Computation of mixed-strategy equilibria in the repeated prisoner's dilemma

Juho Timonen

School of Science

Bachelor's thesis

Espoo December 16, 2015

Thesis supervisor:

Prof. Harri Ehtamo

Thesis advisor:

D. Sc. (Tech.) Kimmo Berg

Aalto University
School of Science

Aalto University
School of Science

Aalto University, P.O. BOX 11000, 00076 AALTO
www.aalto.fi

Abstract of bachelor's thesis

Author Juho Timonen	
Title of thesis Computation of mixed-strategy equilibria in the repeated prisoner's dilemma	
Degree programme Degree Programme in Science and Technology	
Major Mathematics and Systems Analysis	**Code of major** SCI3029
Supervisor Prof. Harri Ehtamo	
Thesis advisor D. Sc. (Tech.) Kimmo Berg	
Date 16.12.2015	**Number of pages** iii+ 23 **Language** english

Abstract

Repeated games model long-term relationships and competition. On each round of the game, players receive a payoff depending on their own and other players' actions. They discount future payoffs to present value with a discount factor. Complexity of such games increases remarkably when players are allowed to randomize between their pure actions. A player's strategy assigns an action for every possible history of play and a strategy profile consists of all players' strategies. A strategy profile is a subgame-perfect equilibrium if no player has a profitable deviation after any history.

Very little is known about the set of subgame-perfect equilibrium payoffs for repeated mixed-strategy games. Finding this set can be very complex even for simple two-player games. This study aims to develop an algorithm for finding this equilibrium set in prisoner's dilemma. The algorithm is based on a useful fixed-point characterization of the set, presented by Abreu, Pierce and Stachetti. It finds the equilibrium payoff set by first assuming a certain continuation payoff set, and then iteratively calculating the set of total payoffs. If the set of total payoffs converges, the algorithm has finished its task. The algorithm will only find the equilibrium payoff set and does not provide us with information about the strategies that produce these payoffs.

Implementing the algorithm turned out to be complex for a prisoner's dilemma with an arbitrary discount factor. A classification of 2 x 2 games by Borm was utilized for systematic treatment of games where expected continuation payoffs are included in the stage game payoffs. Doing this calculation efficiently was the most difficult task in the algorithm's implementation.

The set of subgame-perfect equilibria was successfully calculated for prisoner's dilemmas with big discount factors. Some results were also obtained for small discount factors, but the problem still needs more thorough treatment. This study focused on prisoner's dilemma to simplify the computational task. However, with small discount factors, implementation of the algorithm brought ideas that could be expanded and thus solve the equilibrium sets for other 2 x 2 games, too.

Keywords Game theory, repeated games, mixed strategy, subgame-perfect equilibrium, payoff set, prisoner's dilemma

Contents

1 Introduction

Is ”live every day like it is your last” a good advice? Why do people not behave according to it? That is because they know that their life is a series of repeated interactions. They do not spend their paycheck on the first day of the month because they know that it results in a very unpleasant future. Repeated games, especially infinitely repeated games, have a very different nature than games that are only played once.

In an infinitely repeated game, players play the same stage game infinitely many times. If the game is a pure-strategy game, players choose a certain action on every round. In a mixed-strategy game, players are allowed to randomize over their pure actions. Strategy profile of the game defines how players will act after any history. On each round, each player receives a payoff that depends on all player’s actions on that round. Players discount their future payoffs to present value with factor δ.

In 1965, Selten introduced concept of a subgame-perfect equilibrium [11]. It is a strategy profile from which no player has incentive to deviate after any history. Finding strategy profiles that are subgame-perfect equilibria is a very nontrivial problem. However, Blackwell’s important result from dynamic programming [6] guarantees that only the profitability of one-shot deviations has to be checked. Berg and Kitti have used elementary subpaths to find equilibrium strategies among pure strategies [4].

The set of subgame-perfect equilibria in repeated mixed-strategy games has remained a mystery. This is propably because finding Nash equilibria is difficult enough for games with pure strategies. Abreu, Pearce and Stacchetti presented a fixed-point characterization of the set for repeated games with imperfect monitoring [2],[3]. Berg and Schoenmakers studied the characterization for mixed strategies [5]. It can be seen as a special case of imperfect monitoring. See Chapter 9 in [8].

Aim of this study is to implement an iterative algorithm that calculates the equilibrium payoff set for a repeated prisoner’s dilemma with mixed strategies. The key idea is to use the fixed-point characterization of Abreu, Pierce and Stachetti. We study games in which public correlation is not allowed and players observe only the realized pure actions and not the randomization probabilities that the other players use. Only the equilibrium payoffs are of interest. A fixed-point algorithm does not tell us what kind of strategies are needed to get a certain equilibrium payoff. Obtained results are compared to pure-strategy games.

Prisoner’s dilemma is one of the most famous concepts in game theory. Focusing on it, computational challenges of the problem are more realistic to overcome as there are only two player’s and two pure actions. Although the study focuses on prisoner’s dilemma, new information is obtained with ideas to widen the scope of research in the future.

2 Theoretical background

This section introduces the basic concepts that are used to describe games. The difference between single shot games and repeated games is clarified with an example. This study focuses on 2×2 games and therefore a special notation for such games is presented. The key theorem needed to understand the presented algorithm, fixed-point characterization of subgame-perfect equilibria, is also explained.

2.1 The stage game

2.1.1 Basic notation and concepts

A game that is played only once is called a *stage game*. The set of players is denoted by $N = \left\{1, ..., n\right\}$. Each player i has a set of possible *pure actions* A_i and the set of possible *pure-action profiles* is $A = \times_{i \in N} A_i$. This paper studies games in which players are allowed to randomize over their pure actions. Player i's set of possible *mixed-action profiles* Q_i is a set of probability distributions over A_i. Also, we denote the cartesian product of those by $Q = \times_{i \in N} Q_i$. A pure action $a_i \in A_i$ is actually a mixed action $q_i \in Q_i$, a distribution that assigns probability 1 on a and 0 on the other pure actions.

The set of pure-action profiles that can be realized when mixed-action profile q is played is called the *support* of q and denoted $Supp(q) = \times_{i \in N} Supp(q_i)$, where

$$Supp(q_i) = \left\{a_i \in A_i | q_i(a_i) > 0\right\}. \tag{1}$$

Moreover, for each $a \in Supp(q)$, the probability for a realizing is given by

$$\pi_q(a) = \prod_{j \in N} q_j(a_j). \tag{2}$$

Now we can define a *payoff* function $u : Q \to \mathbb{R}^n$. When a mixed-action profile q is played in a stage game, player i receives payoff

$$u_i(q) = \sum_{a \in A} u_i(a) \pi_q(a). \tag{3}$$

Player i's *minmax* payoff

$$\underline{v}_i = \min_{q_{-i} \in Q_{-i}} \max_{q_i \in Q_i} u_i(q_i, q_{-i}) \tag{4}$$

is the lowest payoff he can be forced to. Here q_{-i} is player i's opponents' action profile from the set $Q_{-i} = \times_{j \in N, j \neq i} Q_j$.

The set of *individually rational* payoffs $\mathcal{F}^*$ can only consist of points where no player gets a payoff less than his minmax [8].

$$\mathcal{F}^\star = \{v \in \mathcal{F}^\dagger \mid v_i \geq \underline{v}_i\}, \tag{5}$$

where $\mathcal{F}^\dagger$ is the convex hull of stage-game payoff points.

2.1.2 2 × 2 bimatrix games

This thesis focuses on games with two players both having two possible pure actions. In such a game, $N = 2$ and $|A_i| = 2$, $i \in \{1,2\}$. Let us denote the actions $A_1 = \{T, B\}$ and $A_2 = \{L, R\}$. In bimatrix representation, this means that player 1 chooses between rows and player 2 between columns. Throughout this thesis, we use four letters

- $a = \{T, L\}$
- $b = \{T, R\}$
- $c = \{B, L\}$
- $d = \{B, R\}$

to denote the four possible pure-action profiles. Using this notation, $A = A_1 \times A_2 = \{a, b, c, d\}$. In a bimatrix, the four cells represent the four pure-action profiles. Every cell has a payoff vector (u_1, u_2) representing the payoffs from that profile.

