Dissertation

submitted to the
Combined Faculties for the Natural Sciences and for Mathematics
of the Ruperto-Carola University of Heidelberg, Germany

for the degree of
Doctor of Natural Sciences

Put forward by
M.Sc. Benjamin Herdeanu
born in Helmstedt

Oral examination: November 3, 2021

Emergence of Cooperation in Evolutionary Social Interaction Networks

Referees:

Prof. Dr. Kurt Roth
Prof. Dr. Björn Malte Schäfer

Emergence of Cooperation in Evolutionary Social Interaction Networks Cooperation is the cornerstone of life and human societies—its evolution is a perennial question. Evolutionary Game Theory models elucidated mechanisms promoting cooperation. However, they pay little attention to its emergence, operate with predefined states of defectors and cooperators, and presume defection as the natural state. Models need more realism. Here, I introduce *ReCooDy*, a model combining population dynamics, limited resources, dynamic networks, coevolution of nine parameters, and optional social dilemma interactions, all implemented in the *Utopia* framework, which allows performant, flexible, and reliable computer simulation. *ReCooDy* investigates the emergence of cooperation and defection in a generalized continuous public goods interaction that incorporates "true defection"—exploitative destruction for selfish benefits. I show that macroscopically classifying agents as selfish or selfless oversimplifies the intricacy of the emerging mesoscopic social dilemma. Simulations exhibit the emergence of cooperation, even for minute synergies if interacting is not crucial. For moderate synergy, agents evolve specializations that depend vitally on the interactions. Further, defection emerges naturally as a response to cooperation. *ReCooDy* exhibits recurrent dynamics patterns with history dependence and frequent strategy collapses through cascaded thresholds caused by Red Queen dynamics. Thus, simulations reveal *ReCooDy*'s deterministically chaotic self-organized nature and the overall unintuitive phenomenology of more realistic social dynamics modeling.

Emergenz von Kooperation in evolutionären, sozialen Interaktionsnetzwerken Kooperation ist der Grundstein des Lebens und menschlicher Gesellschaften – ihre Evolution ist eine immerwährende Frage. Evolutionäre Spieltheorie-Modelle haben Mechanismen zur Verstärkung von Kooperation aufgezeigt. Sie berücksichtigen jedoch kaum die Emergenz von Kooperation, arbeiten mit vordefinierten „Defekteur“ und „Kooperateur“ Zuständen und gehen davon aus, dass Defektion der natürliche Zustand sei. Modelle brauchen mehr Realismus. In dieser Arbeit stelle ich *ReCooDy* vor – ein Modell, das Populationsdynamik, begrenzte Ressourcen, dynamische Netzwerke, Koevolution von neun Parametern und optionale soziale Dilemmainteraktionen kombiniert. Es ist im *Utopia*-Framework implementiert, das performante, flexible und zuverlässige Computersimulation ermöglicht. *ReCooDy* untersucht die Entstehung von Kooperation und Defektion in einem verallgemeinerten, kontinuierlichen Öffentliche Güter Spiel, das „wahre Defektion“ beinhaltet – ausbeuterische Zerstörung für egoistische Vorteile. Ich zeige, dass die makroskopische Agentenklassifizierung als egoistisch oder selbstlos die Komplexität des entstehenden mesoskopischen sozialen Dilemmas zu sehr vereinfacht. Simulationen zeigen die Entstehung von Kooperation, selbst für geringe Synergien, solange Interaktionen nicht lebensnotwendig sind. Bei moderaten Synergien entwickeln Agenten Spezialisierungen, die entscheidend von Interaktionen abhängen. Defektion entsteht hierbei als natürliche Folge auf Kooperation. *ReCooDy* zeigt wiederkehrende, geschichtsabhängige dynamische Strukturen und häufige kaskadierte Strategiezusammenbrüche durch Überschreitung von Schwellenwerten, getrieben durch eine „Red Queen“-Dynamik. Damit zeigen Simulationen die deterministisch chaotische, selbstorganisierte Natur von *ReCooDy* und die insgesamt unintuitive Phänomenologie einer realistischeren Modellierung sozialer Dynamiken.

Contents

1 **Introduction** 11

2 **Background** 15
2.1 Evolutionary Game Theory and the Evolution of Cooperation 15
2.1.1 Evolutionary Game Theory 15
2.1.2 The Game 16
2.1.2.1 Symmetric Two-Player Games 16
2.1.2.2 Tragedy of the Commons and the Public Goods Game . . 16
2.1.3 Evolutionary Update 17
2.1.4 Cooperation in Nature and Evolutionary Game Theory Applications 18
2.1.5 Continuous Games 19
2.1.6 Spatiality 19
2.1.7 Static Networks 19
2.1.8 Dynamic Network 20
2.1.9 Link Inheritance 20
2.1.10 Eco-Evolutionary Games 20
2.2 Network Science 20
2.2.1 Networks and Graphs 21
2.2.2 Properties 21
2.2.3 Generating Algorithms 22
2.3 Evolution Mechanics 23

3 **Utopia: A Comprehensive Modeling Framework for Complex and Evolving Systems** 25
3.1 Motivation 25
3.2 Features 27
3.2.1 Conceptualizing a Research Question 27
3.2.2 Implementing a Model 27
3.2.3 Simulating a Model and Generating Data 29
3.2.4 Evaluating Data 30
3.2.5 Testing, Review, and Workflow 34
3.3 Applications and Experience 34
3.4 Summary 35

4 ***ReCooDy*: The Resource-Flow-Based Cooperation Dynamics Model** 37
4.1 Structure and Concepts 37
4.2 Resource Extraction 40
4.2.1 Basic Resource Extraction 41
4.2.2 Synergistic Resource 42
4.2.2.1 Creating Goods 44

4.2.2.2 Creating Bads . . . 45
4.2.2.3 Unify Creation and Destruction . . . 47
4.2.2.4 Classification . . . 49
4.2.2.5 Finite Resources . . . 51
4.3 Living . . . 52
4.4 Linking . . . 52
4.4.1 Adding Links . . . 53
4.4.2 Removing Links . . . 54
4.4.3 Summary and Discussion . . . 54
4.5 Death . . . 56
4.6 Birth . . . 57
4.7 Initialization . . . 58
4.8 Summary, Storylines, and Discussion . . . 61

5 The Social Dilemma **67**
5.1 Tragedy of the Commons . . . 67
5.2 Personal Dilemma . . . 68
5.3 Competition Dilemma . . . 69
5.4 Instructive Example . . . 71
5.5 Offspring Benefit . . . 71
5.6 Summary and Discussion . . . 73

6 Evolution of Cooperation and Defection **77**
6.1 Overview . . . 77
6.2 Emergence of Cooperation . . . 80
6.2.1 Emergence of Infrequent Cooperation in Sparse Effective Networks 82
6.2.2 Summary . . . 85
6.3 Transition Regime . . . 86
6.3.1 Rich Dynamics . . . 88
6.3.2 Dynamical Phases and their Transitions . . . 91
6.3.3 Long-term Evolution – Defective Attractor . . . 99
6.3.3.1 Trigger Towards Growing Defection . . . 102
6.3.3.2 Eventual Collapse into Infrequent Defection . . . 106
6.3.4 Summary and Discussion . . . 109
6.4 High Synergy Regime . . . 112
6.4.1 Dynamical Phases and their Transitions . . . 114
6.4.2 Long-Term Evolution . . . 115
6.4.3 Summary and Discussion . . . 123

7 Summary and Discussion **125**

8 Conclusion and Outlook **133**

Acknowledgements **135**

Appendices **137**

A Supplementary Figures **139**

B List of Figures & Tables **151**
B.1 List of Figures 151
B.2 List of Tables 158

Bibliography of Own Publications **159**

Bibliography **161**

Symbols **175**

Index & Acronyms **179**

1 Introduction

> "Scientific advances often come from uncovering a hitherto unseen aspect of things as a result, not so much of using some new instrument, but rather from looking at objects from a different angle. This look is necessarily guided by a certain idea of what the so-called reality might be."
>
> *François Jacob* (1977)

Our world, with all of its observable complexity, is the result of a great unfolding. It autonomously generates hierarchical structures, which are ubiquitous, observable in physical, biological, social, and cultural systems (*Simon* 1962). They emerge from evolutionary and complex systems. Evolution is not restricted to biological systems but entails a much broader scope, including our cultural (*Boyd and Richerson* 2005; *Richerson* et al. 2016) and technological world (*Taylor and Dorin* 2020). Biological evolution is the most prominent and the best-understood system – because of its long history of unfolding over a few billion years and its long research history. Humankind's cultural and emerging technological (r)evolution operate on top of our biological world. However, today more than ever, the biological world is intertwined with our cultural and technological one, even inseparable. The rate at which our world is changing is unprecedented, with unseen magnitude across scales and massive feedbacks between our societies, life, and the environment.

Early in Earth's history, some 4Gy ago (*Dodd* et al. 2017), the first living beings—replicators emerging in an RNA world—came into existence (*Szathmáry and Maynard Smith* 1997). Biological evolution started shaping organisms through variation, natural selection, and inheritance as *Darwin* (1859) found in his seminal book "On the origin of species by means of natural selection, or the preservation of favoured races in the struggle for life". Survival of the fittest[1] is, " perhaps, the one profound idea in science that we can all readily understand" (*Maynard Smith and Szathmáry* 1999, p. 1). Evolution works as a sequence of gradually accumulated small changes as well as recombination and adaption of already existing inventions best described as tinkering (*Jacob* 1977). It contrasts the engineer's workflow of having a specific aim for which a specific solution is tailored and for which a new start from the beginning is always possible. Although, with increasing complexity also the engineer becomes a tinkerer (*Solé and Valverde* 2020). Importantly, evolution does not operate in a vacuum but is strongly embedded in the biotic and abiotic environment. Thus, modern views on evolution such as evolutionary connectionism (*Watson* et al. 2016) emphasize feedback mechanisms with the ecology

[1] *Spencer* (1866, p. 444) coined the phrase "Survival of the fittest", which concisely sums up Darwin's theory.

(eco-evo dynamics) (*Schoener* 2011; *Post and Palkovacs* 2009; *Turcotte* et al. 2011), the developmental organization (eco-devo), and the transition in individuality (evo-ego) (*Watson* et al. 2016; *Watson and Szathmáry* 2016). Recognizing that evolutionary and ecological time scales overlap (*Murugan* et al. 2021) and that human impact on both is enormous (*Steffen* et al. 2015; *Steffen* et al. 2018; *Cavicchioli* et al. 2019) calls for a more comprehensive view on evolutionary systems required to understand and cope with humanity's big challenges such as climate change and loss in biodiversity.

During our World's history, several major evolutionary transitions happened that had an enormous impact (*Maynard Smith and Szathmáry* 1995; *Szathmáry and Maynard Smith* 1995). Examples are the emergence of the genetic code, the eukaryotic cell, multicellularity, eusociality, and language. Hierarchically lower entities recombined and integrated into higher-level entities opening unseen levels of complexity as well as possibility (*Maynard Smith and Szathmáry* 1995; *Szathmáry and Maynard Smith* 1995; *Maynard Smith and Szathmáry* 1999; *Szathmáry* 2015). Such major evolutionary transitions cannot happen without groups of entities forming and maintaining themselves (*Bourke* 2011). Group formation itself requires cooperation. Indeed, looking at our World, we find cooperation to be ubiquitous. Not only within each hierarchical level of organization but also among them. Bacteria communicate and coordinate themselves via quorum sensing (*Whiteley* et al. 2017), cells cooperate to form multicellular organisms, plants communicate with other plants (*Yoneya and Takabayashi* 2014), microbes (*Wenke* et al. 2010), and animals (*Leonard and Francis* 2017), eusocial animals may transform into distributed superorganisms, at times with sophisticated agricultures (*Hölldobler and Wilson* 2010), humans coexist with their huge microbiome (*Sender* et al. 2016), and human societies exhibit previously unseen levels of cooperation (*Fehr and Fischbacher* 2003; *Perc* et al. 2017), the foundation of our cultural and technological progress. In short, "[a]ll of life is social" (*Frank* 2007).

The evolution of cooperation is a perennial question. In its strongest formulation, the question becomes: Why should entities pay an incurred cost to create a direct benefit for their apparent competitors in the survival of the fittest setting? Evolutionary Game Theory (EGT) elucidated the question of how cooperation evolves in evolutionary systems (*Nowak* 2006b; *Brown* 2016; *Friedman and Barry* 2016; *Newton* 2018) and was first introduced by *Lewontin* (1961), *Maynard Smith* (1972), and *Maynard Smith and Price* (1973). EGT investigates evolutionary settings, in which the outcome of an individual depends not only on its actions (strategy) but also on the strategies of others. Over the years, a plethora of EGT research emerged investigating the evolution of cooperation in diverse fields such as biology, psychology, sociology, politics, economics, and virtual systems (*Friedman and Barry* 2016; *Perc* et al. 2017; *Nowak* 2006a; *Szabó and Fáth* 2007). On the most fundamental level, *Wilson and Wilson* (2007) concisely summarize the challenge and motivation underlying the existence of cooperation: "Selfishness beats altruism within groups. Altruistic groups beat selfish groups. Everything else is commentary".

Most EGT results come from simple models that raise the question of how much realism they represent and to what extent the obtained results apply to real-world scenarios. Already early research provides examples: The winning strategy in the repeated iterated prisoner's dilemma of Axelrod's tournament was tit-for-tat (*Axelrod and Hamilton* 1981). However, it loses against the Pavlovian win-stay, lose-shift strategy as soon as occasional errors are introduced (*Nowak and Sigmund* 1993). In real life, evolution does not operate by optimizing one or two dimensions. It provides a vast space of possibilities and opportunities from which a single realization pathway is followed[2]. However, in EGT modeling, the curse of dimensionality and the exploding system size, which we can expect, usually inhibit such endeavors. Evolution can happen rapidly, especially in changing ecologies (*Murugan* et al. 2021), which needs to be addressed adequately in modeling. Already simple changes in ecological assumptions such as abandoning the constant population size restriction through varying sized groups can yield different results in EGT: Coexistence of cooperation and defection in a public goods game (PGG) but extinction in a prisoner's dilemma (PD) (*Hauert* et al. 2006). *García and Traulsen* (2019) showed that increasing the strategy space, including arguably stupid strategies and letting the model, not the modeler, decide which path evolution can choose, yields unexpected results; Rare and counterintuitive strategies can have a significant effect on the outcome. Interactions are usually non-linear and non-trivial such that single parameters cannot always capture their complexity (van *Cleve and Akçay* 2014). More fundamentally, many questions in evolution and ecology cannot be answered by simple equations in structured populations (*Ibsen-Jensen* et al. 2015). Contrasting most modeling results, several experiments in behavioral economics show declining levels of cooperation through network structure in human networks, indicating that models are still missing realism (*Sánchez* 2018). In section 2.1, I present a more comprehensive range of examples showing current limitations. In general, researchers emphasize the importance of coevolution as it can lead to unexpected and unintuitive results that deepen our understanding (*Perc and Szolnoki* 2010; *McNamara* 2013; *Akçay* 2020); the too reductionist approach can be misleading.

