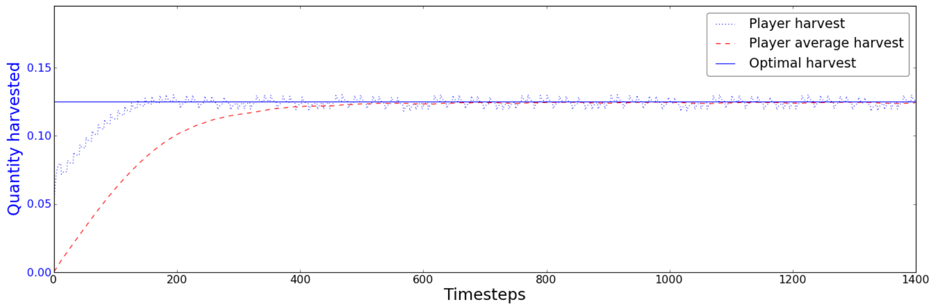


Figure 14: The time evolution of the quantity harvested and the average quantity harvested for a player using the probability jumping strategy and a resource pool with a growth rate of 0.5. The discount rate for the average is taken to be 0.01 and the strategy uses the parameters $d = 10$, $\Delta h = 0.01$.

harvest increases for a number of jumps as the resource pool level is not decreasing fast enough for the effect to be noticed by the strategy.

On average the system is slightly over-harvesting, but the average harvest is still at the equilibrium, as can be seen in figure 14. It is worth noting that the length of these simulations is much longer than for the previous strategies. Whereas before we were seeing stabilisation in around 20 timesteps we now see it in around 300.

It would be tempting to look at values of the discount rate higher than $i = 0.01$, but this results in the average harvest just following the fluctuations of the harvest, rendering using average pointless. An example can be seen in figure 15, where $i = 0.1$. While in this case the impact is debatable, for two-player interactions this becomes more of an issue.

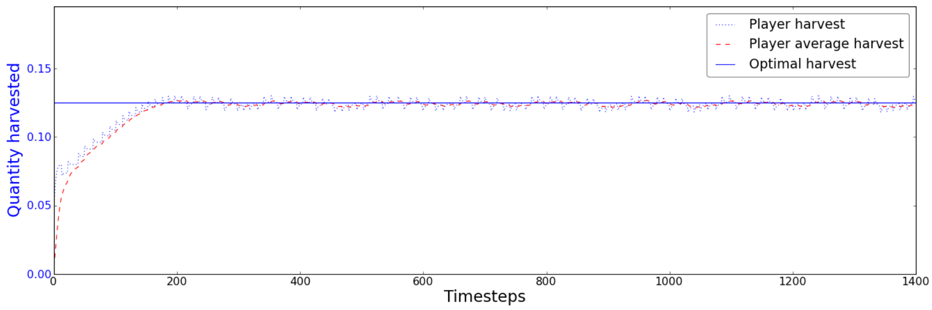


Figure 15: The time evolution of the quantity harvested and the average quantity harvested for a player using the probability jumping strategy and a resource pool with a growth rate of 0.5. The discount rate for the average is taken to be 0.1 and the strategy uses the parameters $d = 10$, $\Delta h = 0.01$.

Two harvesters

We will now look at strategies and simulations where two players are harvesting in the same pool.

The initial harvest intensity will be chosen from a uniform random distribution, ranging between zero and a maximum value.

Two new sets of strategies will be introduced for the two harvester simulations. The first aligns its harvest intensity to that of whichever player has the current best average harvest, *max*, whereas the second are a set of different modifications to the probability jumping strategy that allows it to react to the harvesting changes of the other player.

Max

When there is more than one player harvesting the most simple harvesting strategy is to to adopt the currently most successful harvest intensity. Success in this case is measured as being the player with the highest k_{av} for the given pool.

When playing against itself we can see in figure 16 that the player who is harvesting the least quickly changes its harvest intensity to that of the other player. The resource pool then stabilises as it would do for a constant harvest intensity.