Example 2.1. Table 1 defines a stage game PD1, in which $u(a) = (3,3)$, $u(b) = (0,4)$, $u(c) = (4,0)$ and $u(d) = (1,1)$.

	L	R
T	$3,3$	$0,4$
B	$4,0$	$1,1$

Table 1: Game PD1.

2.1.3 Nash Equilibrium

We denote the mixed-action profile of all players excluding i by q_{-i}. It is an element of the set $Q_{-i} = \times_{j \in N, j \neq i} Q_j$.

Definition 1. Action profile $q \in Q$ is a *Nash equilibrium* of the stage game if for all $q_i' \in Q_i$ and $i \in N$,

$$u_i(q) \geq u_i(q_i', q_{-i}). \tag{6}$$

According to Nash's existence theorem [9], all stage games have at least one mixed- or pure-action Nash equilibrium.

Example 2.2. In game PD1, d is a Nash equilibrium.

2.2 Repeated games

2.2.1 Play in a repeated game

In an infinitely repeated game, players play the same stage game again and again infinitely many times. The game is assumed to be perfectly monitored, meaning that the players observe every realized pure-action profile that is played and remember all earlier moves. However, players can only observe the realized actions and not the actual probability distributions that others possibly use. Thus, to make randomization possible, a mixed action q_i must produce exactly the same expected payoff for every $a_j \in Supp(q_j)$, for all $j \in N, j \neq i$.

Let H^t be the set of all possible histories after t rounds of play, with a typical element $h^t = \left\{q^1, ..., q^t\right\}$, where q^t is the action profile played in the tth round of play. Moreover, $H = \bigcup_{t=0}^{\infty} H^t$ is the set of all possible histories.

In a repeated game, each player i has a *strategy* that is a function $\sigma_i : H \to Q_i$, telling the player what to do after any history. The players' strategies form a strategy profile $\sigma = \left\{\sigma_1, ..., \sigma_n\right\}$. Starting from initial history $h^0 = \emptyset$, σ induces a path of play for the game. On round t, strategy profile σ yields a payoff $u_i^t = u_i(q^t(\sigma))$ for player i. During the game, the player receives a flow of payoffs $\left\{u_i^1, ..., u_i^t, ...\right\}$ and discounts them with factor $\delta_i \in (0, 1)$. The expected total normalized payoff from the game for player i is

$$U_i(\sigma) = \mathbb{E}\Big[(1-\delta)\sum_{t=0}^{\infty} \delta^t u_i(q^t(\sigma))\Big]. \tag{7}$$

2.2.2 Subgame-perfect equilibrium

In Section 2.1.3 we introduced the concept of Nash equilibrium for a single-shot game and now we extend it to repeated games.

Definition 2. A strategy profile σ is a Nash equilibrium of the repeated game if for all σ_i' and $i \in N$,

$$U_i(\sigma) \geq U_i(\sigma_i', \sigma_{-i}), \tag{8}$$

where σ_{-i} is player i's opponents' strategy profile.

However, a Nash equilibrium does not demand optimal behaviour in all possible situations. We define $\sigma|h$ to be the strategy profile after history h.

Definition 3. A strategy profile σ is a *subgame-perfect equilibrium (SPE)* of the repeated game game if for all σ_i', $h \in H$ and $i \in N$,

$$U_i(\sigma|h) \geq U_i(\sigma_i', \sigma_{-i}|h). \tag{9}$$

Strategy profiles that are SPE do not allow irrational behaviour after histories during which someone has deviated from the equilibrium path. In the course of a SPE, we can interpret the game in the following way: On each round, players play a stage game in which payoffs are formed by taking into account the expected continuation payoffs, i.e., calculating what the total payoff from the game will be if a certain profile is played on that round.

Example 2.3. If game PD1 is infinitely repeated and players are patient enough, the following strategy profile is a subgame-perfect equilibrium:

- Play T/L on every round unless either player has deviated
- If either player has deviated, play B/R infinitely.

If players are patient enough, i.e., δ is not very small, no player wants to deviate because it will result in a smaller total payoff. This strategy will result in a payoff

$$U = (1-\delta) \cdot 3 + \delta \cdot 3 = 3 \tag{10}$$

for both players. This is better than what was possible to reach in the same single-shot game.

2.2.3 Fixed-point characterization

A useful characterization for finding equilibria of repeated games was developed by Abreu, Pearce and Stachetti in [2],[3]. Let the most profitable deviation outside $Supp(q_i)$ be the pure action that yields

$$d_i(q) = \max_{a_i' \in A_i \setminus Supp(q_i)} u_i(a_i', q_{-i}) \tag{11}$$

and the punishment payoff of player i in any compact set W be

$$p_i(W) = \min \Big\{ w_i, w \in W \Big\}. \tag{12}$$

Suppose that $a \in Supp(q)$ induces a continuation payoff $x(a)$. Then a mixed-action profile q has an expected continuation payoff

$$w = \sum_{a \in Supp(q)} x(a) \pi_q(a). \tag{13}$$

Definition 4. [5] A pair (q, w), where $w \in W$, is admissible with respect to W, if for all $i \in N$

$$(1-\delta_i)u_i(a) + \delta_i w_i \geq (1-\delta_i)d_i(q) + \delta_i p_i(W). \tag{14}$$

These are the *incentive compatibility (IC)* conditions.

Consider a stage game where payoffs are formed by combining the continuation payoffs with the payoffs of the original stage game. Thus action profile a will give the payoff vector

$$\mu(a) = (I-\delta)u(a) + \delta x(a). \tag{15}$$

We denote the set of all equilibrium payoffs in this stage game by $M(x)$.

Lemma 2.1. [5] *The set of subgame-perfect equilibrium payoffs, denoted by V, is the largest fixed point of the following mapping B:*

$$W = B(W) = \bigcup_{x(a)\in W} M(x), \tag{16}$$

where (q, w) is admissible with respect to W, $w = \sum_{a\in Supp(q)} x(a)\pi_q(a)$ and q is a Nash equilibrium in a stage game defined by the continuation payoffs $x \in W^{|A|}$.

3 Research problem and methods

Very little is known about the set of equilibrium payoffs in repeated mixed-strategy games. The aim of this study is to develop methods to find out what this set will look like when given a 2×2 bimatrix game and the players' discount factors. In this section, algorithm for reaching the goal is presented first in abstract level and then more specifically for prisoner's dilemma.

3.1 Algorithm

Let V be the set of subgame-perfect equilibrium payoffs. Idea of the algorithm for finding V in a 2×2 game is the following:

1. Choose $W_0 = [m_1, M_1] \times [m_2, M_2]$, where
    - m_i is the minmax payoff for player i
    - M_i is the maximal stage game payoff for player i

    This should be a sufficiently large rectangle to surely contain V.
2. $W \leftarrow W_0$.
3. Suppose that the set W can be used as continuation payoffs after any pure-action profile played first.
4. Calculate four affine transformations

$$X(\alpha) = (1 - \delta)u(a) + \delta W \tag{17}$$

    for $\alpha = \{a, b, c, d\}$. This means that for all four pure actions played on the first round, you calculate what the set of possible total payoffs is, as W is assumed to be the set of all possible continuation payoffs.
5. Calcuate $B(W)$, which is now the union of the sets of equilibrium payoffs in all possible stage games where payoffs $\mu(\alpha)$ can be chosen freely from $X(\alpha)$. Let us denote this operation by

$$\mathcal{E}(X) = \bigcup_{\mu(\alpha) \in X(\alpha)} M(\mu), \tag{18}$$

    where $X = \{X(a), X(b), X(c), X(d)\}$ and $M(\mu)$ is the set of equilibria for a game with payoffs $\mu(\alpha)$, $\alpha = \{a, b, c, d\}$.
6. If $B(W) \neq W$, $W \leftarrow B(W)$ and go to Step 3. Else $B(W) = V$.

