SIR KARL POPPER SCHULE
WIEDNER GYMNASIUM

Vorwissenschaftliche Arbeit im Rahmen der Reifeprüfung

Simulation verschiedener Strategien im Prisoner's Dilemma

Wie das Unterteilen der Population in zwei Gruppen evolutionäre Spiele beeinflusst

Sebastian Pablik

8D 2016/17

Bundesgymnasium und Bundesrealgymnasium Wien 4
Wiedner Gymnasium/Sir Karl Popper Schule
A-1040 Wien, Wiedner Gürtel 68

Betreuungslehrperson: Mag.ª Sabine Bogner

Vorgelegt am 17.02.2017

Zusammenfassung

Das wiederholte Prisoner's Dilemma ist eines der führenden Paradigmen zum Aufkommen von Kooperation zwischen egoistischen Akteuren. Nach einer Einführung in seine Grundlagen, wird in der folgenden Arbeit das Verhalten von Gruppen im evolutionären Prisoner's Dilemma untersucht. Hierfür werden den Spielern einer Population zwei Sets stochastischer Strategien zugewiesen, die es ihnen erlauben, ihren Mitspieler als Mitglied einer von zwei möglichen Gruppen zu identifizieren und dementsprechend unterschiedlich zu agieren. Es wird mit einer Simulation festgestellt, dass im Normalfall eine der Gruppen die Population vollkommen dominiert. Diese Phasen sind von sehr effektiver Kooperation geprägt. Koexistenz beider Gruppen ist hingegen nur unter bestimmten Bedingungen möglich und führt zu deutlich verminderten Kooperationsraten. Insgesamt wirkt sich das Unterteilen in zwei Gruppen aber nachweisbar positiv auf den Durchschnittsgewinn aus.

Inhaltsverzeichnis

1 Grundlegendes

Fast jeder ist heutzutage mit dem Begriff *Spieltheorie* vertraut und vor allem ein Spiel wird häufig geradezu als Synonym für sie verstanden – das Prisoner's Dilemma. Die Regeln wurden erstmals 1950 von Merril Flood und Melvin Dresher definiert, seinen Namen aber erhielt es von dem von Albert Tucker präsentierten Beispiel (vgl. Kollock 1998, 185).

1.1 Das unwiederholte Prisoner's Dilemma

Zwei Verdächtige eines Verbrechens werden separat und ohne die Möglichkeit, untereinander zu kommunizieren, verhört. Beiden wird folgendes mitgeteilt:

1. Gesteht nur einer von beiden, während der andere leugnet, so wird ersterer als Kronzeuge freigelassen, zweiterer aber erhält eine Gefängnisstrafe von 5 Jahren.

2. Gestehen beide, so erhalten sie beide eine Gefängnisstrafe von 4 Jahren.

3. Leugnen beide, so erhalten sie beide eine Gefängnisstrafe von 2 Jahren.

(vgl. Tucker 1983, 228) Das Dilemma ergibt sich dadurch, dass einerseits der 3. Fall, gemeinsames Leugnen, mit nur 4 Jahren verteilter Gesamtstrafe eindeutig das beste Ergebnis für die Gefangenen als Kollektiv ist, es andererseits aber, vollkommen unabhängig von der Entscheidung des Mitspielers, für das Individuum stets besser ist zu gestehen als zu leugnen. Falls Spieler[1] 1 leugnet, kann Spieler 2 jedwede Strafe vermeiden, indem er Kronzeuge wird. Gesteht Spieler 1, ist Spieler 2 gezwungen, ebenfalls zu gestehen, um nicht selbst die Höchststrafe von 5 Jahren zu erhalten. Gleichzeitig führt dies aber zum 2. Fall, der unter beiden Gefangenen verteilt die höchstmögliche Gesamtstrafe bedeutet.

[1] Hier, wie auch im Folgenden, steht das Wort *Spieler* für das abstrakte spieltheoretische Konzept einer handlungsfähigen Entität und bezieht sich nicht nur auf menschliche Spieler und Spielerinnen

II I	Leugnen	Gestehen
Leugnen	-2 -2	0 -5
Gestehen	-5 0	-4 -4

Tabelle 1: Tabellarische Darstellung des Dilemmas. Zu jedmöglicher Entscheidung der Spieler I und II werden die entsprechenden Gewinne angegeben. In diesem Fall wird um Strafen gespielt, weshalb die Werte negativ sind.

Bei jedem endlichen Spiel kann ein sogenanntes Nash-Equilibrium gefunden werden, eine Situation, in der es für keinen der beiden Spieler Sinn ergibt, seine Strategie zu verändern, vorausgesetzt der andere tut dies ebenfalls nicht. Im standardmäßigen Prisoner‘s Dilemma ist dies der Fall wenn beide Gefangenen gestehen (vgl. Nash 1951).

Der Begriff *Prisoner's Dilemma* steht nicht nur für ein bestimmtes Spiel, sondern umfasst alle Zweipersonenspiele, in denen die Spieler zwischen zwei Aktionen wählen können [2], mit folgenden Gewinnwerten:

R (Reward – für beidseitige Kooperation (C-C))
T (Temptation – für Ausbeutung des kooperierenden Gegners (D-C)),
S (Sucker – für den Ausgebeuteten (C-D)) und
P (Punishment – für beidseitige Defektion (D-D))
Diese Werte müssen in einem Prisoner's Dilemma folgende Bedingungen erfüllen:

1. $T > R > P > S$
2. $2R > T + S$ (andernfalls würde Ausbeutung den höchsten Gesamtgewinn für die Population erzielen)

(vgl. Nowak et al. 1993, 56)

[2] Im obigen Beispiel *Leugnen* und *Gestehen*, diese werden aber üblicherweise durch C für *Cooperate* und D für *Defect* bezeichnet

II / I	C	D
C	R, R	S, T
D	T, S	P, P

Tabelle 2: Verallgemeinerte Darstellung des Prisoner's Dilemmas

Ein bedeutendes Beispiel für eine Untergruppe des Prisoner's Dilemmas ist das sogenannte *Donation Game*. Zwei Spieler haben unabhängig voneinander die Möglichkeit, c(cost) einzuzahlen, damit der Mitspieler b(benefit) erhält, wobei $0 < c < b$. Auf die allgemeine Darstellung angewandt bedeutet das R=b-c, T=b, S=-c, P=0, weshalb alle Beispiele des Prisoner's Dilemmas für die T-R=P-S auch als Donation Game beschrieben werden können (vgl. Hilbe et al. 2013, 3).

1.2 Das unendlich wiederholte Prisoner's Dilemma

Während es beim einmaligen Spiel in Wahrheit kein Dilemma gibt, da in D eine eindeutig dominante Strategie existiert, kann dasselbe nicht behauptet werden, wenn es unendlich wiederholt wird, bzw. endlich, aber keiner der beiden Spieler den Endzeitpunkt kennt. Die Gefangenen werden nicht nur einmal verhaftet und verhört, sondern immer und immer wieder. Jedes Mal erinnern sie sich beide, was bei den früheren Verhören passiert ist. Es ist zwar immer noch in jeder Runde für beide Spieler individuell besser zu defektieren, gleichzeitig ist es aber auch unabhängig von der eigenen Entscheidung besser, wenn der Mitspieler kooperiert. Das bedeutet, dass Strategien nun auch darauf basieren können, das Gegenüber dazu zu bewegen, in zukünftigen Wiederholungen C zu spielen. Die Angst vor Bestrafung bzw. Hoffnung auf Belohnung kann so vollkommen egoistische Agenten dazu verleiten, scheinbar selbstlos zu handeln (vgl. Fundenberg et al. 1990, 274). Dies funktioniert allerdings nur, solange es zukünftige Runden gibt, in denen die Entscheidung des Mitspielers beeinflusst werden kann. Kooperation kommt deshalb nur schwerlich in Spielen mit bekannter Runden- oder Zeitbegrenzung auf. In der letzten Runde gibt es keinen Grund mehr zu kooperieren, woraus folgt, dass es in der vorletzten

Runde ebenfalls keinen Anreiz dazu gibt. Jede letzte Entscheidung ist ein Beispiel des unwiederholten Prisoner's Dilemmas, in dem D die dominante Strategie ist, und dadurch in Wahrheit keine Entscheidung mehr. Deshalb ist in der Bezeichnung *wiederholtes Prisoner's Dilemma* üblicherweise impliziert, dass das Spiel – in der Theorie – unendlich of wiederholt wird.