Although "the evolution of cooperation is not the puzzle it used to be" as *Akçay* (2020) claims, a comprehensive understanding remains an open question (*Perc* et al. 2017), especially if we look at its sheer abundance most prominently in our human societies. Further, the emergence of cooperation out of the blue from a natural population is usually not addressed in models. Even if models let strategies evolve in continuous trait spaces as in *Doebeli* et al. (2004), they generally do not account for the evolutionary origin of the interaction structure itself. Furthermore, EGT models typically do not account for the evolutionary origin of defection. In EGT models, defection means not paying a cost but still profiting from a benefit. However, the word itself would actually imply a worse quality: the active destruction of goods for personal benefit. This I will call true defection. We could naturally assume the existence of such a strategy, for example, from parasitic exploitation of host systems. Still, the concept was not yet introduced in evolutionary games to the best of my knowledge. Furthermore, models usually do not include the agents' lifetime development and resource extraction and usage cycles, i.e., population dynamics. Also, there typically are unlimited resources available in the environment. Therefore, I introduce in this dissertation *ReCooDy*—the *R*esource-flow-based *Coo*peration *Dy*namics model—, which addresses all the mentioned issues with the aim of increasing the realism of models concerning the evolution of cooperation.

A more realistic modeling approach comes with several challenges. In contrast to most physical systems, social systems do not have known fundamental laws like, for instance, the Navier-Stokes equation for fluid dynamics. Agent-based computer models and their simulation become the primary research tool in such situations (*Adami* et al. 2016; *Macal*

[2] Ideally, such an open-ended evolution should be incorporated in modeling and simulating to allow for inventions and the creation of qualitatively new behavior. *Taylor and Dorin* (2020) and *Taylor* (2016) present approaches in that direction.

2016). However, assuring the correctness of the modeling results becomes even more demanding if results rely strongly on computer simulation models. There is a need for a comprehensive but flexible and reliable modeling framework as well as testing of research models. Still, it is a promising path towards a deeper understanding of the sophisticated, multilayered cooperation found in our societies nowadays, which, in general, may help us to address our future challenges.

2 Background

In this chapter, I will present a brief theoretical and conceptual background that is helpful to understand the concepts and results presented in this thesis. It is not a complete or comprehensive introduction of any field, which would not be reasonably realizable within the scope of this thesis and the amount of research and results within each research field. Instead, I focus on introducing the fundamental concepts, definitions, and foundational literature that form the conceptual scaffolding of this thesis. I open paths to the respective literature for the interested reader to dive deeper into the individual topics.

2.1 Evolutionary Game Theory and the Evolution of Cooperation

The outcome of one's actions depends on the actions of others. This simple idea captures the essence of game theory, the mathematical field that studies interactions between strategic actors, which von *Neumann* (1928) pioneered. *Lewontin* (1961) first applied game theory to evolutionary contexts by considering populations playing against nature with the aim of minimizing their extinction rates. However, only with *Maynard Smith* (1972) and *Maynard Smith and Price* (1973) Evolutionary Game Theory became popular, who used it to explain animal conflict and introduced the Evolutionary Stable Strategy (ESS). An ESS is stable over evolutionary time in the sense that another strategy cannot invade it. The new field of EGT let to an explosion of research soon after *Axelrod and Hamilton* (1981)'s famous tournament and the seminal book of *Maynard Smith* (1982).

2.1.1 Evolutionary Game Theory

EGT is a powerful tool that elucidated the question of how cooperation evolves (*Brown* 2016; *Nowak* 2006a; *Friedman and Barry* 2016). Typically, cooperation means that an agent pays a cost to create a benefit for others (*Nowak* 2006b). In the evolutionary context, the question arises of how cooperation can even exist and be stable if others are at the same time competitors. EGT is a huge and interdisciplinary research field applied in biology, psychology, sociology, economics, and virtual worlds to investigate social dilemma situations (*Friedman and Barry* 2016; *Perc* et al. 2017; *Nowak* 2006a). Research has discovered a multitude of mechanisms promoting the evolution of cooperation (*Nowak* 2006a) such as kin selection[1] (*Hamilton* 1964), group selection (*Wilson and Wilson* 2007),

[1] Kin and group selection, and more generally inclusive fitness theory and multilevel selection, have always been strongly disputed. (*Nowak* et al. 2010) reignited the debate explaining how eusociality, the initial cornerstone of Hamilton's rule (*Hamilton* 1964), can be explained by group selection alone, concluding that kin selection theory is not needed to explain altruistic behavior. The selfish-gene idea that humans are just vehicles for their genes, on which evolution solely operates (*Dawkins* 1976), would be wrong. Instead, evolution would operate on multiple hierarchical levels (*Wilson and Wilson* 2007), including cultural ones (*Boyd and Richerson* 1982; *Richerson* et al. 2016). It provoked a huge

direct reciprocity for repeated interactions (*Trivers* 1971; *Axelrod and Hamilton* 1981), indirect reciprocity (*Nowak and Sigmund* 1998), punishment (*Clutton-Brock and Parker* 1995; *Fehr and Gächter* 2002), altruistic punishment (*Fowler* 2005), volunteering (*Hauert* et al. 2002), social norms (*Hauert* et al. 2007), spatiality (*Nowak and May* 1992; *Hauert and Doebeli* 2004), and network structure (*Lieberman* et al. 2005; *Santos* et al. 2008; *Szabó and Fáth* 2007). Additionally, there are specialized mechanisms capable of explaining the evolution of cooperation, such as microbes that induce altruistic behavior in their hosts (*Lewin-Epstein* et al. 2017).

2.1.2 The Game

The most important element of an EGT model is the game itself. A plethora of games exist that model pair-wise social interactions as well as group interactions (*Szabó and Fáth* 2007; *Gintis* 2009). They typically model social dilemma situations, which individual games qualitatively capture.

2.1.2.1 Symmetric Two-Player Games

The archetypical symmetric two-player game is the prisoner's dilemma (PD), in which two actors interact through cooperation or defection (*Rapoport* et al. 1965). Usually, the illustrative example story deals with two prisoners that can cooperate by staying silent or defect by betraying the other in court. If both cooperate, they pay a cost but receive a Reward (R). If one defects, (s)he gets a higher Temptation reward (T) while the defector receives the so-called "Succer's" payoff (S). However, if both defect, they receive a Punishment (P). Thus, if $T > R > P > S$, we call the game a prisoner's dilemma (PD). Relating the four parameters in a different way results in changed game quality. The snowdrift game (SD) (also *chicken game* or *Hawk-Dove* game) describes situations in which cooperating always yields a personal profit while defecting when the other cooperates yields an even higher one. With both defecting, both lose. The typical story follows two opposite people shoveling through a snowdrift to clear the path or doing nothing. Thus, if $T > R > S > P$ we call the game a snowdrift game (SD)[2] (*Hauert and Doebeli* 2004). In a harmony game (HG), cooperation is the best strategy for both players as $R > S > P$ and $R > T > P$. Here, there is no social dilemma. In a coordination game (CG), two players are required to coordinate their strategy to create a valuable outcome for both. Evolutionary games can also contain more than two discrete strategies, such as in the rock-paper-scissors game (RPS). For a more comprehensive overview, I refer the interested reader to standard textbooks and reviews such as (*Szabó and Fáth* 2007; *Gintis* 2009; *Friedman and Barry* 2016).

2.1.2.2 Tragedy of the Commons and the Public Goods Game

The most prominent evolutionary game, which models group interactions, is the public goods game (PGG) (*Trivers* 1971). It mathematically captures the tragedy of the commons

community response (*Abbot* et al. 2011). Since then, the dispute continues with positions ranging from kin and group selection are equivalent or both needed (*Gardner* 2020; *Birch* 2017; *Gardner* 2015; *Birch and Okasha* 2015) to kin selection has no predictable power at all (van *Veelen* 2020; *Nowak* et al. 2017; van *Veelen* et al. 2017; *Allen and Nowak* 2016).

[2] Usually, the parameters have a different notation to match the respective storyline.

by *Hardin* (1968), also known as the free-rider problem. He coined the phrase and build on ideas first described in *Lloyd* (1833): Overuse of a common meadow by cattle destroys the common good leaving everyone emptyhanded. However, individuals have no incentive to reduce or keep their number of cattle constant. The situation is a social dilemma. In the public goods game (PGG), a population of N players interacts with each other. Each player can decide whether to cooperate and pay a cost to create a public good or not to pay a cost, i.e., defect. The accumulated costs are transformed into public goods by multiplying them with the synergy factor $1 < r < N$. Each player profits equally from the public goods. We can write down the payoff of the cooperators P_C and the defectors P_D when N_C agents cooperate and pay an equal cost c:

$$P_C = \frac{r \cdot N_C \cdot c}{N} - c \tag{2.1}$$

$$P_D = \frac{r \cdot N_C \cdot c}{N}. \tag{2.2}$$

We assure that the game models a social dilemma through the restriction $1 < r < N$. Otherwise, the game would generate no synergies, or a cooperator's share of public goods would exceed the paid cost eliminating the dilemma. From a selfish perspective, each player is tempted to minimize the personal cost, but if no one cooperates, nobody receives a payoff. The PGG and the PD share many defining features such that it is possible to mathematically transform them into each other (*Hauert and Szabó* 2003). Therefore, the PGG is sometimes called the N-person Prisoner's Dilemma.

2.1.3 Evolutionary Update

In evolutionary game theory, we typically focus on the frequency change of strategies over generations of agents. We use iterated versions of games, combine them with an evolutionary update mechanism and solve the replicator equation in the simplest settings (*Hofbauer and Sigmund* 1998). Evolutionary update rules are often simple birth-death, death-birth, or imitation rules that select agents randomly and update their strategies accordingly to the agent's fitness (*Nowak* 2006a). The payoff from the games determines the agent's fitness. For simplicity, models often presume weak selection, meaning that significant differences in agents' payoffs result in small fitness differences. It facilitates mathematical analysis. However, *Wu* et al. (2013) shows that although most EGT results rely on weak-selection, "[...] qualitative changes with changing selection intensity arise almost certainly in the case of a large number of strategies". *Adami* et al. (2016) conclude that for scenarios that are analytically not feasible, e.g., outside the weak selection limit, agent-based models and their simulation opens the door to predict outcomes still.

Under evolutionary update rules, both the PD and the PGG will yield entirely defective populations in unstructured population settings with random encounters. In such well-mixed systems, their ESS is a state of pure defection (*Hofbauer and Sigmund* 1998). However, as mentioned above, the introduction of more realism through, for example, spatiality, population structure, reciprocity, and the selection on the level of kin or groups can overcome a state of defection and let cooperation evolve (*Nowak* 2006b).

2.1.4 Cooperation in Nature and Evolutionary Game Theory Applications

Let us explore a few examples of cooperation in nature and how research uses EGT to encompass the question of its evolution. Vampire bats sharing food with their peers are a prominent example of cooperation in nature (*Wilkinson* 1990). The behavior can be explained through reciprocity, the promise of a future reward for an action, through models based on the iterated Prisoner's Dilemma. Reciprocity can be direct, coming from the former interaction partner, or indirect if future benefits come from another individual of the population. However, whether the simple iterated Prisoner's Dilemma is a valid representation of the actual behavior is questionable because they lack key characteristics such as continuous investments, the choice of interaction partners, and the integration of different qualities of cooperation as experiments indicate (*Carter and Wilkinson* 2015; *Carter* 2014). The model seems to lack realism.

Bacteria use quorum sensing to communicate and coordinate, which we can model as a PGG (*Whiteley* et al. 2017). They invest energy and resources to produce chemical signal molecules that trigger a collective behavior once a critical environmental concentration is exceeded. Cooperating bacteria produce chemicals while defective bacteria do not. In reality, such forms of cooperation can be embedded in a greater symbiotic environment as the example of the *Vibrio fischeri* bacterium and the Hawaiian bobtail squid, *Euprymna scolopes* shows (*McFall-Ngai* 2014). The bacteria live inside the squid and use quorum sensing to coordinate luminescence that, in return, helps the former with hunting at dawn by disguising him from its prey. Their coevolved interdependence building on top of this symbiosis reveals an astonishing sophistication (*Visick* et al. 2021).

Moreover, plants cooperate by communicating via molecules released into the air (*Baldwin and Schultz* 1983; *Farmer and Ryan* 1990). Also, researchers start addressing questions in plant ecology with EGT (*Mcnickle and Dybzinski* 2013). Ants and other eusocial organisms form cooperative superorganisms that rely on cooperation to survive as a whole at times with symbiotic fungi and bacteria relations (*Hölldobler and Wilson* 1990; *Hölldobler and Wilson* 2010). Researchers also interpret cancerous cells in multicellular organisms as defectors in an overall cooperative multicellular organism (*Michor* et al. 2004; *Axelrod* et al. 2006; *Aktipis* 2016; *Bozic and Nowak* 2013).

Human cooperation is unique (*Fehr and Fischbacher* 2003) in its scale and sophistication. It is the foundation of our societies and the basis of the cultural and technological (r)evolution. Our language, which enables us to communicate and transfer concepts efficiently (*Pinker* 1994), is a profoundly cooperative faculty. *Perc* et al. (2017) provides an overview of how EGT models are used to explain human cooperation. However, for cooperation among humans, it is not clear to what extent the existing results obtained by simple EGT models accurately describe reality. There is a gap between theoretical models and experimental results from simple experiments in behavioral economics (*Sánchez* 2018). These experiments indicate that evolutionary games played on networks inhibit rather than promote cooperative behavior (*Cassar* 2007; *Kirchkamp and Nagel* 2007; *Gracia-Lázaro* et al. 2012; *Grujíc* et al. 2010) (see section 2.1.7 and section 2.1.8 for network models). However, *Rand* et al. (2014) experimentally found that a static network can promote the evolution of cooperation and *Fowler and Christakis* (2009) found cascades of cooperation in social networks. EGT models most probably do not yet incorporate the complexity and realm needed to adequately represent human cooperation with all its different layers of sophistication.

2.1.5 Continuous Games

It is natural to assume continuous strategy spaces in continuous games (*Doebeli and Hauert* 2005). *Killingback* et al. (1999) introduced a continuous version of the iterated PD and *Doebeli* et al. (2004) a continuous SD game. The strategy update can depend in simple settings on the partner's previous move (*Killingback* et al. 1999) or previous payoffs (*Doebeli and Knowlton* 1998; *Scheuring* 2005). In *Wahl and Nowak* (1999) costs evolve according to small mutations of successful strategies in a continuous iterated PD showing that in the long run, the dilemma "is either characterized by unending cycles or by stable polymorphisms of cooperators and defectors". *Doebeli* et al. (2004) use a continuous SD modified to incorporate costly investments that benefit both the defectors as well as the cooperators and show that cooperators and defectors can evolve to coexist in such a setting. *McNamara* et al. (2008) use continuous versions of both the PD and the SD to show that the lifespan of players can be important when cooperation and choosiness coevolve. *McGill and Brown* (2007) offers a broader review of games with continuous traits, and *McNamara* (2013) emphasizes the importance of using continuous strategies in contrast to discrete ones.