An example of the first iteration of the algorithm is presented in Figure 1. Choosing W_0 to be a rectangle will cause the set W to always consist of only X-Y polygons,

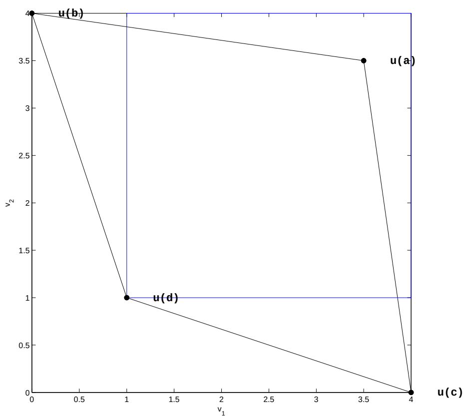

Figure 1: Example of mapping B. The blue square represents W. Green squares are four affine transformations of W and denoted $X(a)$, $X(b)$, $X(c)$ and $X(d)$. Here the discount factor $\delta_i = 0.24$ for both i. The red polygon represents $B(W)$.

which makes the task of calculating $B(W)$ easier.

Definition 5. [10] An $X - Y$ *polygon* is a simple polygon with only horizontal and vertical edges.

The hardest part of the algorithm is without a doubt Step 5. In order to be able to perform the equilibrium calculations, we first need to divide sets $X(\alpha)$ into polygons for which it can be done easier. The division consists not only of dividing the sets into parts with a more convenient shape but also dividing them into parts that are oriented more conveniently with respect to each other.

5.1 If W is not a connected set and consists of K simple polygons, we need to first divide all $X(\alpha)$ into simple polygons $X(\alpha)_j$, where $j \in \{1, 2, ..., K\}$.

5.2 Let us now assume that we have four separate simple X-Y polygons P_A, P_B, P_C and P_D oriented like in Figure 2. Let us then define points

$$p_B = \left(\min_{x \in P_B} x, \max_{y \in P_B} y\right) \tag{19}$$

and

$$p_C = \left(\max_{x \in P_C} x, \min_{y \in P_C} y\right). \tag{20}$$

X-Y oriented lines through both points divide P_A and P_D into maximum of four parts each, leaving the remaining parts oriented more conveniently. Figure 2 illustrates an example.

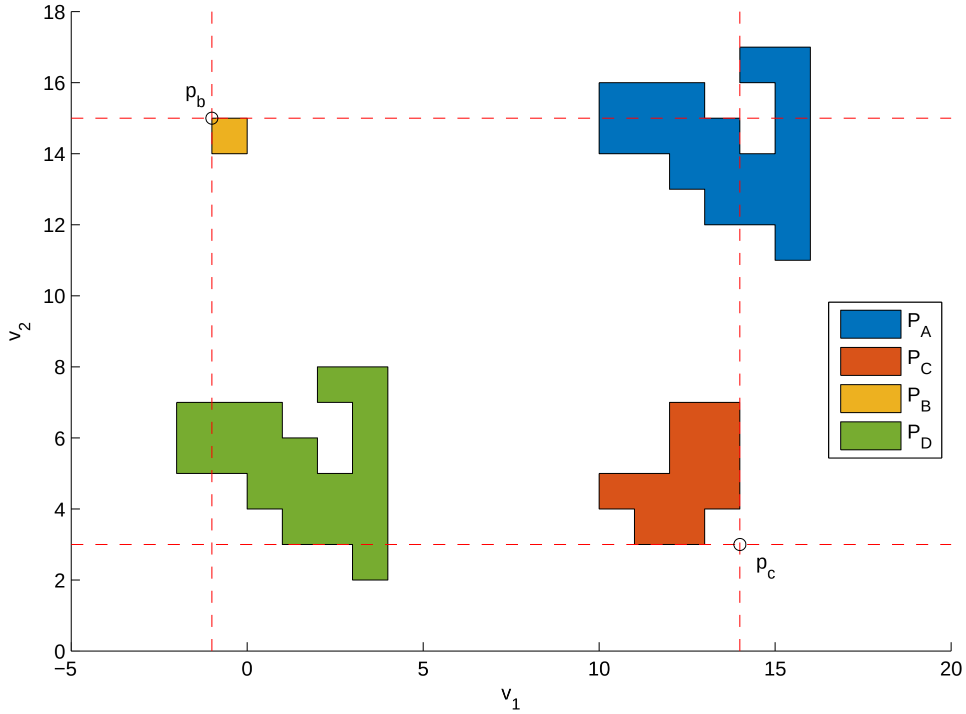

Figure 2: Example of four simple polygons. X-Y oriented lines through points p_B and p_C divide P_A and P_D into maximum of four parts.

5.3 Finally to simplify the calculation, we need to split the remaining polygons into X-Y convex parts.

Definition 6. [10] An $X-Y$ *convex* polygon is an X-Y polygon, such that within the polygon, for any two points lying on the same vertical or horizontal line, the straight line segment between them lies completely inside the polygon.

Let us now assume that we have divided all four $X(\alpha)$ into groups $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$ and $\mathcal{D}$. Now we need to perform the operation $\mathcal{E}(P)$ for all combinations $P = \{P_A, P_B, P_C, P_D\}$, where $P_A \in \mathcal{A}$, $P_B \in \mathcal{B}$, $P_C \in \mathcal{C}$ and $P_D \in \mathcal{D}$.

One way to calculate operation $\mathcal{E}$ is to handle possible stage games by dividing them into classes. A useful classification in presented by Borm in [7] where 2×2games are divided into 15 classes.

3.2 Prisoner's dilemma with small discount factors

Let us now consider prisoner's dilemmas with small discount factors. An example of four polygons for which $\mathcal{E}$ has to be calculated is presented in Figure 3. No matter which four polygons we choose, the red dashed lines through p_B and p_C never intersect with P_D. Also, with a sufficiently small discount factor, P_D never has the same y-coordinate with P_C or the same x-coordinate with P_B. This rules out many difficult situations.