1.3 Axelrods Turnier

1980 organisierte Robert Axelrod ein Computerturnier, basierend auf dem wiederholten Prisoner's Dilemma. Experten verschiedenster wissenschaftlicher Disziplinen wurden eingeladen, Programme zu schreiben und einzusenden. Diese würden dann gegen jedes der anderen eingesandten Programme antreten. Zur Verfügung standen ihnen dazu die Ergebnisse aller vorangegangenen Wiederholungen des jeweiligen Spiels, aus denen sie beliebig komplexe Schlüsse ziehen durften. Umso überraschender ist es, dass der letztendliche Sieger eine sogenannte Memory-One-Strategie war, die nur das Ergebnis der letzten Runde nutzt, um sich für C oder D zu entscheiden. Tatsächlich war es mit nur vier Zeilen Code das simpelste aller teilnehmenden Programme – Tit for Tat (TFT). TFT spielt in jeder Runde das, was der Gegner in der letzten Runde gespielt hat. Zu Beginn bietet es Kooperation an, die es so lange aufrecht erhält, als das auch der Gegner tut. Wählt dieser D, antwortet TFT mit Defektion – bis der Mitspieler wieder auf C wechselt (vgl. Axelrod 1980).

Das bedeutet aber nicht, dass TFT sich als Sieger des Prisoner's Dilemmas feiern kann. Tatsächlich verdankt es seinen Triumph zu großen Teilen dem Aufbau des Turniers. Zum einen hängt der Erfolg einer Strategie stark von der Population ab, gegen die sie antritt. Im richtigen Milieu kann jeder dominieren, das zeigen sogenannte Master-Slave Strategien. Diese erkennen einander an einer bestimmten Abfolge von C- und D-Zügen, woraufhin sich die eine der anderen unterwirft und sich für den Rest des Spiels ausbeuten lässt. Mit genügend Slave-Spielern endet solch eine Strategie mit Leichtigkeit an der Spitze jedes Turniers. So gab es auch in Axelrods Turnier von ihm *Kingmaker* genannte Strategien, mit denen TFT merklich besser kooperierte als seine Gegenspieler, und die ihm somit zum Sieg verhalfen (vgl. Axelrod 1980, 10).

1.4 Evolutionäres Prisoner's Dilemma

In der Realität verändert sich die Population selbst. Ineffektive Strategien werden durch effektive ersetzt. Im Kontext des Darwinismus, auf die das Prisoner's Dilemma häufig angewandt wird, gibt es auf Dauer keine Interaktion mit nicht überlebensfähigen Spezies. Deshalb werden Spiele häufig über mehrere Generationen hinweg betrachtet. Nach jedem Ablauf eines Turniers wie Axelrods wechselt ein Spieler, dessen Strategie in dieser Runde zu wenig Gewinn geführt hat, zu der eines Spielers mit höherem Punktgewinn. Von einer erfolgreichen Strategie im evolutionären wiederholten Prisoner's Dilemma spricht man daher als *evolutionär stabil*. Evolutionär stabil bedeutet, dass in einer von dieser Strategie dominierten Population keine konkurrierende Strategie die Oberhand gewinnen kann. Eine klare Vorraussetzung für evolutionär stabile Strategien ist daher, gegen sich selbst das bestmöglich Resultat zu erwirken. (vgl. Fundenberg et al. 1990, 274).

AllC (Always play C), zum Beispiel, ist nicht stabil, weil Strategien wie AllD (Always play D) es ausbeuten können (Gewinn=T), während AllCs untereinander nur Kooperation (Gewinn=R) erreichen. AllD selbst kann zwar nicht ausgebeutet werden, ist aber trotzdem ebenfalls in der Praxis nicht evolutionär stabil. TFT lässt sich nämlich ebenfalls nicht ausbeuten und erzielt mit AllD P (Gegenseitige Defektion) den selben Gewinn, den AllD auch mit sich selbst lukriert. Allerdings kann es mit sich selbst kooperieren, weshalb schon eine geringe Anzahl an TFT-Spielern minimal besser abschneidet als AllD, was auf die Dauer zu exponentiellem Wachstum führt (vgl. Nowak et al. 1992) – evolutionär stabile Strategien müssen also außerdem auch in Populationen mit einer kleinen Menge invadierender Strategien besser abschneiden als der Konkurrent. Nun scheint TFT immer noch evolutionär stabil. Es erzielt gegen sich selbst den höchstmöglichen Gesamtgewinn von 2R, und nie weniger Gewinn als sein Gegenspieler. [3]

Die Vorraussetzung für ein evolutionär stabiles TFT ist allerdings, dass jeder Spieler absolute Kontrolle über seine Handlungen hat. In der Realität sind Fehler und

[3] Nach der ersten Defektion des Gegeners bietet es erst wieder Kooperation an, wenn der Gegner selbst eine Ausbeutung durch TFT in Kauf nimmt. Der einzige Vorteil, den eine andere Strategie ihm gegenüber erreichen kann, ist die letzte Defektion des Spiels zu begehen. Ein Verrat bringt aber nur einen Vorteil von T-S Punkten, und hat in oft bzw. unendlich wiederholten Spielen wenig Bedeutung.

Missverständnisse möglich. So könnte sich in dem namensgebenden Szenario des Prisoner's Dilemmas einer der Gefangenen verplappern, was vom anderen als absichtliches Geständnis interpretiert wird. TFT hat in einem solchen Fall die Schwäche, dass es nicht grundlos vergibt. In einem Spiel mit *Rauschen* $\epsilon > 0$ (Wahrscheinlichkeit, anders zu spielen als beabsichtigt) wechselt TFT beim ersten Fehler zur Ausbeutung (C-D). Da beide stets den Zug des anderen kopieren, wechseln beide so lange zwischen (C-D) und (D-C) bis erneut einer der beiden einen Fehler begeht. Aber selbst dann kommt es nur in 50% der Fälle zur Kooperation (C-C). In den übrigen kommt es zu (D-D), was ebenfalls eine stabile Verhaltensweise ist. Der Hang zur Kooperation kommt also nur von der Aktion des ersten Zuges, und dieser hat bei unendlicher Wiederholung keinerlei Bedeutung. TFT erzielt deshalb gegen sich selbst letztendlich nur $(R+T+S+P)/4$ Gewinn. Es kann also von AllC oder GTFT [4] unterwandert werden (vgl. Fundenberg et al. 1990).

1.5 Strategien

Im Folgenden werden verschiedene Memory-One-Strategien beschrieben. Jede solche Strategie kann durch die vier Wahrscheinlichkeiten $(\mathrm{p}_1, \mathrm{p}_2, \mathrm{p}_3, \mathrm{p}_4)$ beschrieben werden, die mit den Ergebnissen R, T, S, P zusammenhängen. Jeder Wert $\mathrm{p_i}$ gibt die Wahrscheinlichkeit an, mit der die Strategie auf das entsprechende Ergebnis der letzten Runde mit Kooperation antwortet (vgl. Nowak et al. 1993, 57).
Ein hoher p_1 Wert bedeutet, dass sie gegenseitige Kooperation (R) nur selten ohne Provokation bricht, während ein niedriger Wert vor allem bei Strategien auftritt, die von der Ausbeutung AllC-ähnlicher Strategien abhängig sind.
p_2 beschreibt die Wahrscheinlichkeit nach erfolgreicher Ausbeutung des Mitspielers (T) Kooperation anzubieten.[5]
Der Wert p_3 ist die Antwort auf S, gibt also die Wahrscheinlichkeit an, eine Ausbeutung durch den Gegner zu dulden, statt Vergeltung zu üben und ebenfalls D zu

[4] Generous TFT ignoriert mit einer bestimmten Wahrscheinlichkeit eine Defektion des Gegners, wodurch die Gefahr, einen Fehler eines grundsätzlich kooperationswilligen Mitspielers zu bestrafen, verringert wird.