2.1.6 Spatiality

Nowak and May (1992) was the first to introduce a form of spatiality to EGT. He placed players on a two-dimensional grid and let each player interact with its neighbors. This system self-organized into kaleidoscope patterns of changing cooperators and defectors and showed that spatiality could result in the evolution of cooperation. Later, *Hauert and Doebeli* (2004) showed that spatiality does not always promote cooperation by investigating a spatial SD game. A multitude of models relies on spatiality as review such as *Perc* et al. (2017) make clear.

2.1.7 Static Networks

More general, networks are used to model population structure (*Jackson* et al. 2015; *Szabó and Fáth* 2007) and simple representations of space. Network science itself is an increasingly important research field with a huge range of applications (see section 2.2). Its combination with EGT led to completely new insights. In EGT models, networks usually promote the evolution of cooperation (*Battiston* et al. 2020). *Lieberman* et al. (2005) introduced the evolutionary graph theory in which they focussed on two-player games played on networks, in which nodes define players and edges define the interaction neighbors. They found that the graph structure can entirely determine the game dynamics. More recently, *Allen* et al. (2017) found that for a PD game under weak selection, strong pair-wise links between players promote the evolution of cooperation. On networks, two-player games and group games can yield different results. (*Szolnoki* et al. 2009) showed that the indirectly connected next-neighborhood of an agent, connected through group interactions, can have a significant impact. Further research showed that heterogeneous populations modeled through scale-free networks enhance the evolution of cooperation (*Santos and Pacheco* 2005; *Gómez-Gardeñes* et al. 2007; *Santos* et al. 2008), especially if graphs are highly clustered, indicating that hubs and communities promote cooperation (*Assenza* et al. 2008; *Rong* et al. 2010). Multiplex networks that connect the same players through multiple layers of edges further promote the evolution of cooperation through correlated clusters of cooperators (*Wang* et al. 2015).

2.1.8 Dynamic Network

Models do not only cope with static networks but also with dynamic ones. Coevolving networks repeatedly show self-organization into scale-free topologies and self-organized critical states (*Gross and Blasius* 2008). Relinking undesired links to randomly selected new agents can produce topologies similar to small-world networks (*Eguíluz* et al. 2005). *Pacheco* et al. (2006) showed that relinking could change the effective game itself from a PD to a coordination game (CG), or from a SD to a harmony game (HG). Furthermore, removing bad links in combination with randomly created new links can promote the evolution of cooperation through spontaneously emerging multilevel selection (*Szolnoki and Perc* 2009a). *Akçay* (2018) distinguished between relinking locally within the neighborhood and globally within the population. The network and strategy coevolution results in cooperation undermining itself. He found that local links promote and global links inhibit cooperation. Still, the introduction of a linking cost could counteract and prevent the inhibitory effect of global linking.

2.1.9 Link Inheritance

Inheriting social links to offspring can create networks with properties found in human social, economic networks (*Jackson and Rogers* 2007; *Jackson* 2008) and animal worlds (*Ilany and Akçay* 2016). Combining social link inheritance with EGT via a PD shows that cooperation leads to highly connected prosperous networks while defection causes fragmented poor networks (*Cavaliere* et al. 2012). Models that implement strategy and network evolution as independent processes also show sophisticated phenomenology. (*Szolnoki and Perc* 2009b) showed that if agents accumulate links over their lifetime starting with a single link, mechanisms such as Red Queen dynamics and group selection emerge spontaneously, promoting cooperation. Also, allowing agents to optimize their centrality within a network yields strongly fluctuating network topologies and strategy change cascades (*Holme and Ghoshal* 2006).

2.1.10 Eco-Evolutionary Games

More recently, eco-evolutionary games and feedbacks have become increasingly important, also showing that already simple feedback models yield rich and diverse dynamics (*Hauert* et al. 2006; *Stewart and Plotkin* 2014; *Weitz* et al. 2016; *Szolnoki and Chen* 2017; *Tilman* et al. 2020; *Gokhale and Hauert* 2016; *Hauert* et al. 2019; *Wang and Fu* 2020).

2.2 Network Science

Networks are a ubiquitous tool used to investigate entities, their relationship, and their interactions. There is a vast amount of research fields applying networks ranging from chemical reaction networks, over biological, disease spread, social, and opinion dynamics networks to cultural and technological networks. *Barabási and Pósfai* (2016) and *Albert and Barabási* (2002) present an excellent introduction into network theory and the respective fields of application. For a more recent review with a more mathematical focus on network geometry see (*Boguñá* et al. 2021). Moreover, EGT models widely use networks to model players and their interaction structure. The seminal work of *Szabó and Fáth* (2007) provides an excellent overview and starting point into the field of EGT network models. More generally, cooperative models on complex networks exhibit a wide range of critical phenomena such as network structure phase transition, the emergence of single huge connected components, and self-organized criticality (*Dorogovtsev* et al. 2008).

2.2.1 Networks and Graphs

Mathematically, we represent networks through graphs and the underlying mathematical field of graph theory; Thus, we can define networks as graphs for which vertices (also called *nodes*) and edges (also called *links*) have attributes (*Barabási and Pósfai* 2016). In general, a graph $G = (V, E)$ consists of a set of vertices $V = \{1, 2, \cdots, N\}$ and a set of edges $E = (V, V)$ connecting two vertices. The graph has N vertices and L edges that can be directed, undirected, or bidirectional. In this thesis, we will only work with undirected networks. Therefore, we will focus on undirected graphs and networks, for which all edges are undirected in the following.

2.2.2 Properties

The degree k of a vertex is defined as the number of in-edges that target the vertex. In undirected graphs, this in-degree equals the number of edges leaving the vertex, usually referred to as out-degree. The total number of links is then given by

$$L = \frac{1}{2} \sum_{i=1}^{N} k_i. \tag{2.3}$$

Here, the factor comes in to prevent counting links twice. With it, we can calculate the average degree of an undirected graph:

$$\bar{k} = \frac{1}{N} \sum_{i=1}^{N} \overset{(2.3)}{=} \frac{2L}{N}. \tag{2.4}$$

If N_k vertices have the degree k, we can write down the probability that a randomly selected vertex has the degree k, i.e., the degree distribution:

$$p_k = \frac{N_k}{N}. \tag{2.5}$$

We note that the degree distribution is normalized ($\sum_{k=0}^{\infty} p_k = 1$). Now, we can calculate the average degree of the graph:

$$\bar{k} = \sum_{k=0}^{\infty} k p_k. \tag{2.6}$$

A path connects two vertices of a graph through a sequence of edges. The distance of two vertices is the shortest path of two vertices. A graph is connected if all its vertices are connected via paths. If a graph is not connected, it contains connected components. A connected component is the set of all possible vertices that are connected via paths without further connections to other vertices of the graph. Standard textbooks on graph or network theory such as *Barabási and Pósfai* (2016) offer comprehensive introductions and explanations of fundamental graph (network) properties. Here, we focussed on the most important definitions and concepts used throughout this thesis.

Many network properties exist that characterize the network's quantitative and qualitative topological structure. A few examples of quantitative measures are the average degree, the clustering, the betweenness, and network motifs. Often, researchers use these quantities in order to find extract properties that could yield topological explanations of observed system behavior. Probably the most renowned qualitative topological network properties are the small-world and scale-free properties. A network is called a small-world network if the usual distance L between two randomly selected vertices is proportional to the logarithm of the total number of nodes: $L \propto \log N$ (*Watts and Strogatz* 1998). If a network exhibits a degree distribution p_k that (asymptotically) follows a power-law distribution it is called a scale-free network (*Barabási and Albert* 1999). Even though many observable real-world networks are said to exhibit the scale-free property, more recently, researchers emphasize the importance of long-tails of the degree distribution rather than the scale-freeness (*Holme* 2019; *Stumpf and Porter* 2012).

2.2.3 Generating Algorithms

Typically, computational models of real-world systems are based on networks generated via graph generation algorithms structurally replicating key network properties. Such networks are, for example, population structure in agent-based models (*Macal* 2016). A huge variety of network-generating algorithms exist such that we will only mention the most prominent algorithms. Erdös-Rényi networks are random networks that have random connections between their vertices exhibiting a binomial degree distribution (*Erdös and Rényi* 1959). Random networks were first studied by *Solomonoff and Rapoport* (1951). Erdös-Rényi networks exhibit the small-world property. However, typically Watts-Strogatz network are the networks of choice when investigating the small-world property (*Watts and Strogatz* 1998). The generating algorithm can create entirely random networks as edge cases. Barabási-Albert networks are scale-free networks that are built starting from a small spawning network that grows via a preferential attachment mechanism, which determines how to add new vertices and edges. Networks can also be organized and generated in multiple layers of nodes and edges, so-called multilayer networks, that become more and more popular (*Hammoud and Kramer* 2020; *Kivelä* et al. 2014; *Boccaletti* et al. 2014), also in EGT modeling (*Wang* et al. 2015).

Most network-generating algorithms assume a construction process that usually is decoupled from the dynamics of the system that operates on top of the network. They produce static networks constructed with a generator that is presumed but not derived from the system dynamics. In contrast, network generating algorithms that implement social inheritancetake into account the history of the network (*Jackson and Rogers* 2007; *Jackson* 2008). Such social inheritance accounts for the evolutionary origin of network structure and through evolution's tinkering applied to networks replicates common real-world network properties *Solé and Valverde* (2020) and *Ilany and Akçay* (2016) such as modularity (*Solé and Valverde* 2008).

2.3 Evolution Mechanics

Evolution mechanics is the conceptual framework we developed collectively within our research group that summarizes our understanding of how evolution operates. I decided not to present my view of evolution mechanics but instead introduce Utopia, our collectively developed operational framework in chapter 3. Nevertheless, to understand the overarching conceptual embedding of the work presented in this dissertation, I refer the reader to (*Sevinchan* 2021) for an introduction to evolution mechanics from the perspective of the author. It is not required to follow and understand the work presented in this dissertation but offers a broader picture.

3 Utopia: A Comprehensive Modeling Framework for Complex and Evolving Systems

Utopia, the comprehensive open-source modeling framework presented in this chapter, is the result of the collaborative work of many contributors, most notably the initiators and main contributors to the project (in alphabetical ordering) Benjamin Herdeanu (myself), Harald Mack, Lukas Riedel, and Yunus Sevinchan[1].

The work presented in this chapter is based on the following publications, in which I contributed as coauthor:

- *Riedel* et al. (2020)
- *Sevinchan* et al. (2020a)
- *Sevinchan* et al. (2020b).

In the following, I present Utopia from my personal perspective.

3.1 Motivation

Research in complex and evolving systems strongly relies on computer models and simulations answering questions in physical, environmental, biological, economic, and socio-cultural worlds (*Holland* 2006; *Levin* 2003; *Perc* et al. 2017; *Jackson and Zenou* 2015; *Jackson* et al. 2015). Such systems often exhibit non-linearities, hierarchical structures, self-organization, emergence, and intricate dynamics due to coupled processes and overlapping time-scales (*Murugan* et al. 2021; *Goldenfeld and Woese* 2011; *Kauffman* 1993). Rapid advances in computational processing power make computer models ever more powerful research tools. Nowadays, standard tools for research in complex and evolving system are cellular automata (CAs) (*Chopard* et al. 2002; *Wolfram* 1983), agent based models (ABMs) (*Macal* 2016), and network models (*Boccaletti* et al. 2006; *Jackson and Zenou* 2015; *Albert and Barabási* 2002). They enable a fast, heuristic way to investigate systems when corresponding real-life systems are hard or impossible to investigate through lab or in-field experiments. When effective system parameters themselves or their exact values are unknown, computer models open a way to explore such systems. Researchers can quickly try mechanisms, explore different scenarios and hypotheses, and do sensitivity analyses via parameter sweeps.

Complicated models require complicated software that has to be reliable and performant. Complicated models usually consist of many interacting parts that all need to work properly as granular units as well as integrated as a whole. The more subparts a model entails,

[1] In the Acknowledgments, I list all contributors.

i.e., the more complicated it gets, the more difficult and time-consuming it becomes to implement it and test its units as well as its integrated functionality. Each model unit requires an implemented software representation and each model interaction requires an interaction interface of the software. Therefore complicated models lead to complicated software. Modern software engineering workflows encompass increasing complexity and scale through extensive review processes and comprehensive automated code testing. Although correctness and reproducibility are crucial to research, and testing scientific software is pivotal (*Kanewala and Bieman* 2014), software engineering workflows are seldomly applied to research software (*Storer* 2017). Large research software projects that exceed a critical size can only be continuously developed and maintained collectively. Successful collaborations rely on optimized communication. A common language and extensive documentation of the functionality and the corresponding code are required. Moreover, the code itself needs to be modularly structured into building blocks that can flexibly be recombined in order to be competitive, as evolution indicates. In other words, large software projects can only be realized through boosted group-level synergies (*Sevinchan* et al. 2020a).

All of the mentioned inspired us, the Utopia Developer Team, to collectively develop a comprehensive modeling framework for complex and evolving environmental systems: Utopia. Utopia is written in modern C++ for performance-critical functionality and Python, where usability and flexibility are required. Although several open-source modeling frameworks are readily available in various programming languages (*Cardinot* et al. 2019; *Masad and Kazil* 2015; *Vahdati* 2019), such as NetLogo (*Wilensky* 1999), in our perception, none encompasses all our needs for a valuable and versatile modeling framework: comprehensible, flexible, performant, modular, while being developed with high coding standards including review and extensive code testing. With Utopia, we aimed at developing a common conceptual modeling foundation and operational toolkit as well as a common modeling language. We wanted to prevent redundant reimplementations and help users prevent common errors and pitfalls. Further, we designed Utopia to be easy-to-use for new users, allowing its application in teaching. We took advantage of group-level synergies to boost efficient and reliable collaborative research and teaching.

Utopia is a comprehensible framework helping users at all stages of our experienced computational research workflow:

1. conceptualizing a research question,

2. implementing a model,

3. simulating the model and generating data, and

4. evaluating data.

In our experience, all stages are intrinsically linked and need frequent re-iterations. Flexibility and adaptability are key. In the following, I will present an overview of Utopias feature set and explain how Utopia helps researches at each corresponding research workflow stage.

3.2 Features

Utopia incorporates a rich feature set, which is neither useful nor desirable to explore in its entirety within the scope of this thesis. Thus, I will only present an overview and invite the curious reader to visit Utopia's webpage (*Utopia Developer Team* 2021) and the linked repository on the webpage for more information. In the following, we will go through each phase of the mentioned complex and evolving research workflow and explore Utopia's features.