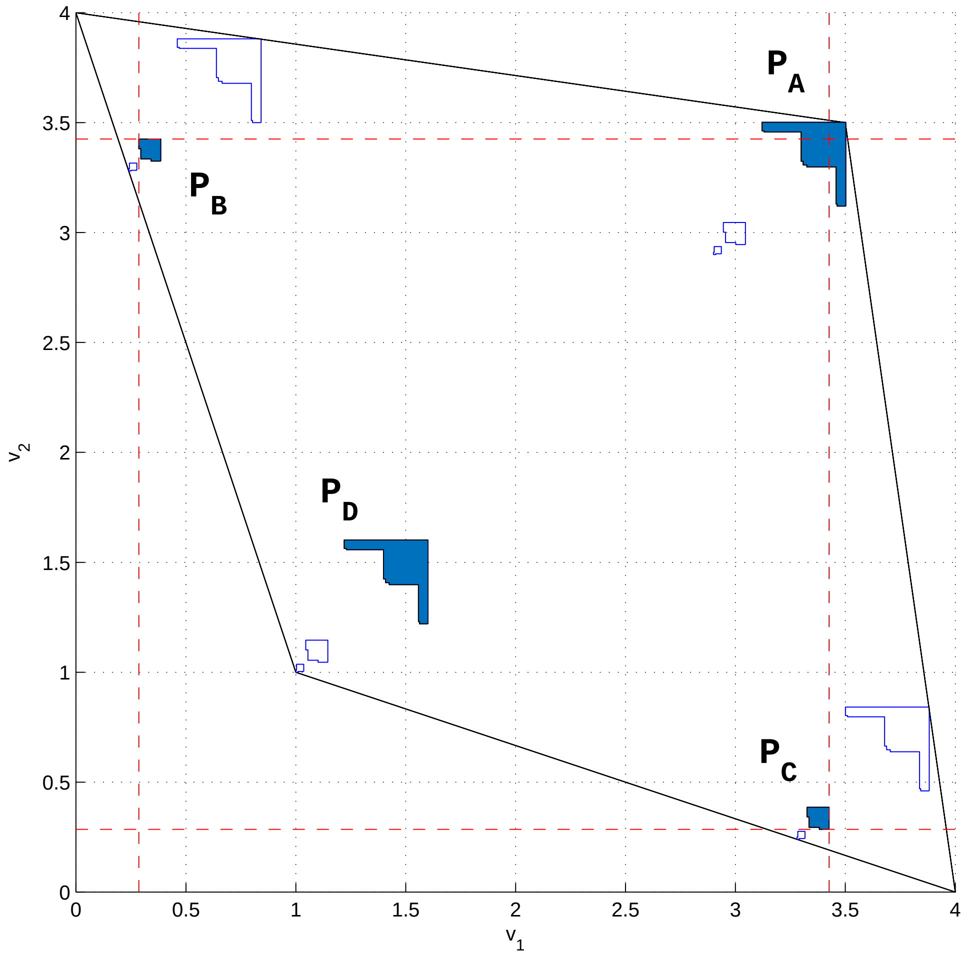

Figure 3: In prisoner's dilemma with small discount factors, only P_A needs to be split. Here $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$ and $\mathcal{D}$ are the four groups of polygons close to each other. All combinations where one polygon is picked from each group have to be examined.

Now in Step 5.2, we only need to divide P_A. Also, equilibrium calculations for the divided polygons are quite simple as it seems that we only need to calculate equilibria for games that are in Borm's class $c14$.

	L	R
T	a_1, a_2	b_1, b_2
B	c_1, c_2	d_1, d_2

Table 2: A 2×2 bimatrix game.

Definition 7. [7] Game in Table 2 is from class $c14$, if

$$\begin{cases} a_1 > c_1, \\ b_1 < d_1, \\ a_2 > b_2, \text{ and} \\ c_2 < d_2, \end{cases} \tag{21}$$

or if all four inequations are true in the reverse direction.

For a class $c14$ stage game, there are two pure-strategy equilibria (a_1, a_2) and (d_1, d_2). In addition, there is one mixed-strategy equilibrium (x_m, y_m), where

$$x_m = \frac{a_1 d_1 - c_1 b_1}{a_1 - b_1 - c_1 + d_1} \tag{22}$$

and

$$y_m = \frac{a_2 d_2 - c_2 b_2}{a_2 - b_2 - c_2 + d_2}. \tag{23}$$

We notice that x_m only depends on player 1's payoffs and y_m on player 2's. Also, the dependence is continuous when restrictions in Equation 21 hold. Therefore, in order to find $\mathcal{E}(P)$ for $P = \{P_A, P_B, P_C, P_D\}$ we only need to find the games that produce extremal payoffs. $\mathcal{E}(P)$ is then a polygon with those payoffs as vertices.

To find extremal payoffs, let us calculate the partial derivates of the mixed-action equilibrium point coordinates.

$$\begin{cases} \frac{\partial x_m}{\partial a_1} = \frac{(b_1 - d_1)(c_1 - d_1)}{(a_1 - b_1 - c_1 + d_1)^2} < 0 \\ \frac{\partial x_m}{\partial b_1} = \frac{(c_1 - a_1)(c_1 - d_1)}{(a_1 - b_1 - c_1 + d_1)^2} < 0 \\ \frac{\partial x_m}{\partial c_1} = \frac{(b_1 - a_1)(b_1 - d_1)}{(a_1 - b_1 - c_1 + d_1)^2} > 0 \\ \frac{\partial x_m}{\partial d_1} = \frac{(a_1 - b_1)(a_1 - c_1)}{(a_1 - b_1 - c_1 + d_1)^2} > 0 \end{cases} \tag{24}$$

$$\begin{cases} \frac{\partial y_m}{\partial a_2} = \frac{(b_2 - d_2)(c_2 - d_2)}{(a_2 - b_2 - c_2 + d_2)^2} < 0 \\ \frac{\partial y_m}{\partial b_2} = \frac{(c_2 - a_2)(c_2 - d_2)}{(a_2 - b_2 - c_2 + d_2)^2} > 0 \\ \frac{\partial y_m}{\partial c_2} = \frac{(b_2 - a_2)(b_2 - d_2)}{(a_2 - b_2 - c_2 + d_2)^2} < 0 \\ \frac{\partial y_m}{\partial d_2} = \frac{(a_2 - b_2)(a_2 - c_2)}{(a_2 - b_2 - c_2 + d_2)^2} > 0 \end{cases} \tag{25}$$

Now, if we want to find the extremal payoff surface $S(i,j)$, where

$$\begin{cases} i = 1 \text{ means objective 1 is to maximize player 1's payoff} \\ i = -1 \text{ means objective 1 is to minimize player 1's payoff} \\ j = 1 \text{ means objective 1 is to maximize player 2's payoff} \\ j = -1 \text{ means objective 1 is to minimize player 2's payoff,} \end{cases}$$

we can use information of the derivatives' signs and take from each polygon the corner points that contribute to the wanted objectives. Then we calculate equilibria for all stage games that can be created using those points. Some of those games do not produce extremal payoffs so we now must distinguish the Pareto points from these obtained equilibrium points. After all 4 surfaces $S(i,j)$ are calculated, we connect the extremal payoff points with an X-Y oriented polygonal chain and we have our set of all equilibria.

A problem arises when we are dealing with four polygons for which the x-coordinates of P_A and P_C or y-coordinates of P_A and P_B can cross. Now, if we follow the instructions of maximizing or minimizing what the derivatives tell us, we get games that are not from class $c14$. In these cases, one has to be more careful when calculating the extremal payoffs but calculation is not very complex for prisoner's dilemmas.

For example, in Figure 4 we have a situation where P_A has been split and we are examinating the case where we have to compute $\mathcal{E}(P'_A, P_B, P_C, P_D)$, where P'_A is the low-left part of P_A. Now when finding $S(1,1)$, the derivatives would tell us to choose from P_C the points that have maximal payoff for player 1 and minimal for player 2. From P'_A, both player's payoffs should be minimized. However, we have a restriction that the point chosen from P_A must have a smaller x-coordinate that the point from C_A, or else the game is not from class $c14$. In this case, x_m and y_m can be maximized by choosing the top-right corner from P'_A, any point with the same x-coordinate from P_C, any point with the same y-coordinate from P_B and any point from P_D. Then (x_m, y_m) is actually the top right corner of P'_A. Due to this, in many prisoner's dilemma games with small discounts the polygons stay in such shape that surface $S(1,1)$ only consists of one point.

3.3 Prisoner's dilemma with big discount factors

When discount factors are big, implementing the algorithm is a lot easier. We only have to calculate class $c1$ equilibria, because now they are possible and cover all others. This means that we find the biggest area that can be covered by rectangles that have one corner in each four polygons. Figure 5 shows an example.

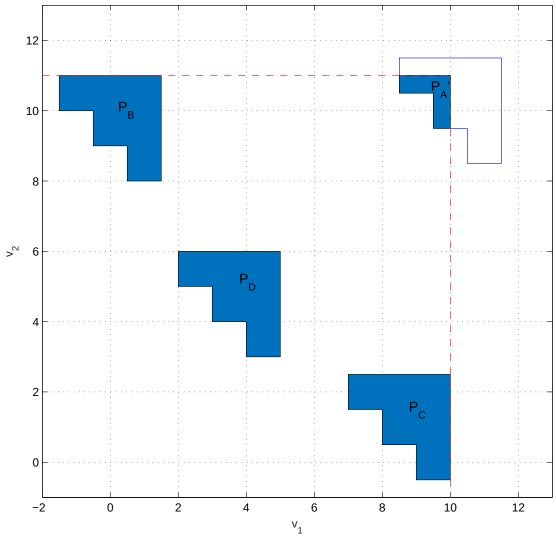

Figure 4: Suppose that a is the point chosen from P'_A. Here the four blue polygons are oriented so that from P_C we have to choose a point from the left of a and from P_B a point that is lower than a in order to create a game from class $c14$.