[5] Man merke, dass $\mathrm{p_i}$ *nicht* der Wahrscheinlichkeit zwischen den Aktionen C und D zu wechseln, bzw. zu stagnieren, entspricht. Ein höheres $\mathrm{p_i}$ bedeutet stets eine höhere Wahrscheinlichkeit zu kooperieren. Ob dies zu einer Veränderung der Situation führt, hängt davon ab, ob der Spieler in der letzten Runde C oder D gespielt hat.

spielen.
Somit steht p_4 für die Wahrscheinlichkeit, gegenseitige Defektion (P) zu brechen, und Kooperation anzubieten. In einem Spiel mit Rauschen ϵ gilt die Vorraussetzung $\epsilon < p_i < (1-\epsilon)$, alle Werte dazwischen sind aber erlaubt. Obwohl die meisten erfolgreichen Strategien, wie TFT, AllC und AllD, p_is aufweisen, die sehr nahe an dem Minimal- oder Maximalwert liegen, während zum Beispiel die Strategie *Random* mit den Wahrscheinlichkeiten (0.5,0.5,0.5,0.5) meist sehr schlecht abschneidet, können einzelne mittelhohe Werte auch eine entscheidende Rolle in bestimmten erfolgreichen Strategien spielen.
Das Rauschen im wiederholten Prisoner's Dilemma kann grundsätzlich auf zwei verschiedene Arten angewandt werden:

1. Mit Wahrscheinlichkeit ϵ wird das Gegenteil der beabsichtigten Aktion gespielt, also C statt D, bzw. D statt C.

2. Mit Wahrscheinlichkeit ϵ wird die gespielte Aktion vom Gegner als ihr Gegenteil missverstanden, zur Berechnung des Gewinns aber wird die gewollte Aktion herangezogen.

Beide Methoden führen dazu, dass zur Berechnung der nächsten Aktion die selben p_is angewandt werden, als hätte der vom Rauschen betroffene Spieler absichtlich die gegenteilige Aktion gewählt. Der Unterschied besteht darin, dass mit der zweiten Methode zuerst der Gewinn ausgezahlt und danach erst Rauschen angewandt wird, wodurch im Gegensatz zur ersten Methode das Ergebnis einer Runde nicht direkt beeinflusst wird. Im Folgenden wird generell die erste Methode angewandt.

1.5.1 AllC

AllC ist die Strategie $(1-\epsilon, 1-\epsilon, 1-\epsilon, 1-\epsilon)$. Jede Strategie mit $p_1 < 1-\epsilon$ oder $p_2 < 1-\epsilon$ ist in der Lage, es zu verdrängen (eine Srategie kann nicht evolutionär stabil sein, wenn andere Strategien mit ihr bessere Ergebnisse erzielen als sie selbst). AllCs Stärke besteht darin mit sich selbst $(1-\epsilon)^2 R + \epsilon(1-\epsilon)(T+S) + \epsilon^2 P$ zu erzielen, den höchsten Gewinn, den eine Strategie mit sich selbst erreichen kann. Dies erlaubt ihm, jede Strategie mit $p_1 = p_2 = 1-\epsilon$ zu unterwandern.

1.5.2 AllD

AllD mit $(\epsilon, \epsilon, \epsilon, \epsilon)$ ist das genaue Gegenteil von AllC. Da es gegen sich selbst den niedrigstmöglichen Wert von $(1-\epsilon)^2 P + \epsilon(1-\epsilon)(T+S) + \epsilon^2 R$ gewinnt, kann es sich gegenüber TFT nur in Populationen mit vielen Spielern mit hohen p_1 und p_2 Werten durchsetzen, im Gegensatz zu AllC kann aber niemand vom Spiel mit AllD direkt profitieren.

1.5.3 TFT

TFT ist die Strategie mit den Wahrscheinlichkeiten $(1-\epsilon, 1-\epsilon, \epsilon, \epsilon)$. Abgesehen von direkten Konsequenzen des Rauschens beschränkt TFT den Gewinn seines Mitspielers stets auf den eigenen, weshalb es, wie AllD, nicht von einem einzelnen Spieler verdrängt werden kann. Da es gegen sich selbst aber nur den zufälligen Gewinn $\frac{R+T+S+P}{4}$ erspielt, kann es von jeder Strategie, die mit sich selbst einen höheren Gewinn erzielt, unterwandert werden.

1.5.4 AllC, AllD und TFT – Equilibrium

In einem auf die drei oben genannten Strategien beschränkten Spiel kommt es also zu einem Kreislauf zwischen AllC, AllD und TFT. Die genaue Veränderung, die in einer solchen Population stattfindet, kann einerseits simuliert werden (siehe Abbildung 1), andererseits aber auch mithilfe einer Gewinnmatrix berechnet werden (siehe Tabelle 3).

	A	B
A	a	b
B	c	d

Tabelle 3: Gewinnmatrix der Strategien A und B. Die Kleinbuchstaben symbolisieren jeweils den durchschnittlichen Gewinn, den die Strategie der entsprechenden Zeile mit der Strategie der entsprechenden Spalte erzielt.

Ist in einer Population α die relative Häufigkeit der Strategie A im Vergleich zur Häufigkeit β der Strategie B, so kann der durchschnittliche Gewinn der jeweiligen Strategien A und B als $\alpha a + \beta b$, bzw. $\alpha c + \beta d$ berechnet werden. Aus dem Verhältnis

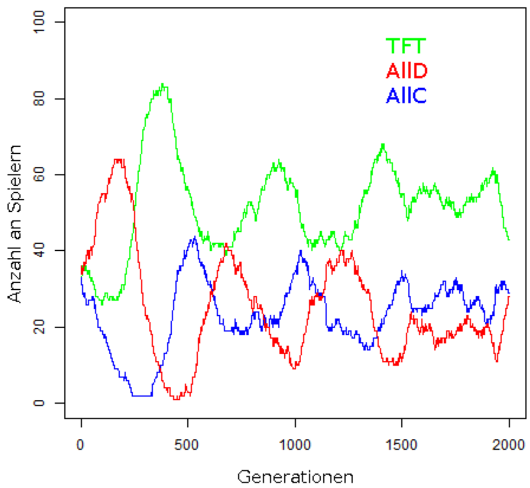

Abbildung 1: Häufigkeit der Strategien AllC, AllD und TFT über einen Zeitraum von 2000 Generationen in einer Simulation des evolutionären Prisoner's Dilemmas, welches auf diese drei Strategien beschränkt ist. Erkennbar ist die zyklische Abfolge AllC - AllD - TFT - AllC - ...

dieser beiden Werte kann die Wahrscheinlichkeit jedmöglicher Veränderung abgeleitet werden. Da in einer unendlich großen Population der Interaktion mit der einfallenden Strategie keinerlei Bedeutung zukommt, ist A gegen einzelne Einwanderer evolutionär stabil wenn $a > c$. Dasselbe gilt für B wenn $b > d$. Diese zwei Möglichkeiten schließen einander allerdings nicht aus. Sind sowohl A als auch B gegeneinander evolutionär stabil, gibt es noch ein drittes (instabiles) Equilibrium in der Verteilung $\alpha : \beta$, für die gilt: $\alpha a + \beta b = \alpha c + \beta d$. Die kleinste Schwankung würde dieses Equilibrium zerbrechen, in einer deterministisch agierenden Population kann es aber bestehen bleiben.

Ein Gleichgewicht kann auch gefunden werden, wenn $a < c$ und $b < d$, also beide Strategien instabil sind. Dadurch, dass sie mehr von Spielern der jeweils anderen Strategie profitieren als von jenen der eigenen, kann keine der beiden Strategien die andere vollkommen verdängen. In diesem Fall gibt es nur ein einziges Equilibrium, zu dem jede solche Population tendiert (vgl. Nowak et al. 2004).