3.2.1 Conceptualizing a Research Question

Conceptualizing a research question requires researchers to create an abstracted representation of the reality, a computational model. Usually, we use and adapt common concepts, routines, and building blocks. Utopia helps users with their design decisions by providing a library of model-building functionality. A common language enables efficient communication within the user community as well as an efficient transfer of modeling concepts. Utopia increases a user's freedom to explore different modeling approaches and provides flexibility to investigate them through interchangeable building blocks and their human-friendly configurability. As a simple example, when conceptualizing the spatiality of a model, changing the underlying grid's periodicity, its neighborhood metric, or the metric distance, each requires a single configuration entry to be changed.

Models can be part of the framework itself as well as independently developed, with Utopia included as a dependency. Even more, Utopia can operate any computer model that provides compatible configuration and data input and output interfaces.

3.2.2 Implementing a Model

Utopia models are derived from a model base class that enables effortless framework integration and provides common functionality. It allows convenient access to the model configuration, the data output, and simulation monitoring functionality and provides the structural scaffolding to implement the model dynamics. The integration also enables the creation of supermodels that couple and control submodels. Utopia relies on several widely used software packages and serialization standards: Armadillo (*Sanderson and Curtin* 2016; *Sanderson and Curtin* 2018) for linear alegbra, Boost Graph for networks and Boost Test for advanced testing (*Siek* et al. 2002), the YAML serialization standard (*Ben-Kiki* et al. 2009) for human-friendly configurations, and HDF5 (*The HDF Group* 1997) for data writing. For the latter, we wrapped the desired C-library functionality into a C++ header-only library and integrated it into Utopia's `DataIO` module. Utopia's `core` library supplies functionality for recurring tasks in ABMs, CAs and network models. Examples range from setting up grids (square, hexagonal) with specific neighborhoods (von Neumann, Moore) over creating common graphs ((un)directed random, scale-free, small-world) and loading in real-world network data, to conveniently applying rules to entities.

Let us look at an instructive example of how Utopia provides convenient functionality and assists users in their modeling choices:

```
// Increment the age of all vertices in a graph
apply_rule<IterateOver::vertices, Update::async, Shuffle::off>(
    [](auto vertex, auto& graph) {       // The rule for a vertex
        ++graph[vertex].state.age;
    },
    graph                                // The graph
);
```

Listing 3.1: Example of a ruled applied to all vertices in a graph.

Listing 3.1 shows a code example, in which we apply the simple rule to increment the age of all vertices of a `graph`. Users can specify over which graph entities to iterate, namely vertices, (inverse) neighbors, or (in/out)edges. They need to choose whether the rule should be applied (a)synchronously, i.e., instantaneously to all entities or in a sequence. If `async` is chosen they have to specify whether the sequence of entities is randomized or not (`Shuffle::on/off`). Only in the first case, a random number generator object needs to be passed to the function. A correct function signature enforcing meaningful decisions is checked during compile time to prevent run-time overheads. The `apply_rule` concept is not only available for network models but also for grid-based CAs and ABMs. It facillitates a model transformation e.g. from a cellular automaton (CA) model to a network model. The example illustrates one of Utopia's design aims to navigate users through design decisions that each modeler needs to take: Does a rule represent an instantaneous event for all entities or rather a sequential one? Do I need to randomize the order in which to apply a rule? Utopia guides users by helping them make conscious design decisions and avoid common modeling pitfalls.

Getting to know a new modeling framework can be challenging, but explanatory template models, extensive documentation, and example models help to minimize hurdles. We can create a new model via a simple command line interface (CLI) command:

```
utopia models copy CopyMeGraph --new-name MyGraph
```

This short command creates the new `MyGraph` model by copying the `CopyMeGraph` blueprint model and sets up all the infrastructure and files needed to create and integrate a new model in Utopia. The model already contains a basic graph, exemplifies dynamic implementations, and most importantly, contains many explanatory comments that allow a learning-by-doing approach during model development. With the `copy` command, any existing Utopia model can conveniently be copied, not only the `CopyMeBare`, `CopyMeGraph`, `CopyMeGrid` blueprint models for bare-basic, grid-based, or graph-based models. Utopia ships with several implemented models that serve as potential starting points for individual research explorations. These range from the classic sand-pile model (*Bak* et al. 1987), over ecological models of forest-fires (*Bak* et al. 1990) as well as predator-prey dynamics, to contagious disease spread models, a simple spatial evolutionary games model, based and expanded on *Nowak and May* (1992), an opinion dynamics network model (*Deffuant* et al. 2000;

Hegselmann and Krause 2002), to mention a selection. These models showcase Utopia's features in actual use-case scenarios as well as Utopia's application spectrum. Moreover, Utopia's comprehensive documentation, including several guides, explains and exemplifies specific features. We aimed at minimizing the investment of getting to know Utopia to maximize efficient research with a powerful and flexible model development and simulation framework.

Utopia usually provides basic, easy-to-use functionality as well as advanced, more powerful, and flexible features. For example, when implementing data writing, model developers can choose between straightforward and easy-to-use manual data writing within a managed `write_data` function, or advanced, powerful writing capabilities with Utopia's `DataManager`. The first allows for simple data writing given by a write start time, an end time, and a step size. The latter enables dataset-specific configuration with more flexibility, e.g., via multiple customizable write intervals or custom state-specific write triggers.

3.2.3 Simulating a Model and Generating Data

Utopia's frontend manages simulation tasks, their hierarchical configuration, and automated evaluation through the `utopya` Python package. It aims at making simulation runs as easy as possible. For this, configurations have sensible default parameters on all levels that can easily be updated. Further, controlling Utopia is possible via a powerful command line interface (CLI) as well as interactive Python sessions. For example, if we want to run the `ReCooDy` model (introduced in chapter 4) via the CLI and have the data automatically evaluated we execute the following command:

```
utopia run ReCooDy
```

To update default configurations for specific model scenarios we can specify a path to a folder (here "emerging_cooperation") that contains a dedicated `run.yml` and `eval.yml` configuration pair:

```
utopia run ReCooDy --scenario emerging_cooperation
```

Utopia's holistic nature requires configurations on different hierarchical levels to allow for flexible simulation control. Configuration is based on the YAML serialization standard, which provides a hierarchical, dictionary-like human-readable way to configure. `utopya` combines the (i) `base`, (ii) `model`, (iii) `user/machine`, (iv) `run` and (v) `cli-update` sub-configurations and stores them in a single `meta` configuration. Starting from (i), next-higher levels can recursivly update previous configurations. This hierarchical frontend, model, and run configuration allows for flexible parameter settings. The frontend parses and recursively updates the frontend configuration, the model, and the specific run to allow for enhanced usability through flexibility.

Utopia's frontend distributes simulation tasks and manages their parallel execution in scenarios that involve parameter sweeps with distributed computation cluster support. Conceptually, we call a single model simulation run together with its configuration and output a universe run. A set of independent universes define a multiverse. Different universes within a multiverse cannot interact, but their simulation output can be evaluated collectively. Investigating complex systems often requires parameter sweeps/multiverse runs such as classifying regimes and analyzing the system's sensitivity. Utopia facilitates multidimensional parameter sweeps by integrating the `paramspace` Python package (*Sevinchan* 2020). By default, multiverse runs are executed in parallel, which is especially useful for simulations on distributed computation clusters.

During simulation, `utopya` monitors its progress, estimates the expected remaining simulation time, and efficiently computes and displays information available for simulation control. Users can configure their simulation to stop if a predefined condition is met, for example, if a population density drops below a threshold or the system reaches its fixed point. Simulation control can reduce the amount of unnecessary costly computations as well as save the user's valuable time.

3.2.4 Evaluating Data

Modern research often relies on generating and analyzing vast amounts of hierarchically structured and semantically heterogeneous data. Hierarchy can arise from modules within a model representation. Semantic heterogeneity means that data may consist of different data structures and types, e.g., multidimensional numerical data, configuration files, metadata, or raw data that only becomes meaningful after processing. These properties generally hamper a holistic approach for handling and automated processing data (*Sevinchan* et al. 2020b).

Still, comprehensive data evaluation is not impossible: We created the dedicated `dantro`[2] Python package for handling, transforming, and visualizing hierarchically structured and semantically heterogeneous data (*Sevinchan* et al. 2020b) and tightly integrated it into Utopia. Similar to the common continuous integration pipelines, `dantro` streamlines all predefined operations into a data processing pipeline: an automated, dynamically configurable operation sequence. This data processing pipelines conveniently integrates and combines functionality from various established Python packages such as `h5py` (*Collette* 2013), `numpy` (van der *Walt* et al. 2011), `xarray` (*Hoyer and Hamman* 2017), `dask` (*Dask Development Team* 2016; *Rocklin* 2015), `matplotlib` (*Hunter* 2007), and `seaborn` (*Waskom* 2021). It provides a general and flexible interface that simplifies interoperability and allows for custom pipeline specializations. Once the data processing pipeline is set up with a one-time overhead, `dantro` alleviates the need to interface with packages individually and enables data evaluation entirely through YAML configurations—no programming in Python is required.

`dantro` contains three main modules that represent the three stages of a data processing pipeline: the data tree, the data transformation framework, and the plotting framework. In the following, I will depict the basic functionality. More information on `dantro` and its features is available in *Sevinchan* et al. (2020b) and its extensive online documentation (*Dantro Developer Team* 2021).

The data tree structures data in a hierarchical tree. Data groups form tree nodes that themselves contain other data groups or data containers. These `dantro` groups and containers each share a common base class that provides a unified interface to handle and traverse the data tree. They are specialized for various data and content types, allowing data type-specific operations. `dantro` provides not only readily available group and

[2] The word "dantro" is a combination of "data" and "dentro", Greek for tree.

container specializations such as for `numpy`, `xarray`, time-series, graph, or grid data but also enables convenient custom specializations. The added level of structural abstraction coming with the data tree enables `dantro` to include generalized features such as lazy and out-of-memory calculations; Huge datasets become manageable even on personal computers.

The data transformation framework enables arbitrary configuration-based data transformations. `dantro` ships with predefined often-used data operations such as many Python build-in operations, operations on `numpy` arrays or `xarray` datasets and data arrays. Users can also specify operations, which in principle expands `dantro` 's functionality to the set of all Python operations. For maximal flexibility, operations are applicable to arbitrary objects within the data tree. Internally, `dantro` constructs a directed acyclic graph (DAG) of operations from the specified YAML configuration entries. It allows for automated file-based caching and reloading of transformation results that significantly reduces the computational time for large, costly data transformation recalculations. In summary, the data transformation framework allows users to transform data via configuration files without the need to write actual code and automatically integrates powerful features such as result caching.

The plotting framework enables convenient, configuration-based data visualization, the final step of the data processing pipeline. The `PlotManager` manages which plots to create, as well as the required handling and transformation of raw data. It provides a convenient way to inherit plot configurations and recursively combine configuration snippets, which ultimately provides a modularized building-block way of generating plots. Furthermore, the integration of the `paramspace` Python package (*Sevinchan* 2020) enables plot parameter sweeps that generate multiple plots for varying plot parameters. The actual visualization and backend management of the plot creation is done via `PlotCreators`[3]. Creators allow to set plot aesthetics via configuration entries and simplify the creation of animations. As an example, `dantro` wraps `xarray`'s facet grid plotting capabilities, makes it config-configurable, and extends it by providing the option to create an animation by adding an optional `frames` dimension. Another example of the plotting frameworks' capabilities is the `multiplot` function that provides a configuration-based interface to plot multiple plots into a figure. It supports all axis-level `matplotlib` and `seaborn` functions, and customized user-specific plot functions. Apart from the library-given plot functions, users can easily integrate custom plot functions into the plotting infrastructure. The plotting framework provides an integrated, powerful, modularized, and config-based way to visualize data.

[3] Currently, the focus lies on `matplotlib`-based plot creation. However, integrating other plotting frameworks such as `altair` (*VanderPlas* et al. 2018) with its declarative Vega-Lite grammar (*Satyanarayan* et al. 2017) would be possible, and the scaffolding already exists.

Showcase

Here, we look at a simple and comprehensive showcase to exemplify how data evaluation works in an actual use-case setting. We focus on the basic functionality and omit configuration details for clarity.

The first basic example creates an animation of a quantity (here, the agents' investments) dependent on another parameter (here, different synergy factors r) plotted column-wise with frames given by varying random number generator seeds. Single frames of the resulting animation are similar to Figure 6.1, but omit the color-encoding for simplicity; Instead, single data points are plotted over each other creating a scatter plot. The plot configuration is shown below in Listing 3.2.

```yaml
# Example eval.yml plot configuration file
investment_facets:
  based_on:                      # (1)
    - .dag.generic.facet_grid
    - .animation.ffmpeg

  creator: multiverse            # (2)

  select_and_combine:            # (3)
    fields:
      data:
        path: data/ReCooDy/d_investment

  x: agent                       # (4)
  y: d_investment
  col: r
  col_wrap: 3
  frames: seed
  linestyle: ''
  marker: .
```

Listing 3.2: Example plot configuration creating an animation of a colum-wise facetted scatter plot similar to Figure 6.1.

The `investment_facets` plot is based on the general configurations of the generic `facet_grid` plot with default animation settings (1). Both configuration snippets are readily available via `utopya`. However, users can update and overwrite settings directly in the plot configuration. The plot uses a multiverse plot creator (2). The creator selects for each universe the two dimensional data located in the 'data/ReCooDy/d_investment' path and combines it into a `xarray.DataArray` with the four dimensions: `agent`, `d_investment`, `r`, and `seed` (3). Note that the sweep dimensions are automatically combined and added. The plot creator passes the configuration options defining the data encoding and layout further on to the `facet_grid` plot function. We can visualize up to 6 data dimensions (here, just 4 for simplicity). Furthermore, `dantro` features automatic encoding of the plot kind and layout, providing shorter configuration files and even more flexibility.

The more comprehensive example configuration below in Listing 3.3 reproduces the right plot of Figure 6.2, a combination of a `seaborn` violinplot and stripplot that shows the distribution of final mean investments dependent on a synergy factor r for varying random number generator seeds.

```
# eval_multiplot.yml - violinplot & stripplot of final mean investments
final_mean_investments:
  based_on:               # (1)
    - style.thesis
    - dag.options.enable_caching

  creator: multiverse     # (2)

  module: dantro.plot_creators.ext_funcs.multiplot
  plot_func: multiplot    # (3)

  select_and_combine:     # (4)
    fields:
      combined_data:
        path: data/ReCooDy/investment_mean
        transform:    # Select last time step
          - .isel: [!dag_prev , {time: -1}]

  transform:              # (5)
    - .to_dataframe: [!dag_tag combined_data]
    - callattr: [!dag_prev , reset_index]
      tag: df

  to_plot:                # (6)
    - function: sns.violinplot
      data: !dag_result df
      inner: quartile
    - function: sns.stripplot
      data: !dag_result df
  x: r
  y: investment_mean

  helpers:                # (7)
    set_limits: {y: [-42, 42]}
```

Listing 3.3: Example plot configuration visualizing the final mean investments.