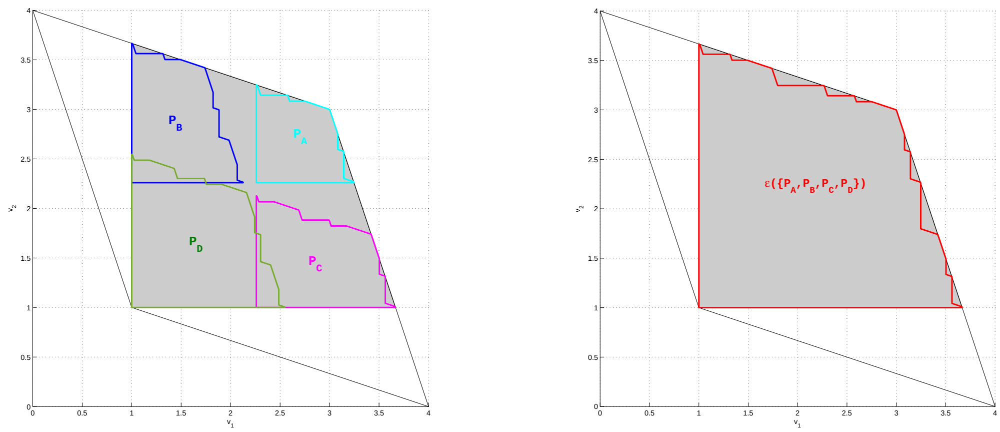

Figure 5: Example of operation $\mathcal{E}(\{P_A, P_B, P_C, P_D\})$ with a relatively big common discount factor. Here, we are computing the third iteration and $\delta_i = 0.58$ for both i. Polygons have been cut so that payoffs less than minmax have been discarded. The red polygon is the area than can be covered with rectangles that have one corner in each small polygon.

4 Results

4.1 Big discount factors

Results for game PD1 with common discount factor $\delta > \frac{1}{3}$ are presented in Appendix A. Actually, these results are produced by defining $W_0 = \mathcal{F}^\star$ in the beginning of the algorithm. In this case W is not X-Y convex. Here, $\frac{1}{3}$ is the smallest common discount factor for which class $c1$ equilibria are possible.

In Figure 9 of Appendix C, equilibrium sets for $\delta = 0.57$ and $\delta = 0.58$ are presented. It turns out that when $\delta < \frac{1}{\sqrt{3}} \approx 0.5774$, W converges to an octagon. When $\delta > \frac{1}{\sqrt{3}}$, the Pareto efficient payoff surface becomes more complex and computation time grows remarkably. This is the point where the pure-strategy profile of playing c or b twice and then a infinitely becomes possible. For a symmetric 2×2 prisoner's dilemma in Table 3 this specific discount factor can be calculated by solving

$$c + \delta c + \delta^2 a = d, \tag{26}$$

which gives

$$\delta = \frac{\sqrt{c^2 - 4a(c-d)} - c}{2a}. \tag{27}$$

	L	R
T	a, a	b, c
B	c, b	d, d

Table 3: A symmetric prisoner's dilemma, where $b < d < a < c$.

When $\delta \geq \frac{2}{3}$, V is the whole set of feasible and individually rational payoffs. In Figure 9 in Appendix C, mixed- and pure-strategy equilibrium sets are compared. Pareto efficient pure-strategy payoffs are corners of V with mixed strategies. It is an interesting result that mixed strategies do not allow equilibrium payoffs that are arbitrary linear combinations of a and c or a and b.

Appendix B has results for game PD1 with unequal discount factors. This allows V to go outside $\mathcal{F}^\star$. When $\delta_i \to 1$ for either i, $V \to W_0$. Figure 6 shows that V can have a billowy edge for big unequal discount factors. Algorithm can also handle asymmetric games.

4.2 Small discount factors

With smaller discount factors the algorithm's computation time grew remarkably. Game PD2 in Table 4 was studied with discount factors $\delta_i \leq \frac{1}{4}$. Figure 10 in Appendix D shows the result for $\delta = 0.24$. When $\delta < \frac{1}{4}$, V seems to become a very complex

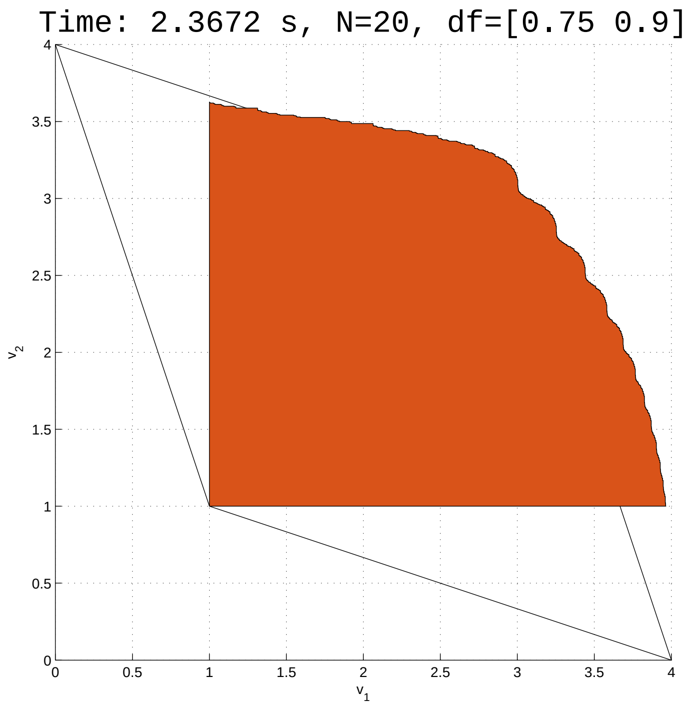

Figure 6: V can have a billowy Pareto surface for some discount factor combinations.

	L	R
T	$3.5, 3.5$	$0, 4$
B	$4, 0$	$1, 1$

Table 4: PD2

fractal surface. Here, nine iterations were calculated. This amount of iterations was not enough to see the final result to which V converges, but only small changes in it occurred on the last iterations.

For $\delta = 0.25$, V turned out remarkably bigger, which can been seen in Figure 11. Twelve iterations were calculated, and the last iterations did not change V in any other way but adding smaller and smaller corners. Thus, final shape of V is clear from the result.

An interesting result is, that V not a connected polygon, so not all randomizations between two equilibria are possible. For small δ, V seems to consist of infinitely many separate polygons. This is a big difference compared to games in which public randomization devices are allowed.

5 Conclusion

In, this study, a new algorithms for finding the set of subgame-perfect equilibrium payoffs in repeated prisoner's dilemmas with mixed strategies were created. Different algorithms were implemented for big and small discount factors. First, an algorithm was presented in a very abstract level for any 2×2 game. The key idea was using the fixed-point characterization by Abreu, Pierce and Stachetti. The algorithm was made iterative, such that it was meant to eventually, after enough iterations, find the wanted set. This study did not in any way consider what kind of strategies would result in the equilibrium payoffs.

The most difficult part of the algorithm was trying to find the union of set of equilibria for all stage games that could be created by choosing each payoff point from a certain set assigned for it. This operation was denoted by $\mathcal{E}$. The operation was divided into parts by handling the possible stage games class by class.

For big discount factors, reliable results were obtained. The algorithm was based on the key observation that only the stage games from Borm's class $c1$ had to be taken into account when computing the operation $\mathcal{E}$. In these games, the equilibrium sets are rectangles. The set of equilibria converged clearly in 15 iterations. This was easily computable, and the algorithm worked also for asymmetrical prisoner's dilemmas.