Die gleichen Berechnungen können auch mit mehr als zwei Strategien durchgeführt werden. So können die drei oben beschriebenen Strategien mit der Gewinnmatrix in

Tabelle 4 beschrieben werden.

	AllC	AllD	TFT
AllC	R	S	R
AllD	T	P	P
TFT	R	P	$\frac{R+T+S+P}{4}$

Tabelle 4: Gewinnmatrix der Strategien AllC, AllD und TFT bei unendlicher Wiederholung mit infinitesimalem Rauschen.

In diesem Fall kann das Equilibrium also berechnet werden als:
$\alpha R + \beta S + \gamma R =$
$= \alpha T + \beta P + \gamma P =$
$= \alpha R + \beta P + \gamma \frac{R+T+S+P}{4}$
In einer Bevölkerung mit α AllC, β AllD und γ TFT sind alle Zeilensummen der Gewinnmatrix (Tabelle 4) ident, somit gibt es keinen Grund für irgendeinen der Spieler, seine Strategie zu verändern.

1.5.5 GTFT

Generous TIT FOR TAT (GTFT) ist unter den *reaktiven* Strategien,[6] die nicht von kooperationsunwilligerern Strategien (solche mit niedrigeren Werten für p oder q) verdrängt werden können, diejenige, die den höchsten Gewinn mit sich selbst erzielt. Wie in 1.5.4 erwähnt, muss auf der Gewinnmatrix (Tabelle 3) $a > c$ sein, damit eine Strategie evolutionär stabil ist. Es muss also GTFT im Spiel mit GTFT mehr Gewinn erzielen als AllD im Spiel mit GTFT. Die exakten Werte (p,q) hängen von den Verhältnissen zwischen R, T, S und P ab, können aber verallgemeinert als $(1-\epsilon, q)$ mit $q = min\{1 - (T-R)/(R-S), (R-P)/(T-P)\}$ beschrieben werden (vgl. Nowak et al. 1992). Bei R=3, T=5, S=0, P=1 entspricht das $(p = 1, q = \frac{1}{3})$. GTFT ist die dominante Strategie in rein reaktiv agierenden Populationen. Aber auch in solchen, die vier verschiedene p_is erlauben, kann es oft über lange Zeiträume dominieren[7] (vgl. Nowak et al. 1993).

[6]Strategien, für die ($p_1 = p_2, p_3 = p_4$). Ihre Reaktion hängt nur vom Zug des Gegners ab. Sie können durch die Wahrscheinlichkeiten (p,q) als Antwort auf die Züge (C,D) angegeben werden (vgl. Nowak et al. 1990)

[7]In stochastischen Simulationen kann jeder denkbaren Veränderung eine Wahrscheinlichkeit $p > 0$ zugewiesen werden. Es bleibt also kein Equilibrium unendlich lange bestehen.

1.5.6 WSLS

Win-Stay, Lose-Shift (WSLS) basiert auf der Strategie $(1-\epsilon, \epsilon, \epsilon, 1-\epsilon)$, die, wie der Name impliziert, immer dann wechselt, wenn es einen der unterdurchschnittlichen Gewinnwerte S und P erzielt hat. Dies bringt mehrere Vorteile: Erstens kann WSLS im Gegensatz zu TFT mit sich selbst nahezu durchgehend kooperieren, da beide von gegenseitiger Defektion sofort zu Kooperation wechseln. Zweitens läuft es nicht wie andere Strategien Gefahr, von AllC unterwandert zu werden. Sobald es einmal durch einen Fehler zur Defektion bewegt wird, bietet es erst dann wieder Kooperation an, wenn der andere sich zur Wehr setzt. Gut,ütige Strategien wie AllD beutet es dadurch aus, was ihren Gewinn beschränkt und ihnen nicht erlaubt an die Macht zu kommen. Allerdings erzielt AllD gegen WSLS einen Gewinn von $\frac{T+P}{2}$, dessen Verhältnis zu R nicht axiomatisch definiert ist. In Spielen, in denen $\frac{T+P}{2} > R$ gilt, ist die reine Strategie WSLS nicht evolutionär stabil. Stattdessen muss, wie bei GTFT, die Wahrscheinlichkeit gesenkt werden bis Defektion sich nicht mehr auszahlt - in diesem Fall auf $(1-\epsilon, \epsilon, x)$ mit $x < \frac{R-P}{T-R}$. Mit diesen angepassten Wahrscheinlichkeiten zeichnet sich WSLS durch eine sehr hohe Stabilität aus und dominiert in vielen Simulationen die Population für den Großteil der Zeit (vgl. Nowak et al. 1993).

2 Prisoner's Dilemma mit zwei Gruppen

Spieltheoretische Modelle im allgemeinen Leben wiederzuerkennen ist oft schwer. Oft werden die Natur des Spiels oder die Spieler selbst auch während des Ablaufes von äußeren Faktoren beeinflusst. So wird im Folgenden eine Variation des Prisoner's Dilemmas betrachtet, in der jeder Spieler zusätzlich zu der Information über den Ausgang der letzten Runde des Spiels seinen Gegner als Mitglied einer von zwei Gruppen erkennen kann. Jeder Spieler spielt deshalb zwei voneinander unabhängige Strategien – eine gegen Spieler der eigenen Gruppe, die andere gegen Spieler der anderen Gruppe. Ansonsten hat die Gruppenzugehörigkeit aber keinerlei Einfluss auf den Spielmechanismus.

2.1 Aufbau der Simulation

Begriffserklärung

Bevölkerung (auch Population)	Gesamtheit aller Spieler
Generation	Zeitpunkt in der Evolution der Bevölkerung
Spieler	Träger einer Strategie; verändert sich nur durch das Wechseln ebenjener
Strategie	Zusammenfassung der vier (bzw. acht) $\mathrm{p_i}$s

Aufbau

In der verwendeten Simulation wird eine Population von 100 Spielern über 50 000 Generationen hinweg beobachtet. Jeder dieser Spieler ist definiert durch acht Werte zwischen 0,03 und 0,97 (Rauschen $\epsilon = 0,03$). Diese geben die Wahrscheinlichkeit zu kooperieren nach den vier Szenarien R,T,S oder P mit jeweils Mitgliedern der eigenen oder anderen Gruppe an. Weiters noch einen Wert 1 oder 2, der die eigene Gruppenzugehörigkeit widerspiegelt. Zu Beginn werden alle Wahrscheinlichkeiten mit Zufallsvariablen der Verteilung von Abbildung 2 befüllt.[8] Die Gruppenzugehörigkeit wird unabhängig von diesen acht Werten zufällig zugeteilt, wobei beide Gruppen gleich wahrscheinlich sind.

In jeder Generation werden für jeden Spieler zufällig 16 Gegner ausgewählt, gegen die er jeweils in einem 15-fach[9] wiederholten Prisoner's Dilemma antritt.[10] Danach

[8]Die Bevorzugung der Wahrscheinlichkeiten an den äußeren Rändern erlaubt ein schnelleres Überwinden des zu Beginn entstehenden Chaos.

[9]Da der Aufbau der Simulation es den Spielern nicht ermöglicht, die derzeitige Runde festzustellen, verletzt auch eine fixierte Rundenbegrenzung nicht das Prinzip der unbekannten Wiederholungsanzahl.