The plot is based on configuration snippets that contain options to set a matplotlib plot style (`style.thesis`) and enable caching, respectively (`dag.options.enable_caching`) (1). These options are combined via recursive updates and can be overwritten. User-defined and library-given configurations are selectable. The multiverse creator actually creates the plot, which takes and processes multiverse data (2). It calls the `multiverse` plot function provided in `dantro`'s multiplot module (3). The creator selects and combines the `investment_mean` data of all universes each with a unique r and `seed` parameter pair, selects only the last time step for each, and returns the data tagged `combined_data` (4). The data is stored in a dimension and coordinate labeled four-dimensional `xarray.DataArray` object, `utopya`'s default numerical data container type. We need to transform it into a `pandas.DataFrame` in order to call `seaborn` plot functions (5). The `seaborn` violinplot and a stripplot functions are called with the transformed data as input (6). We can add further keyword arguments directly below, exemplified with `inner`. The `x` and `y` plotting dimensions are shared and passed to both plot functions. Finally, we can change the figure aesthetics, such as setting the limit of the y axis, as done (7). Various plot helpers are provided that wrap `matplotlib` functions such as setting a title, axis labels, tick formators or locators, or adding horizontal or vertical lines to the figure.

Both examples use data evaluation scenarios presented in this thesis and exemplify how `utopya` and especially `dantro` work in production. Of course, the examples show only a fraction of the data evaluation possibilities. For more examples, readers are invited to visit Utopia's and `dantro`'s documentations (*Utopia Developer Team* 2021; *Dantro Developer Team* 2021).

3.2.5 Testing, Review, and Workflow

Testing scientific software is pivotal to guarantee correctness (*Kanewala and Bieman* 2014). Utopia uses GitLab's continuous integration testing pipeline to test existing and new code automatically. We put an emphasis not only on testing the framework itself but also on helping model developers test their code. Utopia integrates the Boost Test framework to facilitate C++ unit and integration tests of the framework itself and implemented models. To help to test macroscopic model behavior, Utopia provides functionality for Python-based model tests. Having the code base automatically tested proved utterly useful multiple times, preventing severe bugs.

To further increase code quality, code only becomes part of the framework after a successful review process. Utopia's development and maintenance is coordinated on a GitLab server. We use GitLab's rich infrastructure to enable open discussions, coordinate tasks, and carry out reviews. We aim to increase quality, efficiency, and productivity through cooperative synergies.

3.3 Applications and Experience

Utopia is a modeling framework developed both for research and teaching. Within our research group, more than 20 master's and bachelor's theses have been successfully conducted using Utopia as the sole computational research tool. Many students actively contributed to the codebase. Four Ph.D. students have actively developed and used Utopia in their projects with topics ranging from "the feedback between environment and evolving populations with CA and ABMs, the evolution of ecological interaction networks, the emergence of cooperation in dynamic social interaction networks [(this work)], and the development of geometric and polarity properties of the basilar papilla in agent-based vertex models" (*Riedel* et al. 2020). The M.Sc. level physics lecture on "chaotic, complex, and evolving environmental systems" at the Department of Physics and Astronomy, Heidelberg University, incorporated Utopia as a teaching tool in its exercises investigating complex systems. Furthermore, students explored existing models as well as developed and explored their own Utopia models, even with limited programming experience, in semesterly seminars on complex and evolving systems.

We designed Utopia to incorporate a good balance between usability and performance. On the user level, Utopia's operation should be simple and intuitive. Users should be able to operate unknown models in a black-box fashion without any knowledge of implementation details to gain an initial, playful intuition of the system. On the model developer level, the focus was on performance and flexibility. Our experiences, both in teaching and research, indicated that, in our assessment, we reached our initial design goal.

3.4 Summary

“Utopia provides the tools to conveniently implement computer models, perform simulation runs, and evaluate the resulting data.” (*Utopia Developer Team* 2021). Developing, maintaining, and using Utopia was a collective endeavor that fostered cooperation, helped spread best practices of software engineering, and in general, boosted synergies within our research group. Our work shows that it is possible to adopt modern software engineering practices in science; It fosters reusability, reproducibility, sustainability, and reliability of research software (*Sevinchan* et al. 2020a).

4 *ReCooDy*: The Resource-Flow-Based Cooperation Dynamics Model

The *R*esource-flow-based *Coo*peration *Dy*namics model—short *ReCooDy*—is a tool designed to investigate the emergence of cooperation in evolving populations of agents that need to extract resources, either independently or via high-risk high-reward interactions, to survive, procreate, and modify their social interaction structure. Agents are abstracted entities having internal traits that define their actions and internal states that accumulate outcomes of their actions. Their state depends on their traits, their social environment structure, and the actions of others. To represent this abstraction in our language, we will refer to an agent with the neutral pronoun *it* throughout this thesis. Due to their abstracted nature, agents do not necessarily represent one specific real-world entity; Instead, I structurally identify properties of real-world entities such as individuals, institutions, organizations, or groups of individuals in specific real-world systems and develop expectations reliant on the presence or absence of such process structures.

4.1 Structure and Concepts

ReCooDy is formulated and propagated as an iterated computer simulation model. It is implemented in the `Utopia` framework (see chapter 3). A model simulation has populations of agents, which can change over time. Rule functions that operate on all N_t agents of a population at a given time $t \in \mathbb{N}_{\geq 0}$ implement the individual processes. All processes are executed consecutively in each time step, determining the system state at the following time step $t + 1$. Figure 4.1 illustrates all implemented processes and their sequence. It serves as a concise summary and guideline to keep in mind throughout this chapter.

Agents live in an environment providing a limited amount of resources per time. A consistent resource-flow modeling is a central concept of *ReCooDy*. Resources flow into the system via resource extraction and dissipate as every action—surviving, procreating, interacting, and linking—requires resources. Agents either extract basic resources individually or synergistic resources collectively via synergistic interactions with their social environment. The former is a comparatively secure way of receiving resources, in which success is determined by an agent's (costly) strength trait. The latter is based on a high-risk, high-gain social dilemma interaction, which builds on a PGG, extends it to a continuous cost space, and incorporates public goods destruction through the notion of true defection. Cooperative strategies are prone to exploitation to more defective strategies while potentially yielding enormous resource gains. All resources an agent extracts accumulate into its internal resource reservoir. Agents with more resources have a higher survival chance, can create more offspring, can invest more, can take more risks, and can better optimize their interaction partners, i.e., their social environment.

Evolution is the integral process determining which strategy sets are competitive, thus, which system states are actually realized. In the real world, evolution operates within an

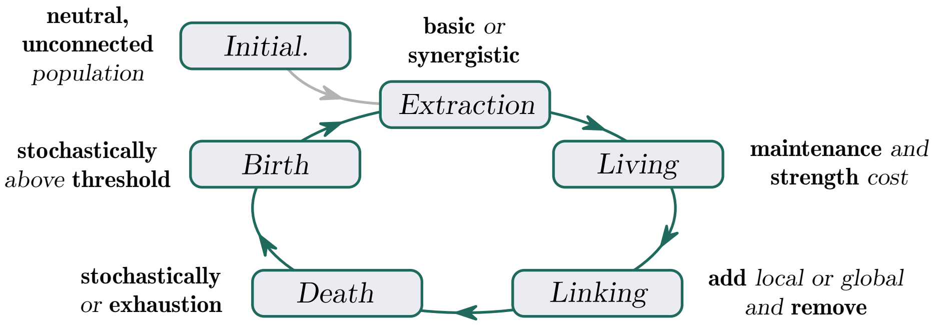

neutral, unconnected *population*
Initial.
basic *or* **synergistic**
Extraction
stochastically *above* **threshold**
Birth
Living
maintenance *and* **strength** *cost*
stochastically *or* **exhaustion**
Death
Linking
add *local or global* *and* **remove**

Figure 4.1: The *ReCooDy* process cycle. At the beginning of a simulation run, we initialize *ReCooDy* with a neutral, unconnected population and set up all relevant model parameters. Then, the actual process cycle starts. One cycle defines one model iteration and one time step. As an overview, I concisely summarize the model processes in the following, which I will, however, explain in detail within the corresponding sections of this chapter. First, agents either extract basic resources independently with probability p_b or, with probability $p_s = 1 - p_b$, synergistic resources through generalized public goods game (gPGG) interactions that exhibit social dilemma characteristics. Then, agents pay a constant cost of living c_l and a strength cost c_s. The evolving strength trait determines their success during basic resource extraction. They use their internal resources to link, i.e., they add or remove interaction connections to others to optimized their social environment. If their internally stored resources fall below a death threshold t_δ they die of exhaustion; They get removed from the population with all their links. Death also occurs randomly with probability p_δ per agent and time step. If their resources exceed a birth threshold t_β they create an offspring with probability p_β per agent and time step, pay a birth cost c_β and transfer resources j_β to their offspring. Afterward, the process cycle starts again with resource extraction.

exorbitantly high possibility space, in which not only parameters but rules themselves change over time[1] What we experience is merely a single realization that unfolds. We usually use fitness measures to quantify success in evolutionary systems, but more recently *Doebeli* et al. (2017) emphasize a more mechanistic approach fundamentally based on birth and death processes. EGT usually embraces simple death-birth, birth-death, or imitation processes, which generally speaking combine both processes into a single strategy update rule (e.g., see common textbooks such as *Nowak* (2006a) and *Friedman and Barry* (2016)). In contrast, *ReCooDy* decouples and separately implements birth and death processes. This separation softens arbitrarily strict modeling restrictions, e.g., replacing randomly dying agents without changes to population structure and neglecting any agent development. In *ReCooDy*, the agent's social environment develops over time by being able to add and remove links to other agents. Links are not a priori always socially inherited as, for example, done by *Ilany and Akçay* (2016). Nevertheless, social inheritance can still evolve indirectly in *ReCooDy* as agents are capable of evolving local link creation to their parents' neighbors. However, they could also evolve to form global links or no links at all. *ReCooDy* includes an individual development phase over an agent's lifetime in the form of constructing and optimizing its social interaction environment and separates birth and death processes from each other to increase realism.

The amount of resources flowing into the system needs to be limited to maintain computational feasibility and increase realism. Without limited resource inflows, we could easily imagine settings with higher birth rates than death rates, which would lead to more and more agents as well as diverging simulation times. With limited resource inflows, the model can only sustain a certain number of agents; Thus, it indirectly exhibits a maximal carrying capacity. The exact number of agents and the number of links depend on how agents evolve to extract and distribute their resources. Operationally, limited resources prevent diverging computational times and, conceptually, they increase the model's realism because real populations usually live in limited environments and need to share limited resources as simple logistic growth population models such as *Verhulst* (1838) already indicate.

Limited resources prevent the system from exploding, introduce more realism, and increase the evolutionary pressure on the agents living in a highly competitive and dynamically changing (social) environment. Limited resources raise the issue of distribution as soon as agents want to extract more resources than there are available resources. This distribution issue will automatically increase the competition, thus, the pressure to be successful. The main factor determining whether an agent will be successful is its strategies—the set of agent-specific traits resulting in a combination of comparatively high birth rates and low death rates. It contains information on how an agent will (stochastically) act within its environment. Its social environment will develop over time. We will see that specific local network configurations can be crucial because all processes include thresholds that determine whether an agent survives, procreates, links, and also how much it invests. With all of these in mind, we could expect that successful strategies potentially need to adapt over time in order to remain successful.

Before we dive into the model itself, let us shortly reflect on the language used in

[1] In recent years, new research paths embracing the idea of open-ended evolution emerged—letting parameters as well as the rules themselves evolve— which could be a promising way to deepen our understanding of how evolution operates and potentially boosting our technological advances (see for example (*Taylor and Dorin* 2020; *Adams* et al. 2017)).

this thesis. I will use anthropomorphic language elements that do not necessarily entail their complete corresponding set of connotations, which we would anticipate from our accumulated low-level world experience. Instead, they represent high-level analogs of the underlying low-level concepts. Consequently, words such as *cooperator*, *social environment*, or *resource* do not have a direct real-world meaning but need to be translated and projected to the respective structural real-world concept they represent. For example: Within the EGT high-level modeling context, a cooperator is usually condensed into a single defining property: paying a personal cost to create a benefit for others. Structurally, we can often identify a real-world low-level analog when looking at a specific scenario. However, usually the real-world situation entails much more complexity, is higher dimensional, and requires various qualities to describe it in its whole adequately. E.g., whether and to what extent an individual cooperates can depend on its phenotypically translated genetic predisposition, its personal history also concerning the interaction partners, whether it follows a bigger underlying plan, and so on. Further, dependent on the situation, time, and context, different people sense, recognize, and grade cooperative behavior differently; finding a standard all-entailing measure is complicated. These are just a few example lines of thoughts that we could follow even deeper. In short, we should be aware that the language elements and introduced respective concepts in this thesis are high-level representations of their low-level real-world analogs and, as such, neither entails all of their complexity nor necessarily all their semantic notions.

ReCooDy is a complicated and complex model; It incorporates several mechanistic processes and consists of many interacting agents highly coupled and dependent on each other through resource flows. The increased modeling complexity introduces more realism but comes with the severe challenge of mathematical feasibility. To the best of my knowledge, we do not yet have the adequate mathematical tools to extract a general understanding from a system as complex and complicated as *ReCooDy*. Instead, we currently need to apply a more explorative and heuristic way of extracting knowledge based on computer simulations, their observation, and deduced consistent explanations of these observations. All of which makes the investigation of *ReCooDy* a promising and exciting but challenging endeavor.

4.2 Resource Extraction

Agents extract resources to increase their internal resource reservoir R through one of two ways[2]: (i) independent extraction of basic resources $\mathfrak{R}_b$, or (ii) group extraction of a synergistic resource $\mathfrak{R}_s$. The probabilities p_b and $p_s := 1 - p_b$ determine whether to extract the basic or the synergistic resource, respectively, in each time step t for each agent[3]. They are evolving agent-specific traits, i.e., they get inherited with variation via small mutations from parent to offspring and selected through birth and death processes (see section 4.6 and section 4.5). Thus, each time step t, the agent population is split into two groups by sampling for each a a random number from a continuous uniform

[2] R is an agent-specific state, and therefore would require a subscript a denoting the agent: R_a. However, for elegance and visual noise reduction, I often omit the subscript a when referring to an agent state or trait, in general. Still, I use the full version, R_a, where it is helpful, useful, or required for better understanding.