For small discount factors, a working algorithm was implemented only for certain discount factors. Implementation was much more difficult, because now stage games from the $c1$ class could not be created when calculating $\mathcal{E}$. An algorithm was implemented by taking into account only stage games from Borm's class $c14$, and it's extremal cases. The results were not as reliable, since omitting all other classes of stage games in some cases was based on a guess.

There are still some discount factors between the small and big factors, for which either of the implemented algorithms cannot be used. This is because for some discount factors the rectangles die away, and the orientation of sets from which the stage game payoffs are chosen is too complicated for the latter algorithm. Also, the algorithm is not reliable when the discount factors are really small or notably unequal.

In order to thoroughly solve this problem for a prisoner's dilemma for any discount factor, more exact treatment is needed. To confirm the results for small discount factors, more classes of stage games have to be checked when calculating $\mathcal{E}$. Also, the algorithm has to be made cleverly in order to keep the computation time realistic, and thus ensuring that enough iterations can be computed. To cover all possible discount factors, one has to carefully calculate the extremal payoff surfaces when implementing the operation $\mathcal{E}$. Various problems can arise, and if they are solved, probably the algorithm for any 2×2 game, or at least for some other games, can be also created then.

References

[1] Abreu, D. (1988) *On the theory of infinitely repeated games with discounting.* Econometrica, 56(2), p. 383

[2] Abreu, D., Pearce, D. and Stacchetti, E. (1986) *Optimal cartel equilibria with imperfect monitoring.* Journal of Economic Theory, 23, p. 251-269

[3] Abreu, D., Pearce, D. and Stacchetti, E. (1990) *Toward a theory of discounted repeated games with imperfect monitoring.* Econometrica, 58(5), p. 1041-1063

[4] Berg, K., Kitti, M. (2012) *Equilibrium paths in discounted supergames.* `http://sal.aalto.fi/publications/pdf-files/mber09b.pdf`

[5] Berg, K. and Schoenmakers, G. (2014), *Construction of randomized subgame-perfect equilibria in repeated games.* Working paper

[6] Blackwell, D. (1965), *Discounted dynamic programming.* Annals of Mathematical Statistics, 36, p. 226-235

[7] Borm, P.E.M. (1987), *A classification of 2x2 bimatrix games.* Cachiers du C.E.R.O., 29(1-2), p. 69-84.

[8] Mailath, G.J. and Samuelsson, L. (2006) *Repeated games and reputations: long-run relationships.* Oxford University Press.

[9] Nash, J.F. (1951) *Non-cooperative games.* The Annals of Mathematics, 54(2), p. 286-295

[10] Nicholl, T.M., Lee, D.T., Liao, Y.Z. and Wong, C.K. (1983), *On the X-Y convex hull of a set of X-Y polygons.* BIT Numerical Mathematics, 23(4), p. 456-471.

[11] Selten, R. (1965) *Spieltheoretische Behandlung eines Oligopolmodells mit Nachfragezeit, Zeitschrift für die Gesammte Staatswissenschaft.* 121, p. 301-324, 667-689

A Results with a large common discount factor

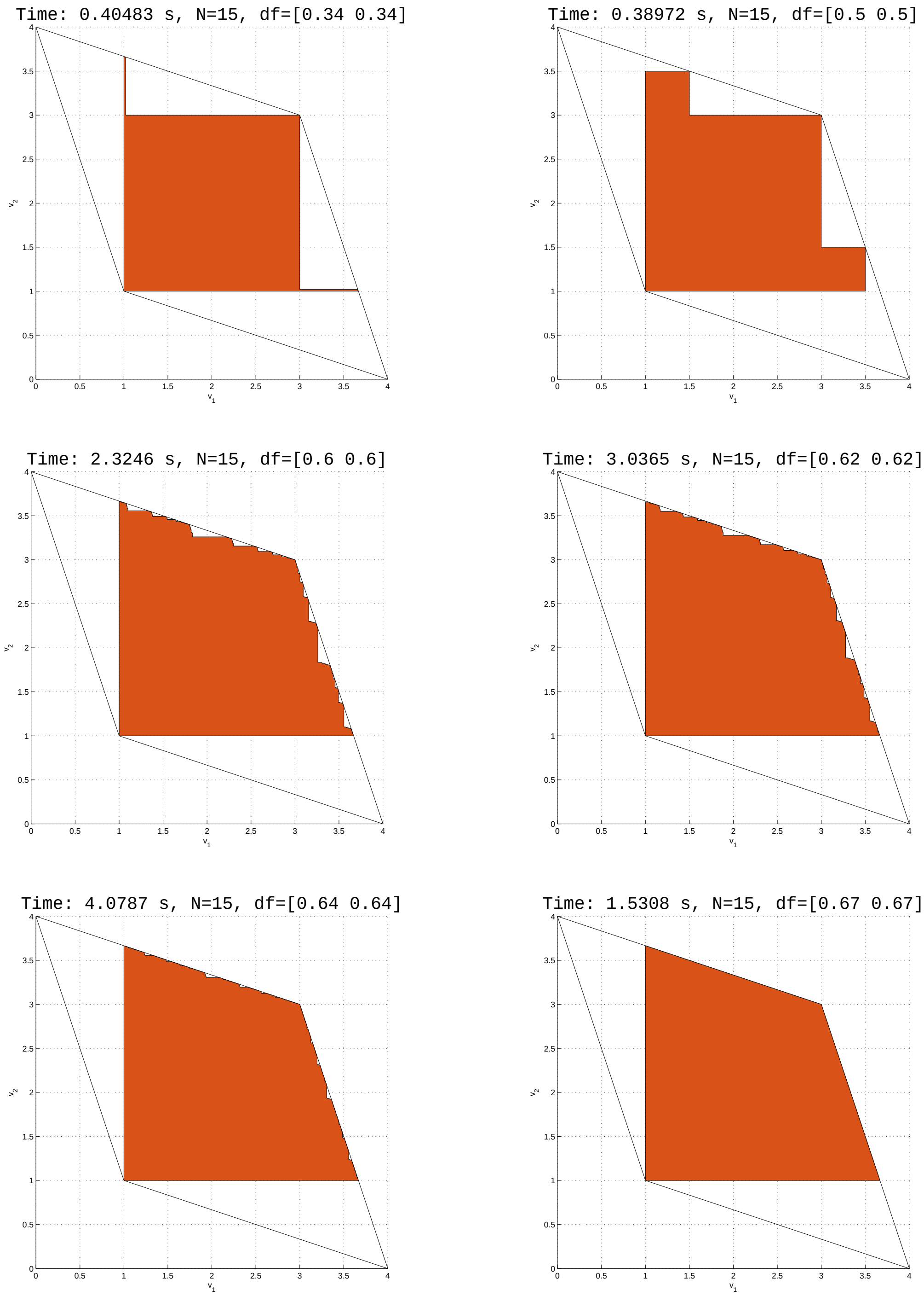

Figure 7: V for game PD1 with common discount factor $\delta \in [0.34, 0.67]$. Images were produced using Matlab. Here, `Time` is the running time of the algorithm, `N` is the amount of iterations and `df` is the discount factor vector. Labels v_1 and v_2 are the total payoffs for player 1 and player 2, respectively.