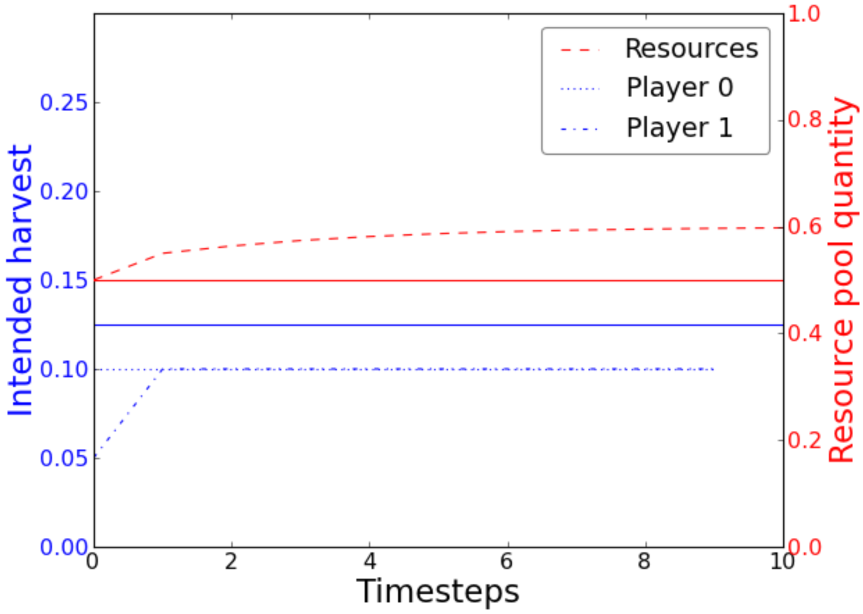


Figure 16: The time evolution of the harvest intensity and pool resource quantity for two players using the max strategy and a resource pool with a growth rate of 0.5.

Reaction to change in resource level

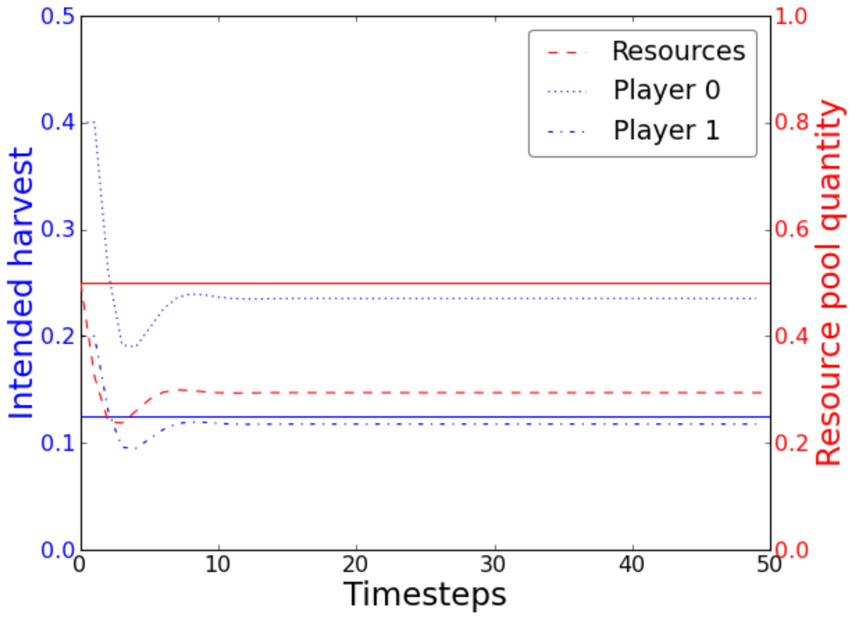


Figure 17: The time evolution of the harvest intensity and pool resource quantity for two players using the reaction strategy and a resource pool with a growth rate of 0.5.

The reaction strategy performs quite similarly on its own as when it is competing with itself, as can be seen in figure 17

Probabilistic jumping

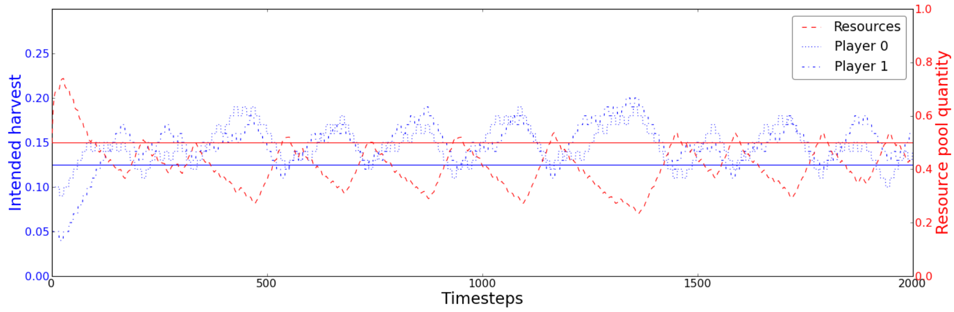


Figure 18: The time evolution of the harvest intensity and pool resource quantity for two players using the probability jump strategy and a resource pool with a growth rate of 0.5.