[10]Ideal wäre es, jeden gegen jeden antreten zu lassen, mit so vielen Wiederholungen in jedem Spiel wie möglich. Ein Testlauf ergab aber, dass bereits bei den letztendlich gewählten Werten die Varianz zwischen den Ergebnissen wiederholter Durchführungen der selben Prozesse sehr gering war

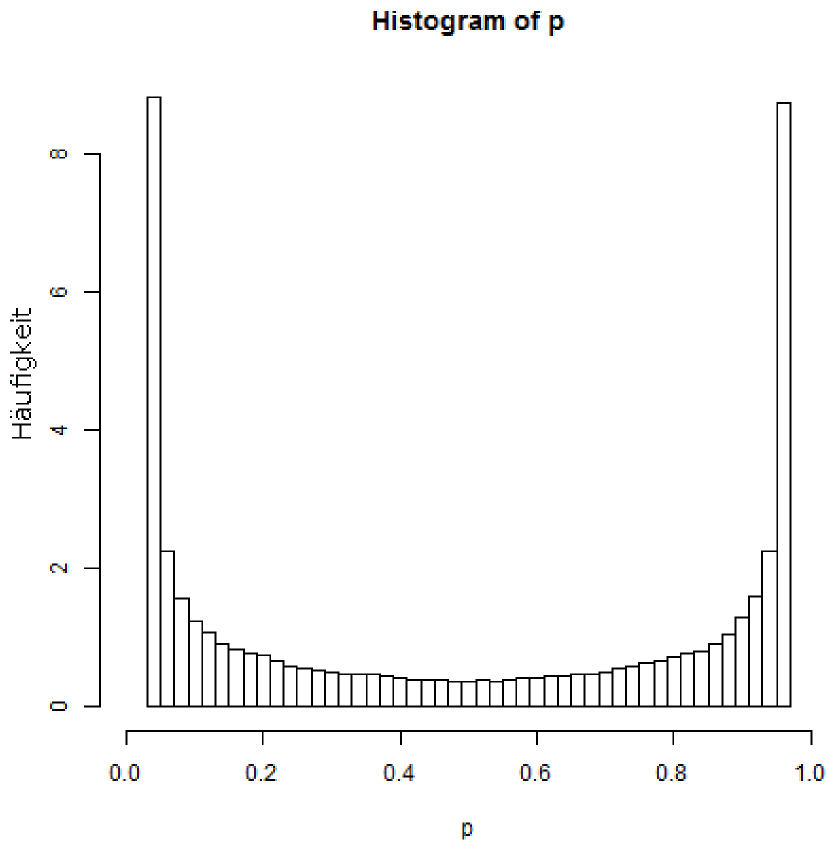

Abbildung 2: Histogramm der relativen Häufigkeit verschiedener p_is bei der zufälligen Erstellung neuer Strategien.

wird die Strategie eines Spielers mit hohem Gewinn verdoppelt, indem ein Spieler mit niedrigem Gewinn sie statt seiner bisherigen Strategie annimmt. Die Wahrscheinlichkeit, seine Strategie weiterzugeben, entspricht hierbei dem Anteil seines Gewinnes am Gesamtgewinn der Population. Gegenteilig wird bei der Auswahl der ausscheidenden Strategie die Differenz zwischen dem eigenen Gewinn und dem Maximalgewinn T beobachtet. Für schnellere Ergebnisse wird dieser Mechanismus in jeder Generation dreimal durchgeführt. Bei jeder dieser Ersetzungen wird Mutation angewandt, indem zu jedem p_is mit einer Wahrscheinlichkeit von 0,05 eine um 0 normalverteilte Zufallsvariable mit einer Standardabweichung $s = 0,05$ hinzugefügt wird. Werte, die hierdurch die definierten Begrenzungen über- bzw. unterschreiten werden auf den nähesten akzeptablen Wert angepasst (vgl. Hilbe et al. 2013, 11).

Die Gruppenzugehörigkeit ist von dieser Veränderung nicht betroffen. Zusätzlich tritt Mutation auch dadurch auf, dass mit einer Wahrscheinlichkeit von 0,02 ein weiterer Spieler durch den oben beschriebenen Algorithmus ausgewählt und durch eine neue zufällige Strategie ersetzt wird. Diese wird mit derselben Methode bestimmt, mit der auch die Anfangspopulation erstellt wird, es sind also beide Gruppenzugehörigkeiten möglich. Die Anwendung von Mutation ist deshalb notwendig, da sonst nur jene Strategien auftreten können, die bereits existieren, was letztendlich zu unendlicher Dominanz einer einzelnen Strategie führen würde.

Unter anderem werden mit diesem Simulationsaufbau die folgenden Fragestellungen untersucht:

- Wie wirkt sich das Einführen von Gruppen auf den durchschnittlichen Gewinn und die Stabilität der Population aus?
- Wie oft dominiert eine Gruppe die andere und wodurch zeichnen sich diese Phasen im Vergleich zu solchen, in denen beide koexistieren, aus?
- Welche Unterschiede bestehen zwischen dem Verhalten zur eigenen Gruppe und dem Verhalten zur anderen?
- Welche Bedingungen führen in Populationen mit zwei Gruppen zu hohem Gewinn/ hoher Stabilität?

2.2 Ergebnisse

2.2.1 Verhalten der Gruppen

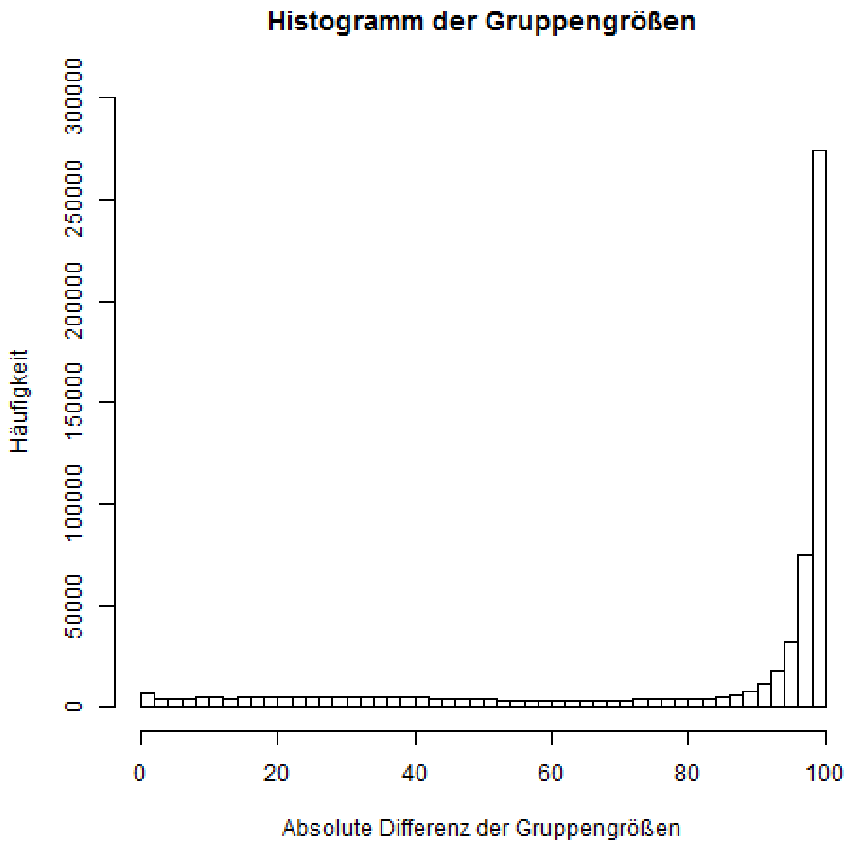

Abbildung 3: Häufigkeit aller möglichen Differenzen zwischen den Gruppengrößen über alle Generationen und Simulationen hinweg.

In allen Durchgängen ist eine klare Tendenz zu vollkommener Dominanz einer einzelnen Gruppe erkennbar (siehe Abbildung 3). In nahezu 50% aller Generationen existiert überhaupt nur eine Gruppe (rechteste Säule), in mehr als zwei Dritteln hält eine der Gruppen mehr als 95% der Bevölkerung inne. Dies lässt sich dadurch erklären, dass eine Gruppe den Gewinn der anderen Gruppe beschränken kann, ohne gleichzeitig ihren eigenen zu vermindern, was den Spielern der kleineren Gruppe unabhängig von ihrer eigenen Strategie einen Nachteil verschafft. Bei gruppeninternen Spielen ist das Verhalten der dominanten Gruppe in stabilen Phasen meist von

Variationen von GTFT geprägt, häufig aber auch durch WSLS oder die GRIM[11] (siehe Abbildung 4). Bemerkenswert ist dabei das nahezu vollkommene Fehlen von AllD. Mit der anderen Gruppe hingegen spielt sie fast ausschließlich AllD.