[3] Throughout the thesis, a will refer to an agent $a \in N_t$ being part of the whole population at a given time t if not otherwise indicated. For example, R_a is the internal resource reservoir of an agent $a \in N_t$.

distribution $\mathcal{U}(0,1)$. If this number is smaller than or equal to the agent-specific probability p_b, agent a belongs to the set of agents that extract basic resources in that time step, $\mathcal{B}_t$. If the number is greater than p_b, agent a belongs to the set of agents that interact and extract synergistic resources in that time step, $\mathcal{S}_t$. During its lifetime, an agent a will extract the basic resource on average $A^{\text{final}} \cdot p_b$ times and the synergistic one $A^{\text{final}} \cdot p_s$ times with A^{final} being the agent's final age.

Both resource types add to the same internal reservoir R of an agent when extracted. Hence, we do not distinguish different resource types for simplicity but unify them into one internal resource reservoir R that provides the means of living and acting. If an agent is strong enough compared to others and extracts basic resources, it receives a fixed amount of basic resources added to its internal resource reservoir. Thus, the reward is only indirectly dependent on others. In contrast, when extracting synergistic resources via group interactions, the reward is directly dependent on others because of intrinsic high-risk, high-gain social dilemma game characteristics (see section 4.2.2).

Both resources have a limited amount available per time step that can sustain only a finite number of agents. There is a steady resource inflow into the system for which agents need to compete. The distribution of resources follows the competitive principle: "the most successful take first". Strength and payoff define success, respectively. Successful agents extract resources first until the resource exhausts; Others receive nothing. We can expect an increased evolutionary pressure to select for successful strategies due to the increased competition for the limited resources. Unsuccessful strategies cannot survive easily. The limited resource inflow into the system limits the system's carrying capacity and adds another level of inter-agent competition, which effectively is expected to increase the evolutionary pressure towards successful strategies.

Introducing two distinct resources with separate extraction mechanisms and parameters offers a way to overcome the competitive exclusion principle (CEP) and allow strategy specialization. Each resource type requires agents to evolve a distinct competitive toolset to be able to extract enough resources to survive. Maintaining both toolsets is costly and most probably not competitive against specialized agents. Theoretically, this lies the foundation for agents to specialize in one of the two resource types. However, only simulations will show whether such strategy specialization actually happens and, if so, under which circumstances. In sum, two distinct resource types potentially provide different niches for agents to survive, which could ultimately result in strategy specialization.

Now, we will look at the detailed formulation and implementation of both distinct ways of resource extraction.

4.2.1 Basic Resource Extraction

All agents within the set of basic resource extracting agents at time t , $a \in \mathcal{B}_t$, intend to extract basic resources $\mathfrak{R}_b$ individually. However, the agents' environment only provides a limited amount of basic resource $\mathcal{A}_b$ available for extraction each time step t. We can think either of a regrowing resource assuming a state independent regrowth rule or a constant flux of resource for example provided by the sun. $\mathcal{A}_b$ determines the maximal global flux of basic resource into the system per time step. If there are enough resources available for all agents $a \in \mathcal{B}_t$ each agent receives an inflow of basic resources j_b added to its internal resource reservoir internal resource reservoir, i.e.,

$$R_a^{(t+\frac{1}{5})} = R_a^{(t)} + j_b \qquad \text{if} \sum_{a \in \mathcal{B}_t} j_b \leq \mathcal{A}_b, \quad \forall a \in \mathcal{B}_t. \tag{4.1}$$

Here, the system did not yet reach its carrying capacity. Each *ReCooDy* process alters the internally stored resources such that one time step consists of multiple sub-steps. The time t increases by a fifth to indicate that the first of five sub-processes that happen during one time step took place.

However, if the environment does not provide enough resources for all, $\sum_{a \in \mathcal{B}_t} j_b > \mathcal{A}_b$, competition arrises and only agents with high strength extract resources. Only the $n_b = \lfloor \mathcal{A}_b / j_b \rfloor$ agents with the highest strength s receive resources in this case. Here, $\lfloor ... \rfloor$ denotes the floor function. The $(n_b + 1)^{\text{th}}$ agent receives the rest and all others do not get anything. Put into an equation, the internal ressource reservoir R_a of agent a updates as follows:

$$R_a^{(t+\frac{1}{5})} = \begin{cases} R_a^{(t)} + j_b & \text{for the } n_b \text{ highest strength } s \text{ agents,} \\ R_a^{(t)} + (\mathcal{A}_b - n_b j_b) & \text{for the agent with } (n_b+1)^{\text{th}} \text{ highest strength } s, \\ R_a^{(t)} & \text{for the rest.} \end{cases} \tag{4.2}$$

Agents have an evolving strength s trait, i.e., it gets inherited with variation via small mutations from parent to offspring and selected through birth and death processes (see section 4.6 and section 4.5). More strength comes with higher costs of keeping up the agent's internal workings (see section 4.3). But the higher s, the more probable it is for an agent to be successful against the competition and actually receive resources.

We define the expected strength μ_s as the probability-weighted strength that is effectively used to extract basic resources over the course of an agent's lifetime, i.e.,

$$\mu_s = p_b \cdot s. \tag{4.3}$$

We will use this quantity in the simulation results to compare an agent's commitment to extract basic resources compared to synergistic resources.

4.2.2 Synergistic Resource

All agents within the set of synergistically extracting agents, $a \in \mathcal{S}_t$, aim to extract synergistic resources through local group interactions. They consitute an undirected, effective interaction network that changes over time t because the set of interacting agents and the network structure update (see section 4.2 and section 4.4). The complete network at a specific time t connects all living agents at that time, i.e., also the ones extracting basic resources, $a \in \mathcal{B}_t$. More formally, all agents $a \in N_t$ consitute the set of vertices V^t of the underlying mathematical graph and their undirected links define the edges $E^t \subseteq \{\{x, y\} | x, y \in V^t \text{ with } x \neq y\}$, which together define the underlying graph at a time t: $G^t = (E^t, V^t)$. The effective interaction network at a time t relies on a mathematical graph with vertices V_{eff} defined as the set of agents participating in the synergistic extraction at that time step, $a \in \mathcal{S}_t$. This set of vertices together with its corresponding set of effective edges $E^t_{\text{eff}} \subseteq \{\{x, y\} | x, y \in V^t_{\text{eff}} \text{ with } x \neq y\}$ form the underlying effective undirected graph:

$G^t_{\text{eff}} = (E^t_{\text{eff}}, V^t_{\text{eff}})$. It defines the actually relevant interaction structure in one specific time step, which is a result of the agent-specific probabilities p_s and p_b. In contrast, the whole network defines the theoretically possible set of interaction partners. Due to the linking process, both change over time (see section 4.4). I refer to the social environment of an agent as its neighborhood and next-neighborhood combined for the entire network and the effective social environment for the effective network. The social environment defines the possible set of agents with which to interact directly or via neighboring sub-interactions and which to modify in the linking process (see section 4.4). The interaction network with all its social environments defines a population structure that, in the EGT context, allows for the evolution of cooperation (*Lieberman* et al. 2005; *Santos and Pacheco* 2005; *Szabó and Fáth* 2007). In contrast to well-mixed systems, in which each agent can interact with each other agent, population structure potentially helps to overcome the CEP by providing the possibility of strategy niche creation.

Figure 4.2 shows a schematics of an effective interaction network that visualizes an agents current effective social environment. In the following, for elegance and simplicity, we will always focus on the effective network and omit to explicitly state that we consider the effective quantities, if not otherwise stated. The central agent is directly connected to three other agents within its neighborhood. Let $\mathcal{K}^t_a$ be the set of all neighbors of agent a at time t and $\mathcal{N}^t_a$ be the same set including agent a itself, i.e., $\mathcal{N}^t_a := \mathcal{K}^t_a \cup \{a\}$. Then, agent a participates in $n^t_a = \#\mathcal{N}^t_a$ sub-interactions, centered around itself as well as each neighbor.

The interaction itself is fundamentally based on a PGG, however, significantly altered compared to the common version (*Trivers* 1971): It entails a continuous strategy space in contrast to the typical binary cooperator-defector dualism and further introduces the notion of true defection by creating bads, i.e., actively destructing goods for selfish reasons. The PGG models a social dilemma group interaction, in which the individual outcome strongly depends on the actions of others. There are plenty of continuous trait games (see section 2.1.5). Here, we use continuous, evolving costs in the PGG and additionally remove the usual notion of defection. There is no overlaying strategy dualism. Instead, I introduce true defection, which occurs if agents have "negative costs" if we use the default PGG language. In the context of *ReCooDy*, negative costs mean that agents actively grab resources and, as a consequence, actively destroy goods. From a different perspective, we can say that they create bads in contrast to the usually created goods for positive costs. To distinct this generalized public goods game (gPGG) from the common PGG and emphasize its generalization, we will in part use a different language, e.g., we will talk about *investments* instead of *costs* or call the *game* an *interaction*.

In the following, I will introduce the gPGG by first focusing on goods creation before looking at bads creation and their unification.

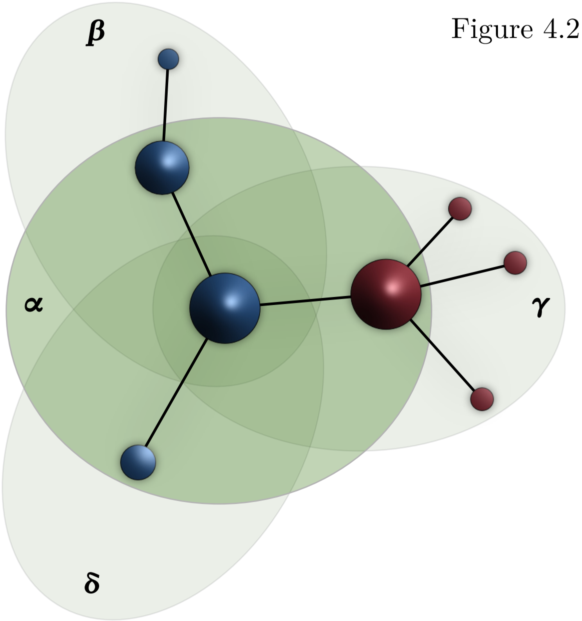


Figure 4.2: Schematic effective interaction network of agents $a \in \mathcal{S}_t$ extracting synergistic resources at time t. The graph exemplifies the local effective social environment of one central agent (big blue). The central agent participates in $n_a^t = 4$ sub-interactions: one centered around itself (green $\boldsymbol{\alpha}$ area), and the other three centered around its respective neighbors (light green $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\delta}$ areas). Each agent a invests an equal share ι_a / n_a^t in all its sub-interaction partitioning the potential gain as well as their risk. Colors indicate the existence of cooperative ($\iota_a > 0$, blue) and defective ($\iota_a < 0$, red) strategies. The vertex size schematically encodes payoff differences resulting from the interactions.

4.2.2.1 Creating Goods

Creating goods is the way to extract resources cooperatively. Let $c_{s_a} > 0$ be the cost an agent $a \in \mathcal{S}_t$ pays to interact with its effective social environment to extract synergistic resources. The resource goods G_v^+ created from the PGG-based sub-interaction centered around vertex v in the interaction network are

$$G_v^+ = r^+ \sum_{a \in \mathcal{N}_v} \frac{c_{s_a}}{n_a}. \tag{4.4}$$

Agents split their cost investments c_{s_a} equally among their sub-interactions (c_{s_a}/n_a), where n_a is the number of agents participating in the subinteraction around agent a. The positive synergy factor $r^+ \in \mathbb{R}^+$ transforms the summed costs to goods. If $r^+ > 1$, the transformed goods are worth more resources than the summed investments. The higher r^+, the more goods are produced and the weaker the social dilemma becomes (see chapter 5 for a more detailed analysis of the inherent social dilemma). All agents profit equally from the created goods, independent of their cost investments. Therefore, the payoff agent a gets from the interaction centered around vertex v is

$$P_{v,a}^+ = \underbrace{\frac{G_v^+}{n_v}}_{\text{shared benefit}} - \underbrace{\frac{c_{s_a}}{n_a}}_{\text{personal cost}}. \tag{4.5}$$

The first part represents the agent's benefit, the fraction of shared goods, and the second represents its cost invested in this specific sub-interaction. Recalling that agents participate in multiple interactions, we note that agent a's total payoff P^+ equals the sum of all sub-interaction payoffs:

$$\begin{aligned} P_a^+ &= \sum_{v \in \mathcal{N}_a} P_{v,a}^+ \\ &\overset{(4.5)}{=} \sum_{v \in \mathcal{N}_a} \left(\frac{G_v^+}{n_v} - \frac{c_{s_a}}{n_a} \right) \\ &\overset{(4.4)}{=} \sum_{v \in \mathcal{N}_a} \left(\frac{r^+}{n_v} \sum_{b \in \mathcal{N}_v} \frac{c_{s_b}}{n_b} - \frac{c_{s_a}}{n_a} \right) \\ &= \sum_{v \in \mathcal{N}_a} \left(\frac{r^+}{n_v} \sum_{b \in \mathcal{N}_v} \frac{c_{s_b}}{n_b} \right) - c_{s_a}. \end{aligned} \tag{4.6}$$

The resource flow from the synergistic resources $\mathfrak{R}_s$ from the interactions centered around vertex v is given by

$$j_{s_v}^+ = G_v^+. \tag{4.7}$$

These resources are provided by the environment and flow into the respective agents. Thus, $j_{s_v}^+$ also equals the total amount of resources interacting agents receive as benefits from the interactions centered around v. We should keep in mind that we assume that enough resources $\mathfrak{R}_s$ are available here for simplicity. I.e., the resources are unlimited, or the system is away from maximal total resource outflows. In section 4.2.2.5 we will focus on settings with finite resources.

4.2.2.2 Creating Bads

Creating bads is the way to extract resources destructively. I introduce the word *bads* as the conceptual opposite of goods to emphasize the conceptual similarity: Creating bads equals destroying goods. The idea behind bads creation (goods destruction) is that agents selfishly grab resources from the common resources, thereby destroying shared goods as collateral damage. We mirror the PGG characteristics by introducing the negative synergy factor, which assures that the agent's personal benefit from grabbing resources outweighs the commonly shared cost from destroying resources; In a sense, such agents incorporate true defection.

Let us first introduce bads creation as a separate concept before unifying both goods and bads creation based on their mathematical similarity.