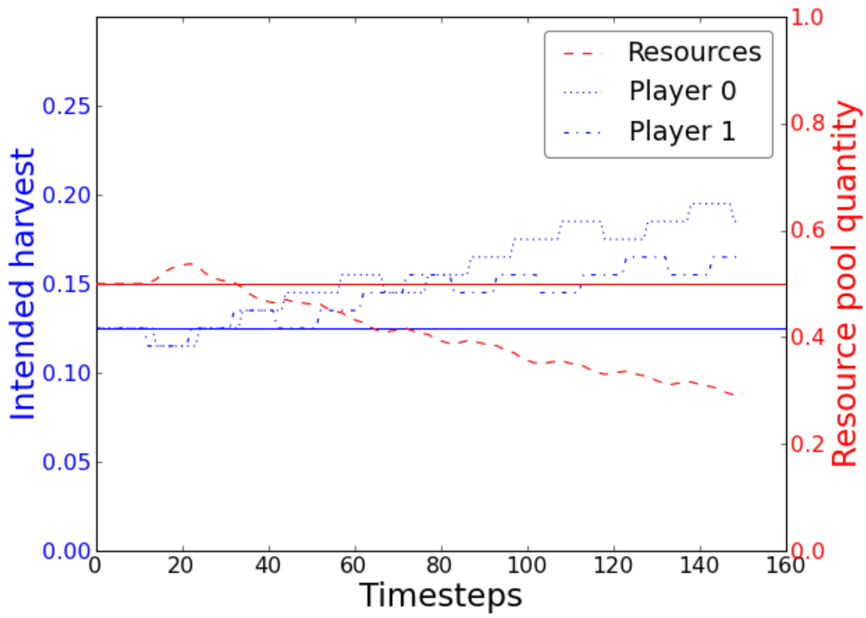


Figure 19: The time evolution of the harvest intensity and pool resource quantity for two players using the probability jump strategy and a resource pool with a growth rate of 0.5.

Comparing figure 18 with figure 13 we can see that when competing against itself the fluctuations become larger and longer, but the stability and ability to reach the optimum are unchanged. This can be explained by the interaction between the player's evaluations of modification to their harvest intensities.

In figure 19 we can see at timestep 30 the resource pool is just above its optimal level and both players are at the optimum harvest intensities and are looking at possible increases in their harvest intensity. Once they have increased their harvest intensity above the optimum the resource pool level begins to decrease below its optimum. The harvests of both players is therefore decreasing and their harvest after d timesteps will be lower than it was at the start. Player 1 reaches this point first and so reverts to the optimal harvest intensity, causing the pool resource to grow slightly. This slight increase in the resource pool leads player 0 to harvest as much when its d timesteps are up as it had before, leading to player 0's harvest intensity to have the opportunity to increase. With player 0's harvest intensity having increased, player 1 finds after d timesteps that its own last decrease in harvest intensity causes it to have smaller harvests than before. It therefore reverses its last change in harvest intensity, thereby increasing it. When player 0 evaluates its latest change in harvest intensity the resource pool is roughly at the level it was d timesteps before, but with its higher harvest intensity its actual harvest is higher, leading it to continue increasing its harvest. This stops when $h(\tau)\, r(\tau) > (h(\tau) + \Delta h)\; r(\tau + d)$ for both players successively.

Probabilistic jumping with punishments

Having found a successful probabilistic jumping algorithm, we attempted to find strategies that were a bit more robust against greedy players. As will be seen, if one player is harvesting above the optimal harvest intensity, then the player using the probabilistic jumping strategy will compensate by harvesting below the optimal harvest intensity, such that the average harvest intensity is at the optimum. The strategies discussed here are inspired by the Tit-for-Tat, *TFT*, strategies that perform so well in two player prisoners dilemma games.

To compare with the TFT strategy, we need to identify when the other player is "defecting", that is to say, when they are current harvesting in a way that is detrimental to the future the state of the resource. For this we cannot just look at when the other player is increasing their harvest intensity, since if the resource pool quantity is above the optimal level it will be best for both players for it to be brought down. A harvest intensity that is constant and above the optimum will equally be detrimental to either the resource pool, or the player.