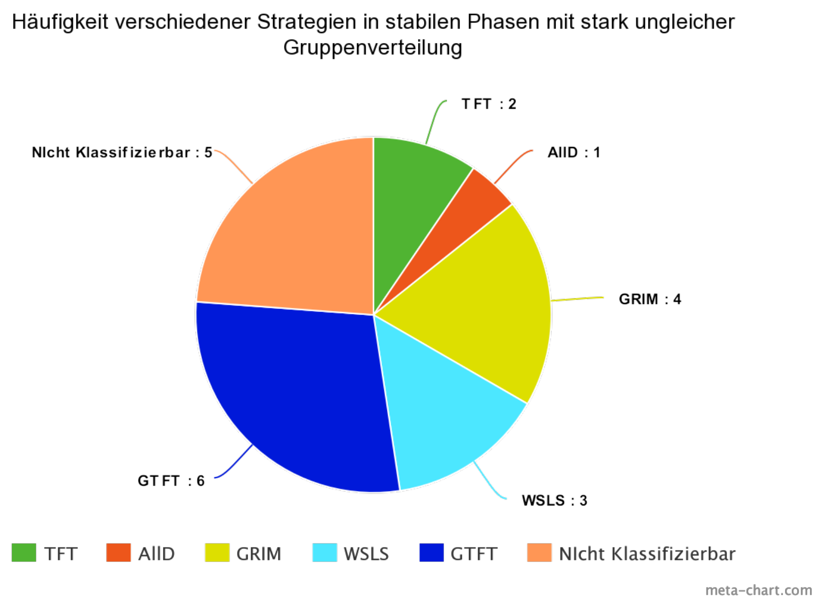

Abbildung 4: Häufigkeit, mit der verschiedene Strategien in stabilen Phasen mit stark ungleichen Gruppenverhältnissen innerhalb der dominanten Gruppe auftreten.

Oft halten die stabilen Phasen auch nicht lange genug an, um sich auf optimale Strategien einzupendeln – denn sobald eine Gruppe die andere komplett verdrängt hat, wird ihr Verhalten zu dieser einem zufälligen Drift ausgesetzt. Da die Strategie gegenüber der anderen Gruppe nicht mehr den Gewinn des Spielers beeinflusst, unterliegt die Veränderung dieser nicht mehr evolutionären Prinzipien, sondern erfährt stattdessen zufällige Schwankungen. Strategien, die sich nur durch das Verhalten zur ausgelöschten Gruppe unterscheiden, teilen die Population zufällig unter sich auf. Sobald einmal zufällig ein hoher Anteil an großzügigen Strategien gegenüber der anderen Gruppe existiert, ist ein Mitglied jener in der Lage, im Durchschnitt mehr Gewinn mit den Mitgliedern der ersten Gruppe zu machen als diese selbst, was ihm erlaubt die Gruppe zu übernehmen.

Ein weiteres Element, das oft zur Machtübernahme der kleineren Gruppe führt, ist

[11] GRIM war eine Strategie in Axelrods Turnier, die nach einmaliger Defektion des Gegners nie wieder Kooperation anbietet. Als Memory-One-Strategie kann es als (1,0,0,0) beschrieben werden

eine Art Trojanisches-Pferd-Effekt. Ein zufällig erstellter Mutant der dominanten Gruppe entsteht mit einer besonders effektiven Strategie gegen die eigene Gruppe, ist aber gegenüber Invasoren der anderen Gruppe nicht evolutionär stabil. Da die kleinere Gruppe bei seinem Auftreten nur eine geringe Rolle spielt, kann er ungestört die Macht übernehmen. Sobald aber die erste passende Strategie als Mitglied der gegnerischen Gruppe entsteht, kann die Bevölkerung sich nicht schnell genug anpassen und das Machtverhältnis dreht sich um. Der erwartete Effekt, dass Zusammenbruch der Kooperation in der größeren Gruppe zu einem Machtwechsel führt, kann ebenfalls eine wichtige Rolle spielen, meist in Verbindung mit dem bereits beschriebenen Trojanisches-Pferd-Effekt.

Auch wenn meist eine Gruppe dominiert, ist Koexistenz beider Gruppen durchaus möglich. Dies ist der Fall, wenn jede der beiden im Durchschnitt mehr von der jeweils anderen profitiert als von sich selbst. Dadurch, dass beiden durch ihr eigenes Wachstum geschadet wird, entsteht ein stabiles Equilibrium. Diese Phasen treten deutlich seltener auf als solche, in denen nur eine Gruppe dominiert, sie sind aber ähnlich stabil. Sie enden meist dann, wenn Kooperation in einer der Gruppen aufkommt, seltener aber auch durch den Zusammenbruch der Kooperation zwischen den Gruppen.

Im Hinblick auf den Gesamtgewinn der Bevölkerung besteht ein klarer Unterschied zwischen Phasen, in denen eine Gruppe mehr als 95% der Bevölkerung einnimmt, und solchen, in denen beide Gruppen in relevanten Größen koexistieren. Homogene Populationen erwirtschaften dabei im Durchschnitt deutlich mehr Gewinn als heterogene (siehe Abbildung 5).

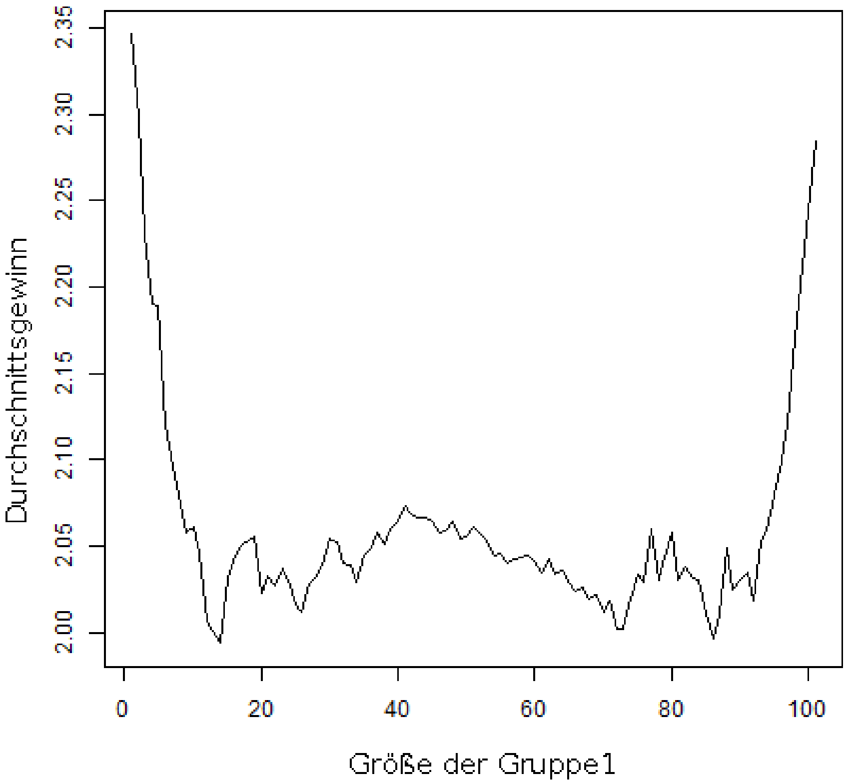

Abbildung 5: Durchschnittlicher Gesamtgewinn der Bevölkerung bei verschiedenen Verteilungen unter den Gruppen. Es ist ein klarer Anstieg gegen die Extremfälle Größe der Gruppe 1 ist 0, und Größe der Gruppe 1 ist 100 erkennbar.

Darstellung der Ergebnisse einer Simulation

Im Folgenden wird das Ergebnis eines Durchlaufes der Simulation anhand von Abbildung 6 erklärt.