Let $g_a > 0$ be the resources an agent $a \in \mathcal{S}_t$ selfishly grabs when interacting with its social environment. The bads B_v created from the PGG-like sub-interaction centered around vertex v in the interaction network are

$$B_v = r^- \sum_{a \in \mathcal{N}_v} \frac{g_a}{n_a}. \tag{4.8}$$

Note the structural similarity to created goods (see equation 4.4). Each agent a grabs equal shares g_a/n_a in each of its sub-interactions. The negative synergy factor $r^- \in \mathbb{R}^+$ transforms the summed grabbed resources into bads. If $r^- > 1$, the transformed bads are worth more resources than the summed grabbings. The higher r^-, the more bads are produced. For the conditions specifying a social dilemma see chapter 5. r^- enhances the negative impact of destroying resources through grabbing. However, as for goods creation, all participating agents share the bads. Therefore, the payoff an agent a receives from the destructive interaction centered around vertex v is

$$P^-_{v,a} = \underbrace{-\frac{B_v}{n_v}}_{\text{shared cost}} + \underbrace{\frac{g_a}{n_a}}_{\text{personal benefit}} . \tag{4.9}$$

The first part corresponds to the collectively shared cost of the created bads, and the second determines the personal benefit of agent a from grabbing resources. Agents participate in multiple interactions due to the network structure. Thus, agent a's total payoff P^- is

$$\begin{aligned} P^-_a &= \sum_{v\in\mathcal{N}_a} P^-_{v,a} \\ &\overset{(4.9)}{=} \sum_{v\in\mathcal{N}_a} \left(-\frac{B_v}{n_v} + \frac{g_a}{n_a}\right) \\ &\overset{(4.8)}{=} \sum_{v\in\mathcal{N}_a} \left(-\frac{r^-}{n_v} \sum_{b\in\mathcal{N}_v} \frac{g_b}{n_b} + \frac{g_a}{n_a}\right) \\ &= -\sum_{v\in\mathcal{N}_a} \left(\frac{r^-}{n_v} \sum_{b\in\mathcal{N}_v} \frac{g_b}{n_b}\right) + g_a. \end{aligned} \tag{4.10}$$

We define the resource outflow from the synergistic resource $\mathfrak{R}_s$ caused by the interaction centered around vertex v as

$$j^-_{s_v} = B_v + \sum_{a\in\mathcal{N}_v} \frac{g_a}{n_a}. \tag{4.11}$$

B_v are the created bads, the destructed resources as collateral damage, and $\sum_{a\in\mathcal{N}_v} \frac{g_a}{n_a}$ are the summed grabbed resources from all participating agents. We consider the directly grabbed resources from the available resources and assume that resources are actively destroyed through bads creation. The definition assures that for equal amounts of invested and grabbed resources, the impact of resource destruction on the resources provided by the environment is comparably higher for an equivalent setting: If we assume an (i) equivalent network, (ii) $r^+ = r^-$, and (iii) $g_a = c_{s_a}$ $\forall a$, we see that $B_v = G^+_v$ directly follows from their definitions (equation 4.8 and equation 4.4) through substitution. With these assumptions, we note that

$$j^-_{s_v} - j^+_{s_v} = B_v + \sum_{a\in\mathcal{N}_v} \frac{g_a}{n_a} - G^+_v \overset{B_v = G^+_v}{=} \sum_{a\in\mathcal{N}_v} \frac{g_a}{n_a}. \tag{4.12}$$

If we recall that $g_a > 0$, we see that for equal costs and grabbings, the outflow of resources from the synergistic resources $\mathfrak{R}_s$ is always larger for goods destruction compared to goods creation. By construction, the negative effect of bads creation is both directly harming neighboring agents' payoffs and reducing the available resources per time step through collateral damage.

Again, we should keep in mind that we assume that enough resources $\mathfrak{R}_s$ are available here for simplicity. I.e., the resources are unlimited, or the system is away from maximal total resource outflows. In section 4.2.2.5 we will focus on settings with finite resources.

4.2.2.3 Unify Creation and Destruction

Goods creation and bads creation exhibit the same underlying mathematical structure. This similarity allows us to unify both processes into one generalized public goods game (gPGG) interaction. We extend the usual realm of the PGG to negative parameters, which lets us incorporate true defection into the social dilemma interaction as we have investigated in the previous section. In the following, we will speak of goods creation and destruction to emphasize the unification of both processes. First, let us define an agent's investment $\iota \in \mathbb{R}$ as

$$\iota := \begin{cases} c_s & \text{if } \geq 0 \\ -g & \text{if } < 0. \end{cases} \tag{4.13}$$

If the agent's investment is positive, it pays a cost to create goods for itself and its effective social environment; If ι is negative, it selfishly grabs resources and harms itself and its effective social environment. The magnitude of ι determines both the amount of created or destructed goods and the personal harm or gain, respectively. For the interaction, we assume that goods creation and destruction have the same impact. Thus, we equate the positive and negative synergy factors and define a unifying synergy factor: $r := r^+ = r^-$. This synergy factor r transforms both positive investments into created goods and negative investments into destroyed goods with the same weighting. In one sub-interaction centered around vertex v, all participating agents create the following net amount of goods[4]:

$$G_v = r \sum_{a \in \mathcal{N}_v} \frac{\iota_a}{n_a}. \tag{4.14}$$

Agents distribute their personal investment equally between all sub-interaction, i.e., the invest ι_b / n_a in each sub-interaction. Each agent receives the same share of net goods. Therefore, the payoff an agent a receives from the sub-interaction centered around v is:

$$P_{v,a} = \underbrace{\frac{G_v}{n_v}}_{\text{group}} - \underbrace{\frac{\iota_a}{n_a}}_{\text{individual}}. \tag{4.15}$$

The first part, G_v / n_v, is the share of goods resulting from the sub-interaction, the second, ι_a / n_a, is the agent's investment, i.e., its subtracted cost or added grabbed resources, respectively[5]. Thus, the former represents the resource flow due to group effects resulting from the interactions, while the latter corresponds to resource flows caused by the

[4] In the following, we will always refer to *goods* instead of *net goods* but will imply the net amount of created and destructed goods.

[5] In the following, we will always refer to *investments* and imply that investments can per definition either be positive (costs for goods creation) or negative (grabbings from goods destruction).

individuals' actions. The resulting total payoff P from all sub-interactions of agent a is

$$\begin{aligned}
P_a &= \sum_{v \in \mathcal{N}_a} P_{v,a} \\
&\overset{(4.15)}{=} \sum_{v \in \mathcal{N}_a} \left(\frac{G_v}{n_v} - \frac{\iota_a}{n_a} \right) \\
&\overset{(4.14)}{=} \sum_{v \in \mathcal{N}_a} \left(\frac{r}{n_v} \sum_{b \in \mathcal{N}_v} \frac{\iota_b}{n_b} - \frac{\iota_a}{n_a} \right) \\
&= \sum_{v \in \mathcal{N}_a} \left(\frac{r}{n_v} \sum_{b \in \mathcal{N}_v} \frac{\iota_b}{n_b} \right) - \iota_a. \qquad (4.16)
\end{aligned}$$

The resource flow out of the synergistic resource $\mathfrak{R}_s$ due to the gPGG interaction centered around v is:

$$j_{s_v} = \underbrace{r \sum_{a \in \mathcal{N}_v} \frac{|\iota_a|}{n_a}}_{\text{creation/destruction}} + \underbrace{\sum_{a \in \mathcal{N}_v} \hat{g}_a}_{\text{grabbing}} \qquad (4.17)$$

with

$$\hat{g}_a := \begin{cases} |\iota_a| & \text{for } \iota_a < 0 \\ 0 & \text{else.} \end{cases} \qquad (4.18)$$

equation 4.17 defines the total resource outflow j_{s_a} that incorporates both goods creation and destruction. Recalling that $r = r^+ = r^-$, we retrieve the resource outflow from goods creation (equation 4.7) if we only consider positive investments $0 < \iota_a = c_{s_a}$ and the outflow from bads creation (equation 4.11) if we only consider negative investments $0 > \iota_a = -g_a$. The unified resource outflow j_{s_a} incorporates both the process of goods creation and goods destruction even in populations in which positive and negative investments ι coexist.

Within the resource-flow-based modeling approach, we note that agents cannot invest anything if they do not have enough resources available to do so. Therefore, we need to distinguish between an agent's investment trait i and its actual investment in the interactions ι. The former is the heritable trait passed on from parent to offspring, while the latter defines the situation-dependent actual action of the agent. We get the investment of an agent a in a time step from its trait via

$$\iota_a^{(t)} = \begin{cases} i_a \text{ if } \leq R_a^{(t)} \\ R_a^{(t)} \text{ else.} \end{cases} \qquad (4.19)$$

For the analysis and agent classification (see section 4.2.2.4), it is useful to introduce the expected investment μ_i of an agent:

$$\mu_i = p_s \cdot \iota. \qquad (4.20)$$

It combines the information of how often an agent interacts (p_s) and, if so, how much it is expected to invest (ι) if it has enough resources. Thus, it yields the expected value of investment within a time step.

4.2.2.4 Classification

Even though the gPGG is based on a continuous investment space, it is useful to introduce an agent classification to facilitate communication and concisely sum up the agents' emerging contextual characteristics. The gPGG has its foundation in the standard PGG. Figure 4.3 illustrates the similarities and differences of the strategies in both versions of the game. In section 2.1.2.2, we already introduced the PGG and its cooperative and defective strategies. Therefore, here we will focus on the strategies of the gPGG and especially its differences from the standard version of the game. Most of the time, we classify agents as cooperator, defector, or loner, but in specific simulation regimes, we will observe gap-separated strategy branches, for which we introduce the more contextual classification into benefactor, profiteer, exploiter, or malefactor.

A cooperator invests resources to create public goods, from which its entire social environment profits. A defector selfishly grabs resources and thereby destroys resources, for which all of its social environment has to pay. A loner relies on basic resources and does not participate in the interactions ($p_s \approx 0$ and $\mu_i \approx 0$). The idea to include not-interacting agents, however, as a predefined strategy, can be found already in *Michor and Nowak* (2002), who showed that Loners promote the evolution of cooperation in a simple EGT model. In *ReCooDy*, the loner strategy emerges. These three definitions in their detailed description expand the usual scope of the strategy set by allowing for continuous, principally unbound magnitude of behavior derived from the expected investments. Most notably, the gPGG redefines defectors as actually malevolent agents that destroy shared public goods to gain a personal benefit, in contrast, to merely not-cooperating agents that receive a public goods share without paying a cost. It refines the concept of defection to incorporate truly selfish and destructive behavior compared to just free-riding, i.e., true defection.

Within the strict PGG framework, *Hauert* et al. (2002) introduced a voluntary participation into the games in a computational model, *Michor and Nowak* (2002) the explicitly defined loner strategy, and *Semmann* et al. (2003) the former in an experimental setup. They found that the loner strategy can promote the evolution of cooperation. However, in these studies, the loner strategy was introduced as a given strategy, whereas in *ReCooDy* it can emerge out of itself if it proves successful. In general, the definitions of cooperator, defector, and loner each entail a broader range of possible microscopic strategy realizations compared to the PGG due to the unbound, continous evolvable investment space.

At times in diverse populations, context hugely matters when identifying, characterizing, and denoting an agent's behavior. Therefore, I introduce a more contextual agent classification by defining benefactors, profiteers, exploiters, and malefactors as illustrated in Figure 4.3. benefactors and profiteers are both cooperative strategies, but a benefactor has a significantly higher expected investment than a profiteer. A gap between the two sub-populations of the cooperative strategies contextually provides a significant difference in strategy. A benefactor is highly cooperative because it invests more and also more often than a profiteer, which makes the latter profit in total from the comparably more beneficial strategy of the first. exploiters and malefactors are both defective strategies, but a malefactor has a significantly lower expected investment than a exploiter. Here again, a gap between the two sub-populations of the defective strategies contextually provides a significant difference in strategy. A malefactor is highly defective because it destroys more public goods and also more often than an exploiter, which lets us coin the first as

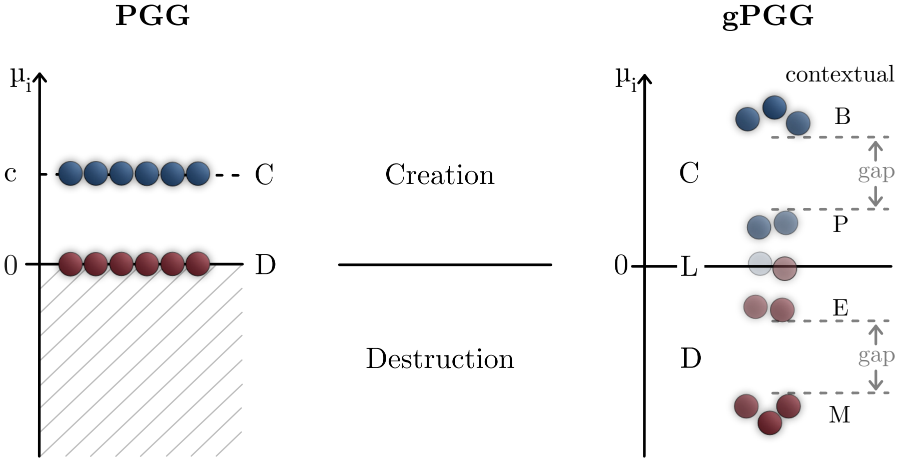


Figure 4.3: Strategy classification for the standard public goods game (PGG) (left) and the generalized public goods game (gPGG) (right). In the PGG, agents are either cooperators (C) or defectors (D). Cooperators usually pay a fixed cost c to create public goods, from which all benefit. Some models include varying costs from a continuous space, but costs are always restricted to positive values. Defectors do not pay a cost but still profit from the generated goods. Thus, the standard PGG only implements public goods creation.
The gPGG generalizes the standard PGG to incorporate public goods creation as well as destruction. Agents are either loners (L), cooperators (C), or defectors (D). Loners do not interact, i.e., they have $\mu_i \approx 0$ with $p_s \approx 0$. Cooperators create public goods through positive investments, and defectors destroy public goods through negative investments. A positive investment means paying a cost and a negative one actively grabbing resources. Thus, the gPGG extends the concept of defection by incorporating true defection—active destruction of goods for a personal benefit. All interacting agents benefit from the created public goods but have to pay for the destructed ones. We will further use a contextual classification: cooperators separate into benefactors (B) with high μ_i and profiteers (P) with low μ_i through a gap in μ_i, and defectors separate into exploiters (E) with low $|\mu_i|$ and malefactors (M) with high $|\mu_i|$ also through a gap. *Blue* represents cooperative strategies and *red* defective ones. More opaque colors represent a comparatively smaller magnitude of cooperative or defective behavior, respectively.

indeed selfishly malevolent and the latter as comparably less selfish but still exploitative. Introducing the contextual classification lets us realize that the connotation of agents can switch during their lifetime dependent on dramatic population changes. For example, former profiteers can suddenly become benefactors after the collapse of a subpopulation. Looking from a different angle, this also means that investing $\mu_{i_a} = 4$ could make an agent a profiteer at early simulation times but a benefactor at later times. The combined strategies of other agents influence how an agent is denoted; In short, context matters. The contextual agent classification lets us denote agents in specific emerging contexts, which is especially useful in extreme dynamics events, in which a distinct subpopulation significantly impacts the dynamics of the whole system.

In general, we note that due to the more complicated interaction in the gPGG, it becomes more complicated to classify and denote agents. As we will experience later on in the results, a flexible contextual characterization is helpful to communicate and explain the results in specific scenarios efficiently.