This set of strategies will identify an "aggressive" act as one where $r(t-2, p) \leq r(t-1, p)$ and $h(t-1, n_{-1}, p) < h(t, n_{-1}, p)$. In other words, when the other player has chosen to increase their harvest intensity despite the resource pool quantity having decreased in the previous timestep.

The evaluation of the player's own current harvest intensity has been chosen to take precedence over the identification of when other player is performing aggressive changes. The identification will therefore only take place during the assessment delay, d.

Three different reactions, or punishments, have been evaluated. The first, *ITFT*, has the player reacting by increasing their harvest intensity by Δh_{pun}. The second strategy, *ETFT*, has the player equalising their harvest intensity with that of the other player. The final strategy, *MTFT*, has the player equalising their harvest intensity with that of the other player only if the other player's harvest intensity is higher.

As the ITFT strategy punishes by increasing the harvest intensity of the player by a fixed amount, the other player may wish to react to this punishment by punishing the other player. This would lead to an unending cycle of reactions. To prevent this the ITFT strategy will not look for aggressive changes by the other player until three timesteps after they changed their harvest intensity. This could potentially be exploited by a strategy that increased its harvest intensity one timestep after the other player changed theirs.

Both ETFT and MTFT will align the the harvest intensities allowed by the Δh discretisation. Since Δh is so small this should have no impact.

All three strategies are vulnerable to a strategy that keeps a high harvest intensity. In all three cases the "punishing aspect would never be activated, leading the opponent to appear as a constant cost each timestep. The Max strategy could potentially fulfil this role.

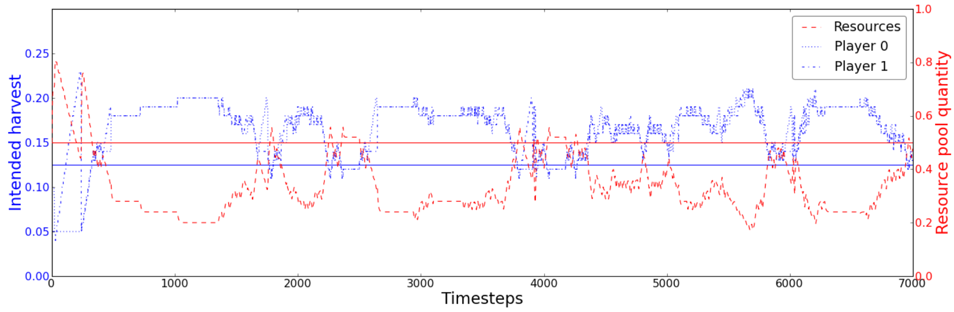


Figure 20: The time evolution of the harvest intensity and pool resource quantity for two players using the ETFT strategy and a resource pool with a growth rate of 0.5.

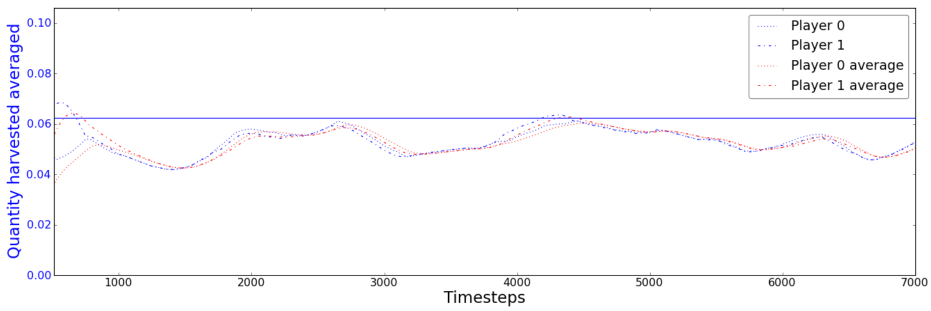


Figure 21: The time evolution of the moving average harvest for two players using the ETFT strategy and a resource pool with a growth rate of 0.5.

The ETFT strategy appears, in figure 20 to reach the optimum equilibrium, much as the probability jump strategy did. However, the fluctuations produced are different, tending to be longer and have greater variation in their structure. The "punishments" that each player produces, identified by the sudden changes in harvest intensity, are as likely to increase the harvest intensity of the punisher as decrease it. This means that half the time the punishment actually acts as a reward for the other player. The harvest reached is not quite optimal, as shown in figure 21.

The ITFT strategy does not appear to be able to produce a stable resource pool, as shown in figure 22. This was tested for a range of punishment values, from 0.005 to 0.1, all producing the same result.