Der erste Graph zeigt den Anteil der ersten Gruppe an der Gesamtbevölkerung, der zweite den Durchschnittsgewinn der Bevölkerung. Die restlichen Graphen sind die Durchschnitte der p_is innerhalb der beiden Gruppen. Die oberen beiden zeigen das Verhalten der Gruppe 1 zu erstens: anderen der Gruppe 1, und zweitens: jenen der Gruppe 2. Ebenso zeigt bei den unteren beiden der obere das Verhalten der Gruppe 2 gegenüber der Gruppe 1 und der untere letztendlich ihr Verhalten gegenüber sich

selbst.
Zu Beginn des Durchlaufes kann sich die Gruppe 1 mit den Strategien WSLS gegen sich selbst und GRIM gegen die andere Gruppe durchsetzen. Im Laufe der nächsten 3000 Generationen steigt der p_4 Wert (Kooperation nach P) langsam an, während gleichzeitig der p_1 Wert sinkt – bis die Strategie nicht mehr in der Lage ist, gegen sich selbst einen mindestens gleich hohen Gewinn zu erzielen wie jedmögliche andere Strategie gegen es erzielt. An diesem Punkt (1) übernimmt eine AllD-ähnliche Strategie innerhalb der Gruppe 1 die Überhand, was sich in einem starken Einbruch des Gesamtgewinns äußert. Dadurch, dass die jetzt vorherrschende Strategie zur Gruppe 2 (zufälligerweise) eine Variation von GTFT spielt, erfährt diese sofort massives Wachstum. Sie spielt mit sich selbst allerdings nur GRIM, welches einen geringeren Durchschnittsgewinn erreicht als das GTFT, das zwischen den Gruppen besteht. Deshalb kann sie nicht selbst Dominanz etablieren, stattdessen beginnt eine kurze Phase der Koexistenz. Die Gruppe 1 erfährt dabei aufgrund ihrer geringen Größe starke zufällige Fluktuationen, was sie kurzzeitig aussterben lässt, ihr dann aber erlaubt, die unoptimierte Gruppe 2 vollkommen zu verdrängen (2). Erneut dominiert die Gruppe 1, diesmal mit GTFT gegenüber sich selbst, und GRIM gegen die andere Gruppe. Aufgrund der geringen Größe der Gruppe 2 nähert sich das Verhalten der dominanten Strategie ihr gegenüber der zufälligen Strategie RANDOM (0.5,0.5,0.5,0.5) an, woraufhin diese kurzzeitig die Überhand gewinnt (3). Es folgt eine chaotische Phase mit sehr geringem Gesamtgewinn, aus der schließlich Gruppe 2 siegreich hervorgeht (4). Deren Strategie erfüllt alle Bedingungen für Dominanz, fällt aber dem oben beschriebenen Trojanisches-Pferd-Effekt zu Opfer: In ihrem Inneren ersetzt die neu aufgekommene Strategie GRIM die zuvor vorherrschende geringfügig kooperationswillige. Diese kann aber von der anderen Gruppe sehr leicht ausgebeutet werden, weshalb wieder die Gruppe 1 übernimmt (5). Die Simulation endet mit einem starken Einbruch der Kooperation innerhalb der dominanten Gruppe, der höchstwahrscheinlich zu einem erneuten Machtwechsel führen würde.

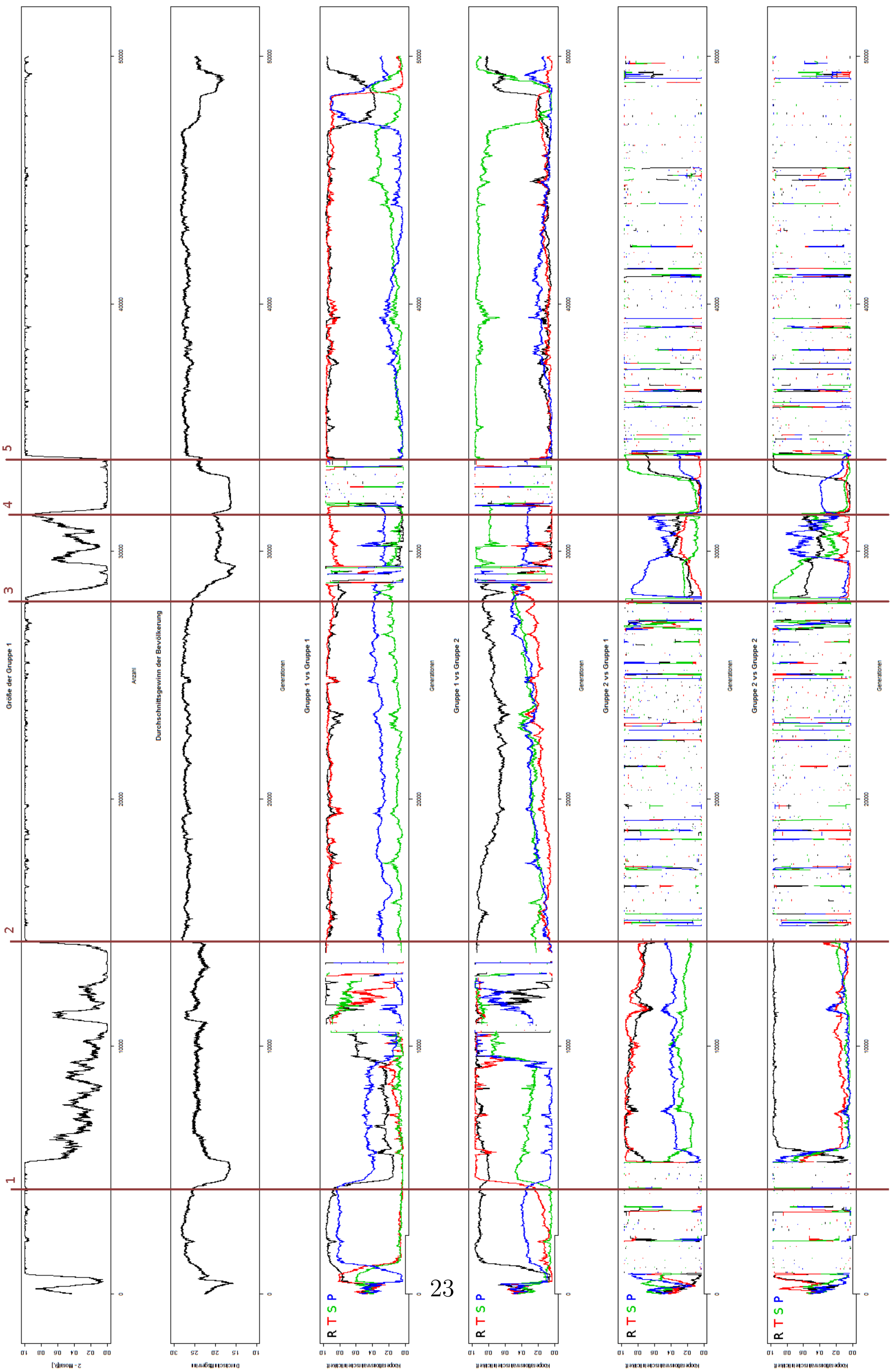

Abbildung 6: Ein Durchlauf der Simulation über 50 000 Generationen.