B Results with big unequal discount factors

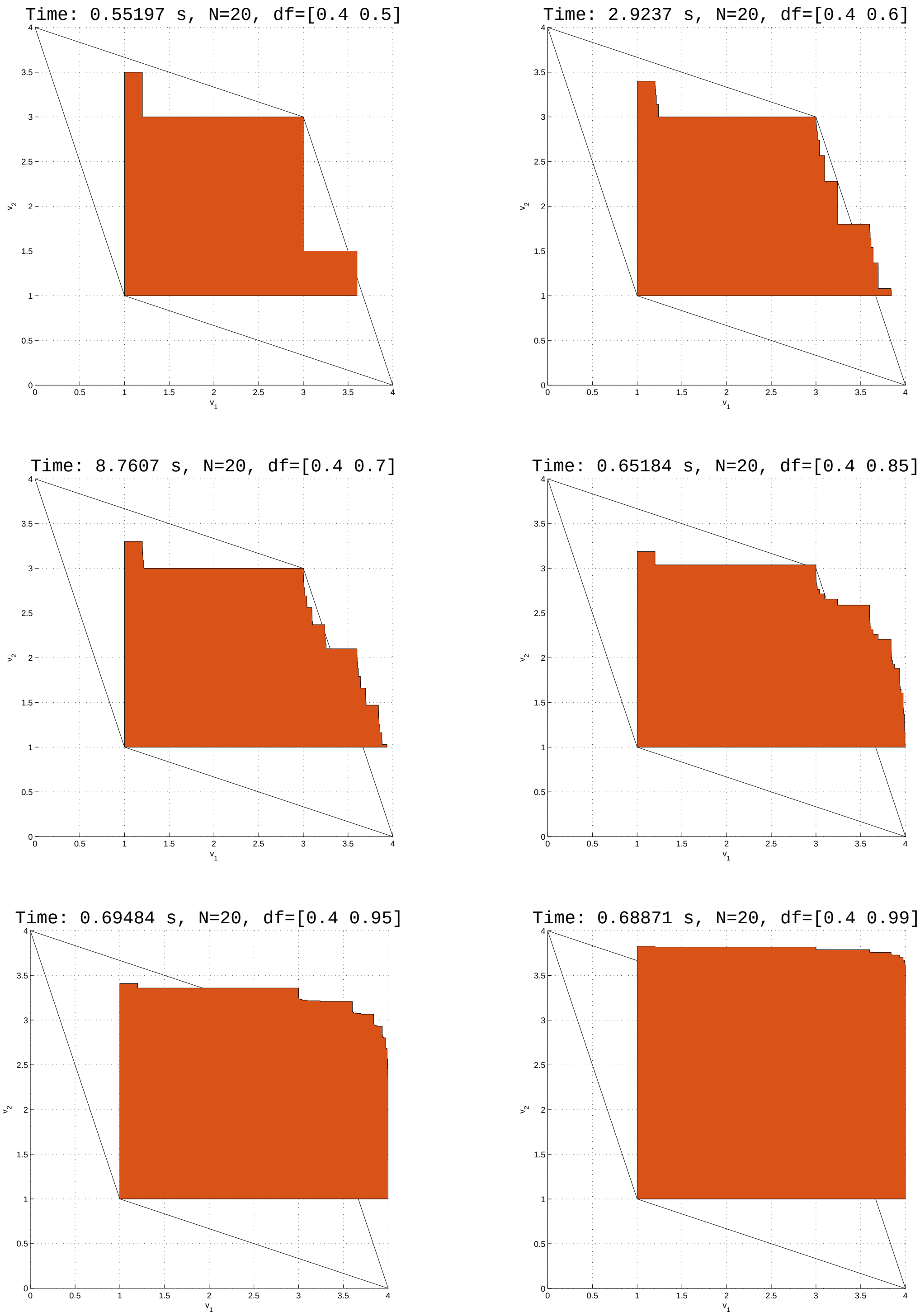

Figure 8: V for unequal discount factors in PD1.

C Comparison of equilibrium payoffs sets with mixed and pure strategies

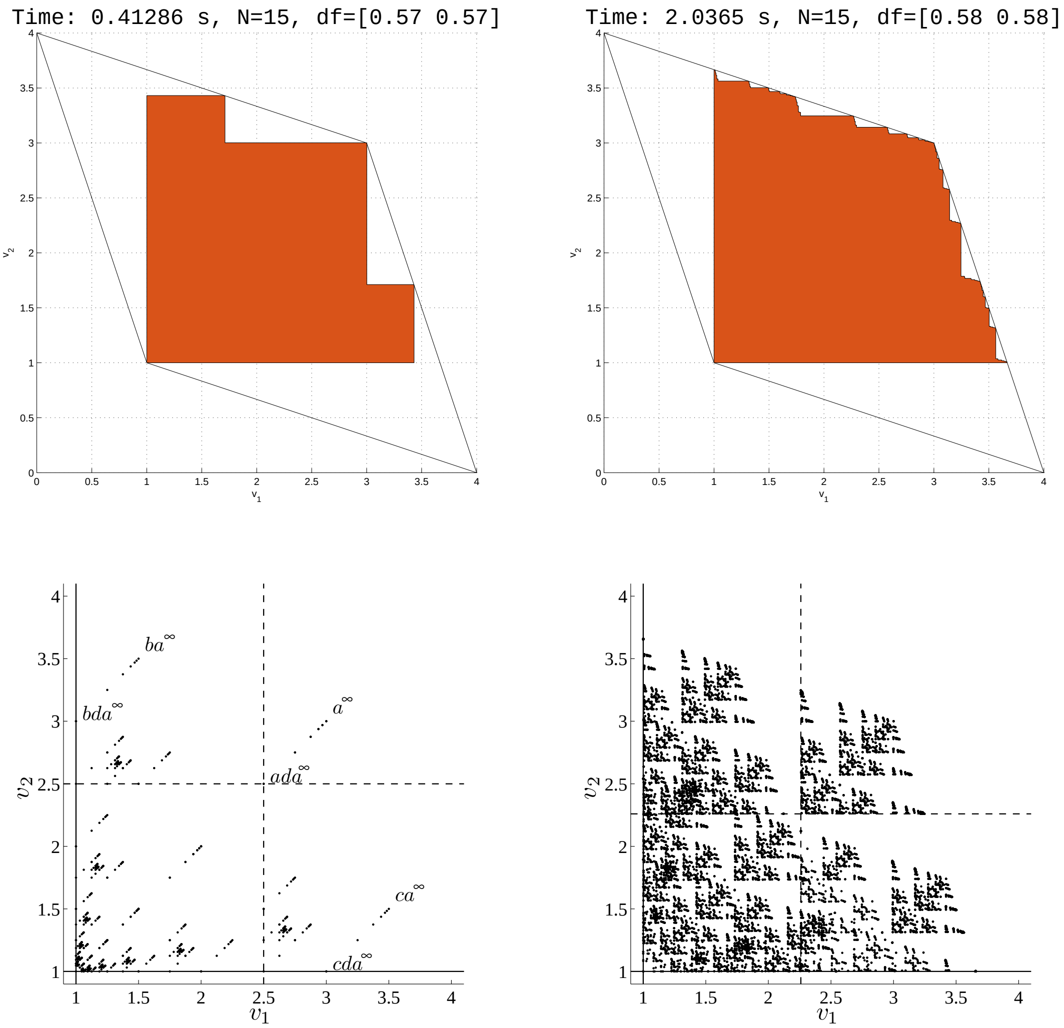

Figure 9: Comparison of the equilibrium payoff sets for mixed and pure strategies. Lower images are from [4]. Here, $\delta = 0.57$ on the left and $\delta = 0.58$ on the right.