The MTFT strategy was designed as an improvement over the ETFT strategy. While it does not have the sudden drops in harvest intensity seen with the ETFT strategy, it does have far larger and longer deviations from the optimum, causing the average harvest for MTFT to be lower than of ETFT. This can be seen in figure 23 and 24. Occasionally, the strategy produces states where the level of the resource pool collapses to nothing, as shown in figure 25.

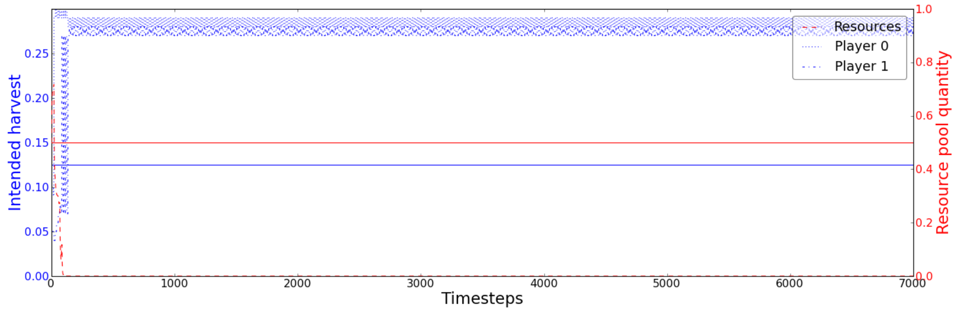


Figure 22: The time evolution of the harvest intensity and pool resource quantity for two players using the ITFT strategy and a resource pool with a growth rate of 0.5. The punishment value in this case is an increase of the harvest intensity by 0.01, equivalent to the standard change in harvest intensity.

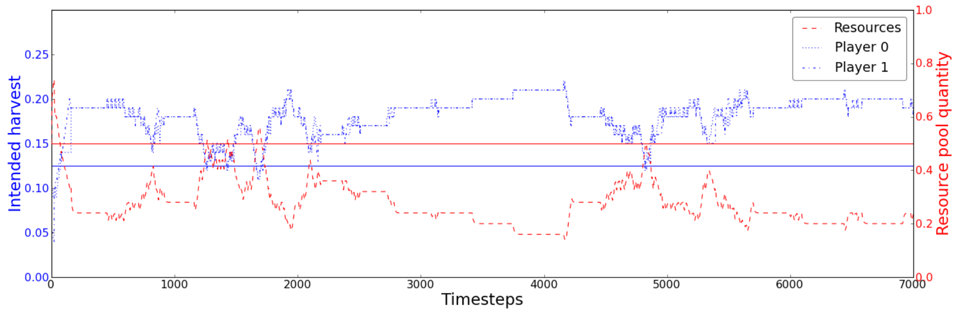


Figure 23: The time evolution of the harvest intensity and pool resource quantity for two players using the MTFT strategy and a resource pool with a growth rate of 0.5.

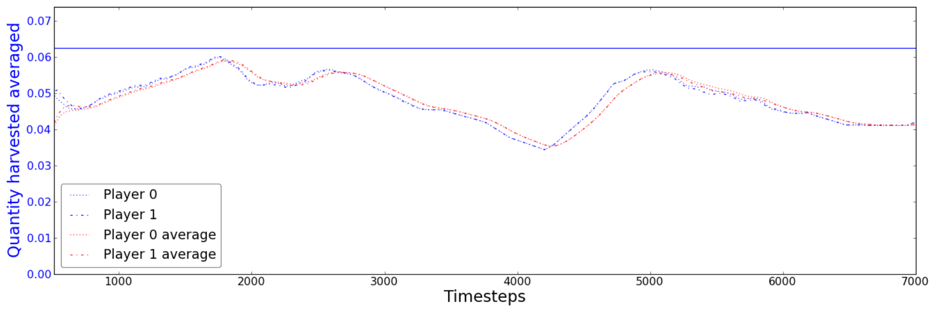


Figure 24: The time evolution of the moving average harvest for two players using the MTFT strategy and a resource pool with a growth rate of 0.5.