2.2.2 Vergleich zum normalen Prisoner's Dilemma

Zum Vergleich wurde die exakt selbe Simulation nochmals durchgeführt, allerdings wurden alle Spieler automatisch der ersten Gruppe zugeteilt. Statt GTFT dominierte hier sehr häufig GRIM und stabile Phasen dauerten teilweise das ganze Spiel an. Überraschend ist der Unterschied im Durchschnittsgewinn. Zu erwarten wäre gewesen, dass die Unterteilung in zwei Gruppen zu vermindertem Gewinn führen würde – tatsächlich aber ist der Durchschnittsgewinn in eben jenen mit $2,207$ deutlich größer als in den Populationen mit einer Gruppe, die nur $2,046$ erreichten. Auch ein t-Test[12] mit Signifikanzniveau 0,05 bestätigte die Hypothese, dass der Gewinn in Simulationen mit nur einer Gruppe geringer ist. Auch eine Verdopplung der Mutationsrate, um dem Wegfallen aller Mutation, die das Verhalten zur Gruppe 2 beeinflussen, entgegenzuwirken, führte nicht zu einer Erhöhung des Durchschnittsgewinns. Ausschlaggebend für den höheren Gewinn in Populationen, die in Gruppen unterteil sind, ist wohl die hinzugekommene Möglichkeit, non-kooperative Strategien effektiv zu bekämpfen, ohne selbst den eigenen Gewinn mit sich selbst zu beschränken. Während normalerweise selbst Strategien wie TFT in einem Spiel mit Rauschen gegen AllD einen gewissen Nachteil erleiden, kann das mit mehreren Gruppen vermieden werden. Denn ein Mitglied der kleineren Gruppe kann mit der eigenen Gruppe kooperativ spielen, und gleichzeitig gegen die AllD-lastige, größere Gruppe ebenfalls eine AllD-ähnliche Strategie wählen. Die Konkurrenz zwischen den beiden Gruppen fungiert also als natürlicher Abwehrmechanismus gegen kooperationsunwillige Strategien.

3 Fazit

Es hat sich gezeigt, dass im evolutionären wiederholten Prisoner's Dilemma mit zwei Gruppen die Population meist von einer der beiden Gruppen vollkommen dominiert wird. Die dominante Gruppe verhält sich in diesem Fall der anderen Gruppe gegenüber unkooperativ, hat mit sich selbst aber deutlich bessere Kooperationsraten

[12] Ein t-Test ist ein statistischer Test mit dem die Wahrscheinlichkeit berechnet wird, dass ein solcher Unterschied (also in der beobachteten Größe oder noch größer) bei Gruppen mit in Wahrheit gleichem Mittelwert zufällig beobachtet werden kann. Unterschreitet diese ein vorher bestimmtes Signifikanzniveau, spricht man von einem signifikanten Ergebnis.

als in der Gesamtpopulation von Spielen ohne Gruppen generell auftreten. Koexistenz beider Gruppen ist auch möglich. Diese ist aber nur dann stabil, wenn beide Gruppen mit sich selbst unkooperativ sind und führt daher zu verringertem Gesamtgewinn. Insgesamt erzielen in Gruppen unterteilte Population aber signifikant höhere Durchschnittsgewinne als solche ohne Gruppen. Weiterführend könnte untersucht werden, wie sich dieser Trend bei einer Erhöhung der Maximalanzahl an Gruppen fortsetzt.

Literatur

Axelrod, Robert. 1980. „Effective choice in the prisoner's dilemma“. Journal of conflict resolution 24 (1): 3–25.

Fundenberg, Drew, und Eric Maskin. 1990. „Evolution and Cooperation in Noisy Repeated Games“. The American Economic Review 80 (2): 274–79.

Hilbe, Christian, Martin A. Nowak, und Karl Sigmund. 2013. „Evolution of extortion in iterated prisoner's dilemma games“. Proceedings of the National Academy of Sciences 110 (17): 6913–6918.

Kollock, Peter. 1998. „Social dilemmas: The anatomy of cooperation“. Annual review of sociology, 183–214.

Nash, John. 1951. „Non-cooperative games“. Annals of mathematics, 286–295.

Nowak, Martin A., Akira Sasaki, Christine Taylor, und Drew Fudenberg. 2004. „Emergence of cooperation and evolutionary stability in finite populations“. Nature 428 (6983): 646–650.

Nowak, Martin A., und Karl Sigmund. 1992. „Tit for tat in heterogenous populations“. Nature 355 (6357): 250.

Nowak, Martin, und Karl Sigmund. 1990. „The evolution of stochastic strategies in the prisoner's dilemma“. Acta Applicandae Mathematicae 20 (3): 247–265.

Nowak, Martin, Karl Sigmund, und others. 1993. „A strategy of win-stay, lose-shift that outperforms tit-for-tat in the Prisoner's Dilemma game“. Nature 364 (6432): 56–58.

Tucker, A. W. 1983. „The Mathematics of Tucker: A Sampler“. The Two-Year College Mathematics Journal 14 (3): 228–32. doi:10.2307/3027092.

Anhang

Code der Simulation in R

```
a<-NULL
b<-NULL
Generationen<-50000
am<-1:100
Ertragsvektor<-NA
noise<-0.03
pay<-matrix(c(1,0,5,3),2,2)
yesno<-c(1,-1)
Population<-array(NA,c(10,100,Generationen+1))
Player<-0
for (o in 1:100) {
Player<-c(noise+(0.5+((1-runif(1)^3)/2)*sample(yesno,1))*(1-2*noise),
   noise+(0.5+((1-runif(1)^3)/2)*sample(yesno,1))*(1-2*noise),
   noise+(0.5+((1-runif(1)^3)/2)*sample(yesno,1))*(1-2*noise),
   noise+(0.5+((1-runif(1)^3)/2)*sample(yesno,1))*(1-2*noise),
   noise+(0.5+((1-runif(1)^3)/2)*sample(yesno,1))*(1-2*noise),
   noise+(0.5+((1-runif(1)^3)/2)*sample(yesno,1))*(1-2*noise),
   noise+(0.5+((1-runif(1)^3)/2)*sample(yesno,1))*(1-2*noise),
   noise+(0.5+((1-runif(1)^3)/2)*sample(yesno,1))*(1-2*noise),sample(1:2,1),0)
Population[,o,1]<-Player
}
for(g in 1:Generationen) {
got<-matrix(NA,3200,2)
for(w in 1:100)
{
for(q in 1:16)
{
Auswahl<-c(w,sample(am[-w],1))
pa<-c(Population[(1+4*(Population[9,Auswahl[2],g]-1)):
   (4+4*(Population[9,Auswahl[2],g]-1)),Auswahl[1],g])
pb<-c(Population[(1+4*(Population[9,Auswahl[1],g]-1)):
```

```
   (4+4*(Population[9,Auswahl[1],g]-1)),Auswahl[2],g])
xb<-1
xa<-1
for(i in 1:15) {
  A<-runif(1)<pa[xa]
  B<-runif(1)<pb[xb]
  xb<-4-2*A-B
  xa<-4-2*B-A
  a[i]<-pay[A+1,B+1]
  b[i]<-pay[B+1,A+1]
}
got[2*(16*(w-1)+q)-1,]<-c(mean(a),Auswahl[1])
got[2*(16*(w-1)+q),]<-c(mean(b),Auswahl[2])
}}
Ertragsvektor<-aggregate(got[,1],by=list(got[,2]),mean)
Population[10,,g]<-Ertragsvektor$x[order(Ertragsvektor$Group.1)]
Population[,,g+1]<-Population[,,g]
for(i in 1:3){
Population[,sample(1:100,prob=(5-Population[10,,g]),1),g+1]<-
apply(cbind(c(rep(noise,t=8),1,0),apply(cbind(
   c(rep(1-noise,t=8),2,200),
   Population[,sample(1:100,prob=Population[10,,g],1),g]+
   (runif(8)<0.05,0,0)*rnorm(10,s=0.05)),1,min)),1,max)
}
if(runif(1)<0.02){
Population[,sample(1:100,prob=(5-Population[10,,g]),1),g+1]<-
c(noise+(0.5+((1-runif(1)^3)/2)*sample(yesno,1))*(1-2*noise),
   noise+(0.5+((1-runif(1)^3)/2)*sample(yesno,1))*(1-2*noise),
   noise+(0.5+((1-runif(1)^3)/2)*sample(yesno,1))*(1-2*noise),
   noise+(0.5+((1-runif(1)^3)/2)*sample(yesno,1))*(1-2*noise),
   noise+(0.5+((1-runif(1)^3)/2)*sample(yesno,1))*(1-2*noise),
   noise+(0.5+((1-runif(1)^3)/2)*sample(yesno,1))*(1-2*noise),
   noise+(0.5+((1-runif(1)^3)/2)*sample(yesno,1))*(1-2*noise),
   noise+(0.5+((1-runif(1)^3)/2)*sample(yesno,1))*(1-2*noise),sample(1:2,1),0)
}}
```