D Results with small discount factors

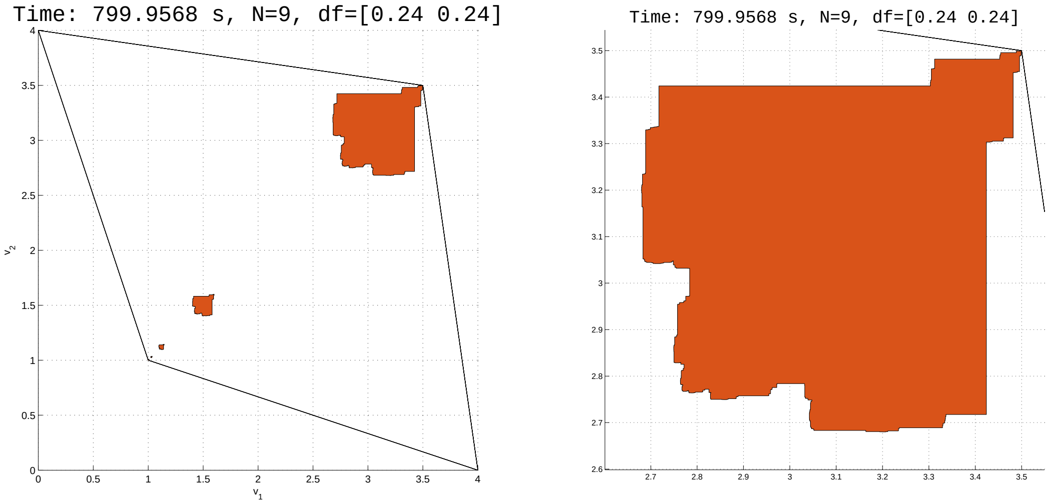

Figure 10: V for common $\delta = 0.24$ in PD2. On the right a magnified image of the biggest polygon after 9 iterations.

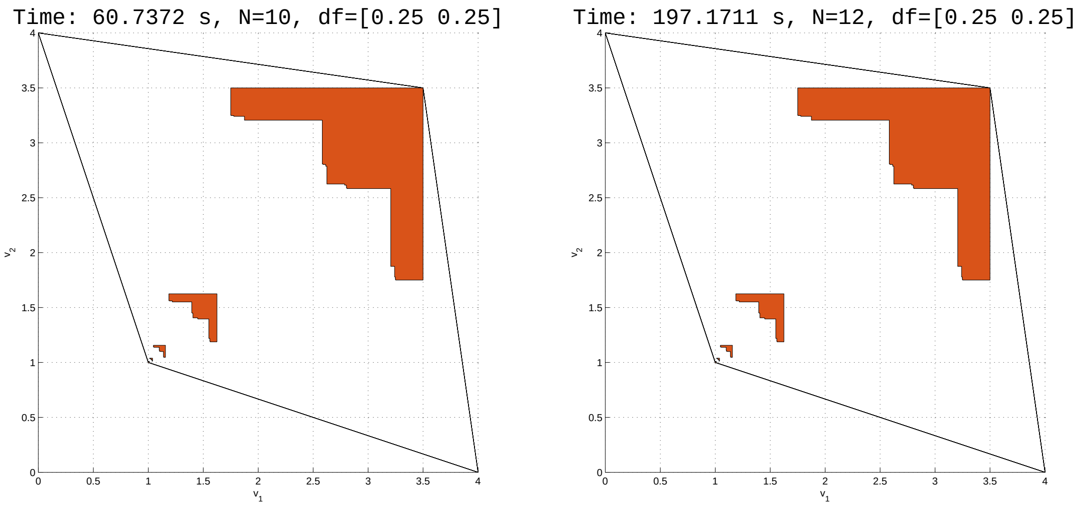

Figure 11: V for common $\delta = 0.25$ in PD2 after 10 and 12 iterations. Figures do not differ in a notable way.

E Finnish summary

Peliteoriassa mallinnetaan pelaajien, esimerkiksi yritysten, välisiä kilpailutilanteita. Pelissä pelaajan omat sekä muiden pelaajien ratkaisut vaikuttavat pelaajan saamaan hyötyyn, ja kukin pelaaja pyrkii maksimoimaan oman hyötynsä. Toistetuissa peleissä samaa peliä toistetaan uudelleen ja uudelleen. Pelaajat diskonttaavat kultakin kierrokselta saamansa hyödyn nykyarvoon tietyn diskonttokertoimen avulla ja laskevat näin pelistä saamansa kokonaishyödyn. Tässä työssä tutkittiin kahden pelaajan pelejä, joissa kummallakin pelaajalla on kullakin kierroksella kaksi vaihtoehtoa. Näistä keskityttiin Vangin ongelma -nimiseen tunnettuun peliin.

Pelaajan strategia määrää, minkä vaihtoehdon hän valitsee tietyn pelihistorian jälkeen. Osapelitäydellisiä tasapainoja ovat strategiaprofiilit, joissa kummallakin pelaajalla on sellainen strategia, että hänen ei kannata poiketa siitä pelihistorian jälkeen. Sekastrategiapeleissä pelaajat saavat käyttää strategioita, joissa tietty vaihtoehto valitaan aina jollain todennäköisyydellä. Puhtaiden strategioiden peleissä nämä tämä todennäköisyys on aina jollekin vaihtoehdolle yksi ja muille nolla. Pelaajien oletetaan havaitsevan vain sen vaihtoehdon, minkä toinen pelaaja tulee valinneeksi, eikä todennäköisyyksiä, joita tämä käyttää. Lisäksi pelaajien oletetaan havaitsevan ja muistavan kaikki omat ja toisen pelaajan valinnat.

Aiemmassa tutkimuksessa ei ole selvitetty, miltä osapelitäydellisten tasapainojen joukko näyttää sekastrategiapeleissä. Joukkoa on karakterisoitu ja puhtaiden strategioiden tapauksessa on laskettu tasapainopolkuja ja niiden tuottamia hyötyjä, jotka ovat joillekin peleille muodostaneet fraktaalimaisia kuvioita. Yleisesti tasapainojen laskemista on pidetty jopa turhana laskennan monimutkaisuuden takia.

Tavoitteena kandidaatintyössä oli luoda Matlab-algoritmi, jolla on mahdollista laskea osapelitäydellisten tasapainojen joukko annetussa toistetussa vangin ongelmassa, jossa pelaajat saavat käyttää sekastrategioita. Algoritmi lähdettiin toteuttamaan käyttämällä aiemmin esitettyä karakterisointia tasapainojoukoille. Ongelmaksi muodostui laskennallinen vaativuus. Tätä helpottamaan pyrittiin keksimään järkeviä keinoja, joilla laskentaa saisi nopeutettua. Tuotetun algoritmin oli määrä laskea iteratiivisesti tasapainojen joukkoa, kun niin sanottujen mahdollisten jatkohyötyjen joukko tunnetaan kullekkin mahdolliselle ensimmäisen pelikierroksen lopputulemalle. Jatkohyötyjen joukkoa päivitettäisiin joka iteraatiolla ja tätä toistetaisiin niin kauan kunnes tasapainojoukko suppenee.

Algoritmi toteutettiin erikseen sekä isojen että pienten diskonttokerrointen vangin ongelmille. Isojen kertoimien tapauksessa laskenta oli huomattavasti helpompaa ja luotettavia tuloksia saatiin. Pienten kertoimien tapauksessa tehokkaan laskenta-algoritmin tuottaminen oli haastavampaa, ja se saatiin toimivana toteutettua vain kapealle diskonttokerrointen joukolle. Tasapainojoukkoa ei pienillä kertoimilla saatu täysin suppenemaan, mutta mielenkiintoisia tuloksia saatiin joillekin kertoimille. Pienten diskonttokerrointen tapauksessa tulosten vahvistamiseksi vaadittaisiin huolellisempaa algoritmin toteutusta.

Avoimeksi kysymykseksi jäi, millainen tasapainojen joukko on todella pienillä diskonttokertoimilla, tai sellaisilla kertoimilla, jotka ovat käsiteltyjen "isojen" ja "pienten" kertoimien välissä. Ongelmaa ei siis kyetty täysin ratkaisemaan sen monimutkaisuuden vuoksi. Kuitenkin varsinkin pienille diskonttokertoimille luotua algoritmia voidaan tulevaisuudessa laajentaa niin, että se toimii mille tahansa kertoimille. Tämä voi kuitenkin vaatia hyvin tarkaa, systemaattista eri tilanteiden käsittelyä algoritmin edetessä ja osoittautua hankalaksi. Ongelman ratketessa algoritmi on luultavasti laajennettavissa moniin muihinkin kahden pelaajan peleihin.