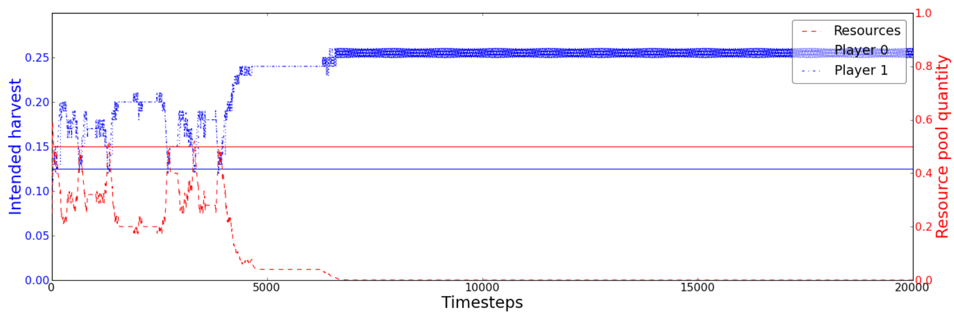


Figure 25: The time evolution of the harvest intensity and pool resource quantity for two players using the MTFT strategy and a resource pool with a growth rate of 0.5.

Cross interactions between the different strategies

With two players and one pool we can look at how successful each strategy is against the other. Two simulations were performed for each pair of strategies, allowing each strategy to start with the higher initial harvest intensity. The two initial harvest intensities that will be used are 0.1 and 0.05.

The results of the simulations are represented in table 3. It can clearly be seen that the ITFT algorithm never leads to any anything other than an empty resource pool. Equally, the MTFT strategy tends to lead the players to over harvest. It performs best against the probability jump strategy, where the system is not only near the optimum, but also the MTFT player harvests more than the probability jump player. The max strategy tends to either keep to the same harvest intensity, or match harvest intensity of the other player. One exception is if the player with the probability jump strategy starts with the higher harvest intensity, which produces a sawtooth once it stabilised. In that case the player using the max strategy kept the highest harvest intensity recorded by either player. The reaction strategy tends to be completely dominated by the other strategies while the resource pool does not go to zero.

Strategy 1	Strategy 2					
	Max	Reaction	Prob Jump	ETFT	ITFT	MTFT
Max	See above	-	-	-	-	-
Reaction	Max matches reaction or stays the same	Identical to one player, but with half the harvest intensity per person	-	-	-	-
Prob Jump	The resource pool empties if prob jump starts with a higher harvest intensity, otherwise the system is over harvested.	Over harvesting	Almost optimal harvest using a higher than optimal harvest intensity	-	-	-
ETFT	The resource pool empties if ETFT starts with a higher harvest intensity, otherwise the system is over harvested.	Optimal harvest with ETFT harvesting the most	Almost optimal harvest using a higher than optimal harvest intensity	Almost optimal harvest using a higher than optimal harvest intensity	-	-
ITFT	Empty resource pool	Empty resource pool	Empty resource pool	Empty resource pool	Empty resource pool	-
MTFT	Significant over harvest	Over harvest	Close to optimal harvest with a higher than optimal harvest intensity. MTFT harvests more.	Over harvest	Empty resource pool	Unstable. Overharvested when not non-zero. Both players match each other.

Table 3: The results from getting any two of the strategies to compete against each other.

Many harvesters

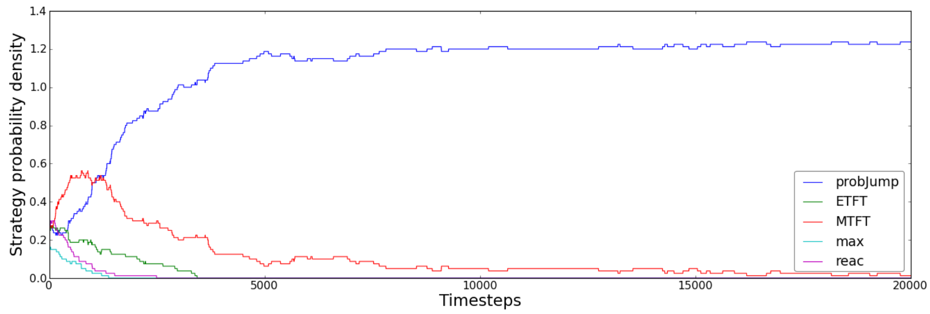


Figure 26: The time evolution of the probability density distribution of strategies on the network.

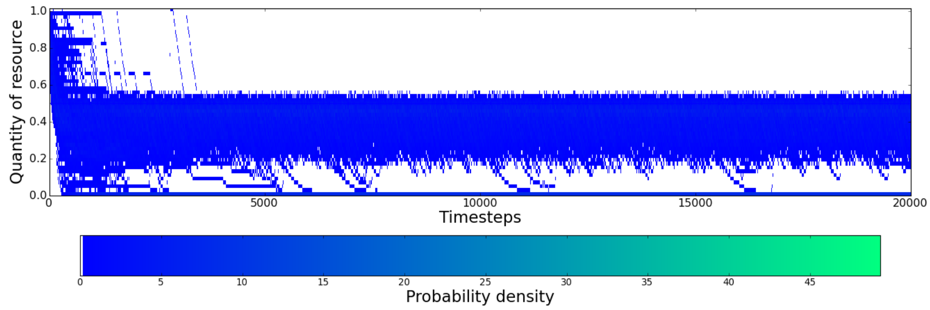


Figure 27: The time evolution of the probability density distribution of pool resource on the network.

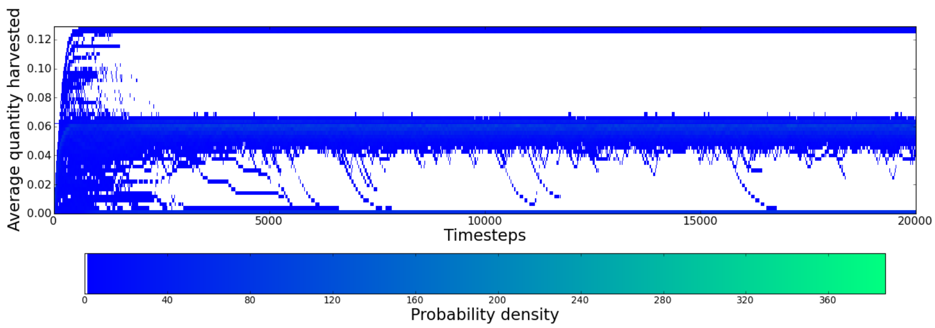


Figure 28: The time evolution of the probability density distribution of k_{av} on the network. For one player the optimum is 0.125 and for two players it would be 0.0625

Having now seen how the strategy interact with each other we can now look at how the distribution of strategies changes on the lattice of pools. The strategy parameters are kept as before, with only one set of parameters used per strategy. All strategies are considered to be initially equally possible and the same initialisation procedures are used as in previous simulations. The grid that will be used will be a 10×10 grid of players, connected in a toroidal lattice. This results in a simulation with 100 players harvesting from 200 pools.

The parameters for the choice of how frequently to re-evaluate the strategy chosen by a player were chosen with the aim of having a change roughly 1% of the time. This would mean that on average there would be an evaluation of the current strategy every 100 timesteps. For a mean, w, of 100 this can be roughly achieved by evaluating when a value is generated above a threshold of $w_{\text{thresh}} = 130$.

Based on the pairwise simulations ITFT has been neglected from these simulations.

When all strategies are made available the system consistently favours the probability jumping strategy, as shown in figure 26. While a consensus is reached on the strategy to be used by all players it does not necessarily lead to the players recieving the same harvest, as figure 28 shows. Here it can be seen that in some pools one player has managed to dominate and reach the single player optimal harvest level. We can also see with figure 27 that not all pools reach the optimal resource saturation, with some being virtually empty.

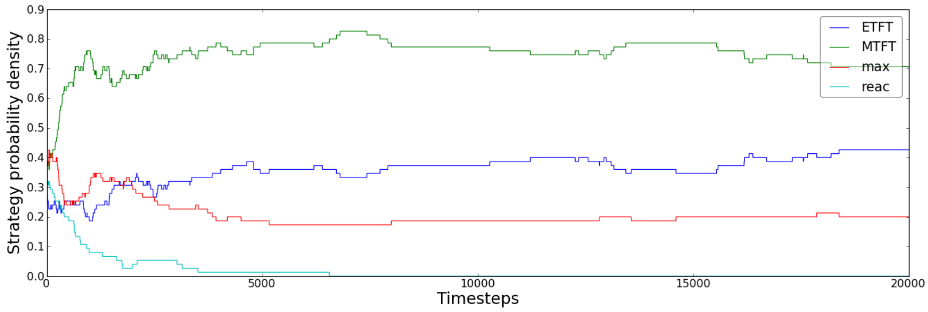


Figure 29: The time evolution of the probability density distribution of strategies on the network.

Having found a completely dominating strategy, simulations were run to see what happened if the dominant strategy was removed. As can be seen in figure 29, the removal of the probability jumping strategy leads to ETFT dominating. This is consistent with the results before, as we have seen that MTFT, while being capable of producing the highest harvests, does not tend to the optimum when playing against itself. The worse harvesting performance of ETFT can be seen by the far greater persistence of MTFT in the system.

Conclusions

This thesis studied strategies for finding the optimal action to play in a dynamic game, where the payoffs depended on the previous actions of all players as well as the reaction of a *nature* player in the form of the natural growth of the resource. When the strategies were played against nature the initial simple strategies were found to consistently identify an equilibrium, but not the optimum. A more complicated strategy, probability jump, was explored and found for some parameter sets to consistently find the optimal system state regardless of the initial state of the system.

A second player was then introduced. The various proposed strategies were compared for their capacity to find the optimal payoffs when playing against themselves and each other. Some variations on the probability jump strategy were studied that allowed for reactions to apparent aggressive behaviour of the other player. It was found that most combinations of strategies led to a non-optimal equilibrium, and in some cases the players ended up depleting the resource pool.

Finally, the games were linked together in a grid, with the games on the edges and one player per node. The players were allowed to change which strategy they used if one of their neighbours was harvesting more on average. The strategies were seen to compete against each other and, with the basic probability jumping strategy dominating whenever it was used. This is not due to its ability to achieve the highest equilibrium harvest, but because it is the most likely to reach the equilibrium harvest when competing against any other strategy.

It was initially hoped that simple strategies could be used to find the optimum. However, no such strategies were found. Given the complexity of strategies, such as probabilistic jumping, it would appear that machine learning algorithms would be interesting to study, as they provide a more general strategy-generating framework, while being of the same level of complexity. The difficulty in using such methods will be in correctly framing the inputs and outputs to these strategies.

Bibliography

[1] E Akiyama. 'Dynamical systems game theory II A new approach to the problem of the social dilemma'. In: *Physica D: Nonlinear Phenomena* 167.1-2 (July 2002), pp. 36–71. ISSN: 01672789. DOI: 10.1016/S0167-2789(02)00402-5.

[2] E. Akiyama and K. Kaneko. 'Dynamical systems game theory and dynamics of games'. In: *Physica D: Nonlinear Phenomena* 147.34 (2000), pp. 221 –258. ISSN: 0167-2789. DOI: 10.1016/S0167-2789(00)00157-3.

[3] R Axelrod and W. Hamilton. 'The evolution of cooperation'. In: *Science* 211.4489 (Mar. 1981), pp. 1390–1396. ISSN: 0036-8075. DOI: 10.1126/science.7466396.

[4] Jordi Delgado. 'Emergence of social conventions in complex networks'. In: *Artificial Intelligence* 141.1-2 (Oct. 2002), pp. 171–185. ISSN: 00043702. DOI: 10.1016/S0004-3702(02)00262-X.

[5] Anders Eriksson and Kristian Lindgren. 'Cooperation driven by mutations in multi-person Prisoner's Dilemma.' In: *Journal of theoretical biology* 232.3 (Feb. 2005), pp. 399–409. ISSN: 0022-5193. DOI: 10.1016/j.jtbi.2004.08.020.

[6] J. Gómez-Gardeñes et al. 'Dynamical Organization of Cooperation in Complex Topologies'. In: *Physical Review Letters* 98.10 (Mar. 2007), pp. 1–4. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.98.108103.

[7] Garrett Hardin. 'The tragedy of the commons.' In: *Science (New York, N.Y.)* 162.3859 (Dec. 1968), pp. 1243–8. ISSN: 0036-8075. DOI: 10.1126/science.162.3859.1243.

[8] Kristian Lindgren. 'Evolutionary phenomena in simple dynamics'. In: *Artificial life II* 10 (1991), pp. 295–312.

[9] Kristian Lindgren and Mats G. Nordahl. 'Evolutionary dynamics of spatial games'. In: *Physica D: Nonlinear Phenomena* 75.1-3 (Aug. 1994), pp. 292–309. ISSN: 01672789. DOI: 10.1016/0167-2789(94)90289-5.

[10] J.M. Pujol et al. 'The role of clustering on the emergence of efficient social conventions'. In: *Proceedings of the 19th international joint conference on Artificial intelligence*. Morgan Kaufmann Publishers Inc., 2005, pp. 965–970.

[11] Martín G Zimmermann and Víctor M Eguíluz. 'Cooperation, social networks, and the emergence of leadership in a prisoner's dilemma with adaptive local interactions.' In: *Physical review. E, Statistical, nonlinear, and soft matter physics* 72.5 Pt 2 (Nov. 2005), p. 056118. ISSN: 1539-3755.