4.2.2.5 Finite Resources

Until now, the mathematical formulation implicitly assumed an infinite amount of resources. But what if there are not enough resources available for all agents anymore? In other words, what if the population reaches the environment's carrying capacity? Let us assume that the synergistic resource $\mathfrak{R}_s$ is limited similar to the basic resource $\mathfrak{R}_b$ (see section 4.2.1). Each time step t, the environment provides the amount $\mathcal{A}_s$ of synergistic resource. This amount defines the maximal resource flow out of the synergistic resource into the system. If $\mathcal{A}_s$ is greater than the sum of all extracted resources through synergistic extraction all agents indeed receive their payoff (equation 4.16) added to their resource reservoir:

$$R_a^{(t+\frac{1}{5})} = R_a^{(t)} + P_a^{(t)}, \quad \text{if} \sum_{a \in \mathcal{S}_t} j_{s_a}^{(t)} < \mathcal{A}_s^{(t)}, \quad \forall a \in \mathcal{S}_t. \tag{4.21}$$

However, if there is not enough resource available in the specific time step, $\sum_{a\in\mathcal{S}_t} j_{s_a} > \mathcal{A}_s$, agents need to compete for the available resources. Agents are successful if they have a high payoff. The n_s agents with the highest payoffs P indeed receive resources. The $(n+1)^{\text{th}}$ highest payoff agent takes in the remaining resources, and all others receive nothing. Importantly, agents that do not receive their payoff as a resource gain still have to pay their contributed cost investments. Thus, positive investments have a gambling characteristics, which in this case yields no return. Grabbing resources does not have such a direct gambling penalty. Still, unsuccessful agents cannot grab resources because there are not resources left. In total, agents internal resources update according as follows:

$$R_a^{(t+\frac{1}{5})} = R_a^{(t)} + \begin{cases} P_a^{(t)} & \text{for the } n \text{ highest-payoff agents,} \\ \left(\mathcal{A}_s^{(t)} - \sum_{i=0}^{n_s} j_{s_i}^{(t)}\right) - \hat{c}_{s_a} & \text{for the } (n+1)^{\text{th}} \text{ highest-payoff agent,} \\ -\hat{c}_{s_a} & \text{for the rest} \end{cases} \tag{4.22}$$

with

$$\hat{c}_{s_a} := \begin{cases} \iota_a & \text{if } \iota_a \geq 0 \\ 0 & \text{else.} \end{cases} \tag{4.23}$$

In summary, extracting synergistic resources is a high-risk, high-reward way of getting resources, in which agents can profit highly from synergies but are at the same time exploitable. They must be successful with their strategy within their social environment to indeed get resources and be able to act and survive.

4.3 Living

Agents need resources to survive and keep up their internal metabolism[6]. Each time step t, all agents pay a constant cost of living c_l and a cost to sustain their personal strength c_s:

$$R_a^{(t+\frac{2}{5})} = R_a^{(t+\frac{1}{5})} - (\underbrace{c_l}_{\text{living}} + \underbrace{c_{s_a}}_{\text{strength}}). \tag{4.24}$$

For simplicity, the strength sustenance cost is equal to an agent's strength: $c_{s_a} = s_a$.

From a modeling perspective, the costs of living introduce dissipation, i.e., a resource sink, in the system, which scales with the number of agents N. In combination with the death of exhaustion (see section 4.5), it assures that the population is constrained and cannot grow infinitely. Moreover, agents cannot evolve infinite strength because the strength cost introduces an indirect strength limit. We can expect agents to evolve an optimal strength that enables them to compete against others for basic resources but still yields return when subtracting the strength cost. The living process with its resource dissipation implements part of a self-regulatory way to limit the population size and computational time while increasing the evolutionary pressure for the agents at the same time.

ReCooDy's living process is perhaps the most straightforward way to model the costs of living costs or even an underlying metabolism. Of course, the living process would need to be adapted to cope with the organism's complexity for more complex organisms. Still, even in its simple form, the living process is crucial because it structurally implements resource dissipation, a property many evolving systems exhibit in some form. Thus, not including living costs at all would arguably be an even greater simplification.

4.4 Linking

Linking enables agents to invest their resources to shape their social environment, potentially providing them with the means to optimize their social niche for maximal synergistic resource extraction. The linking process described in this work is inspired by *Akçay* (2018). However, I significantly extend the process by separating link addition from removal, introducing various linking modes that let agents optimize using the information on others, and introducing a configurable linking cost. The extension provides a multidimensional choice space for agents to link. The evolutionary mechanism determines within

[6] The existence of a metabolism is sometimes seen as life's defining property; *Trifonov* (2011) made a meta-analysis on the definition of life. First, he summarized 123 definitions to "Life is [System, Matter, Chemical (Metabolism), Complexity (Information), (Self-)Reproduction, Evolution (Change), Environment, Energy, Ability,...] where the square brackets correspond to some compact expression containing the words listed within." Finally, he condensed it further into an inclusive overarching definition "Life is self-reproduction with variations.", following *Darwin* (1859).

the coevolutionary setting of *ReCooDy* which linking mechanism actually is competitive. (*García and Traulsen* 2019) found in a PGG setting that "letting evolution, and not modellers, decide which strategies matter" can be crucial for the outcome because unlikely low-abundance strategies "only temporarily pave the way for other strategies". In this sense, the linking mechanism presented below follows *Akçay* (2020) by letting the social setting coevolve with strategy but goes one step further by increasing the choice-space and letting evolution decide on the outcome. Even more, the network needs to develop over an agent's lifetime from the inherited traits.

Each time step, all agents potentially add links and remove links stochastically dependent on their individual traits. Thus, the number of actually existing links within a population at a given time strongly depends on the evolutionary outcome and are not fixed as usually assumed for simplicity in EGT models. In *ReCooDy*, agents first add links to others and afterward remove links. In the following, I will introduce both processes.

4.4.1 Adding Links

Agents add links either locally within their social environment or globally within the whole population. A link addition threshold $t_a \in \mathbb{N}$ with $t_a \leq t_a^{\max}$ determines whether the agent indeed adds a link. It is an evolving agent trait limited to a maximal value $t_a^{\max}$. Per time step, an agent tries to add a link if $k_a \leq t_a$ with k being the agent's number of neighbors/links and degree.
The agent adds a local link with probability p_l and a global link with probability $p_g = 1 - p_l$. How do agents choose their linking targets? For both local and global link addition, agents inhere a heritable linking mode trait that determines which agent to choose as a target; ν_l for local linking and ν_g for global linking, respectively. They represent an integer value in the range $\{0, \cdots, 7\}$ ($\nu_l \in \{0, \cdots, 7\}$ and $\nu_g \in \{0, \cdots, 7\}$). Each value represent an agent trait or state with the following encoding: $0 \leftrightarrow$ *None*, $1 \leftrightarrow R$, $2 \leftrightarrow p_s$, $3 \leftrightarrow i$, $4 \leftrightarrow$ *random*, $5 \leftrightarrow s$, $6 \leftrightarrow G$, and $7 \leftrightarrow P$. For example, $\nu_l = 3$ encodes that the agent links locally to its neighbor with the highest investment trait, $\nu_g = 4$ encodes that the agent links globally to a random agent, and $\nu_l = 0$ encodes that the agent does not add links at all. All linking modes are heritable traits, which are passed on from parent to offspring with rare mutations (see section 4.6 for more details). If there already is a link between source and target agent, no additional link is created; Instead, nothing happens. However, if adding links succeeds both agents potentially need to pay a linking cost: κ_l for the initiating source agent (local addition), γ_l for the target agent (local addition), and their global counterparts κ_g (source) and γ_l (target), respectively[7]. These linking costs are global and constant throughout a simulation for all agents.

We expect that the distinction between local and global link addition optimizes two distinct global network properties: Local link addition allows for an increased network clustering, whereas global link addition reinforces well-mixedness by reducing the average path length; The former should facilitate the evolution of cooperation whereas the latter should inhibit it (*Akçay* 2018).

[7] Throughout this thesis, we will only consider a linking cost for the initiating source agent. However, *ReCooDy*'s implementation includes the possibility to configure a cost for the target agent.

4.4.2 Removing Links

A link removal threshold $t_r \in \mathbb{N}$ with $t_r \leq t_r^{\max}$ determines whether the agent indeed removes a link. It is an evolving agent trait limited to a maximal value $t_r^{\max}$. Per time step, an agent tries to remove multiple links until the condition $t_r < k_a$ is met with k being the agent's number of neighbors/links and degree.
As for link addition, link removal is defined via a link mode ν_r defining which link to cut. The link mode is an integer value in the range $\{0, \cdots, 7\}$, i.e., $\nu_r \in \{0, \cdots, 7\}$. Each value represent an agent trait or state with the following encoding: $0 \leftrightarrow$ *None*, $1 \leftrightarrow R$, $2 \leftrightarrow p_s$, $3 \leftrightarrow i$, $4 \leftrightarrow$ *random*, $5 \leftrightarrow s$, $6 \leftrightarrow G$, and $7 \leftrightarrow P$. For example, $\nu_r = 3$ encodes that the agent removes the link to its neighbor with the lowest investment trait, $\nu_r = 4$ encodes that the agent removes a random link and $\nu_r = 0$ encodes that the agent does not remove links at all. Note that for link removal, the lowest, not the highest value of the respective quantity is relevant in contrast to link addition. The removal link mode ν_r is a heritable trait, which is passed on from parent to offspring with rare mutations (see section 4.6 for more details). Each link removal action costs resources for the initiating agent κ_r and the target agent γ_r[8]. The linking costs are global and constant throughout a simulation for all agents.

4.4.3 Summary and Discussion

To summarize, let us write down the resource update for all linking subprocesses:

$$
\begin{aligned}
R_a^{(t+\frac{3}{5})} = R_a^{(t+\frac{2}{5})}& \\
&- \underbrace{\begin{cases}\kappa_{l_a} & \text{if linking}\\ 0 & \text{else}\end{cases}}_{\text{add local links}}
- \underbrace{\begin{cases}\kappa_{g_a} & \text{if linking}\\ 0 & \text{else}\end{cases}}_{\text{add global links}}
- \underbrace{\begin{cases}\sum \kappa_{r_a} & \text{if cutting}\\ 0 & \text{else}\end{cases}}_{\text{remove links}} \qquad \text{(source)}\\
&- \underbrace{\begin{cases}\gamma_{l_a} & \text{if receiving}\\ 0 & \text{else}\end{cases}}_{\text{receive local links}}
- \underbrace{\begin{cases}\gamma_{g_a} & \text{if receiving}\\ 0 & \text{else}\end{cases}}_{\text{receive global links}}
- \underbrace{\begin{cases}\sum \gamma_{r_a} & \text{if cut}\\ 0 & \text{else}\end{cases}}_{\text{remove links}} \qquad \text{(target)}
\end{aligned}
\tag{4.25}
$$

Agents pay resources if they initiate linking (source agents; middle line) or, in principle, if they are the targets of linking actions (target agents; bottom line). In the results presented later in this thesis, we do not have target costs; thus, all equations in the bottom line are equal to zero. Still, *ReCooDy*'s implementation contains the possibility to set and explore target costs for future exploratory work. The first term for the source and target agents corresponds to the respective local linking cost. The second term corresponds to the respective global linking cost. The last term corresponds to the respective costs of potentially removing multiple links. equation 4.25 gathers the costs for all linking actions described above into one formula that shows the agents' internal resource update.

The link modes for link addition and removal presume different amounts of sensory capabilities within agents' social environment. For example, *random* linking does not require any cognitive capabilities to sensor and process information. Contrasting, resource-focused linking (e.g., $\nu_l = 1$) presumes that agents can sense and evaluate, or at least

[8] Throughout this thesis, only the initiating source agents pay a cost to remove or add links.

estimate, how many resources the target agent has accumulated internally. Importantly, *ReCooDy* provides various possibilities for how to link without judging them unuseful beforehand, and evolution decides which linking strategy is successful.

Adding and removing links are decoupled mechanisms, each with a separate set of parameters, some of which are evolving. The decoupling allows for different linking strategies that can evolve if competitive against others, but most not. Let us look at two examples that can evolve, just focussing on adding v.s. removing links and the resulting interpretation and implication: $t_a \geq t_r$ and $t_a < t_r$. In the first case, agents require more neighbors to add links than to remove links. Thus, they will never initiate link addition and spend resources on it. Instead, they rely on other agents to create links to them to increase their number of neighbors. If the number of neighbors increases high enough through this passive process, they will spend resources to remove as many links as to reach their link removal threshold. This linking strategy saves the agent all link addition costs. It either becomes an isolated agent or, if attractive for link-adding agents, can either accumulate the incoming links without any costs or optimize their social environment efficiently by spending resources only on cutting undesired links. Still, if many others target such agents, they can be exhausted by spending exceeding amounts of resources on removing links, which eventually can result in death from exhaustion. High values of t_r on the one hand decrease this risk, but on the other hand also decrease optimization possibilities. In the latter case, agents spend resources mainly on link addition and not on link removal. They potentially optimize their social environment under the risk of spending too many resources on it. We can expect the influence of both to scale with t_a. Only if such agents become desired targets of others do they need to spend resources on removing links. Here agents can be expected to experience a more dynamically changing social environment, compared to the first case. If both thresholds are approximately equal, $t_a \approx t_r$, agents arguably have the highest potential to optimize their social environment with the downside of having the highest total linking costs. They have the potential to actively choose beneficial target agents and remove not wanted neighbors if they receive links from others—as default, they accept incoming links but immediately remove a link to an unwanted neighbor.

ReCooDy decouples linking from the birth and death processes to include an agents' individual development phase. The decoupling enables to specify time-scales of the individual processes independently, instead of assuming that the development of an individual interaction network happens instantaneously in one model time step. The social environment is not entirely inherited but develops depending on inherited traits and specific local network configurations. The initial link to the parent and with it the parent's neighborhoods are passed on, which both significantly determine the agent's development phase (see section 4.6). The first provides a first interaction partner, and the latter determines the set of available linking partners for local linking. Thus, *ReCooDy* provides inherited predisposition combined with individual development in the social environment.

Having agents develop their social interaction network over their lifetime should facilitate the emergence of cooperation. As we will see in chapter 5, for a fixed synergy factor, the social dilemma is less severe for agents with fewer links and small social environments. Especially at the beginning of their lives, agents do not have many resources and, thus, cannot take many risks. Therefore, we can expect cooperation to evolve for lower synergy factors than scenarios in which links are already directly inherited. Later in their lives, when agents potentially have more internal resources available, we can expect them to survive settings with a large social environments. Such settings increase the social dilemma for the individual, and therefore the individual's risk, but can still yield high profits if many neighbors have higher expected investments than the respective agent. With links building up over time, the individual social dilemma increases, and agents experience more intricate settings, which are potentially more challenging to survive in but can also be expected to be beneficial if they live in cooperative environments.

4.5 Death

Agents die either stochastically with death probability p_δ per time step or from exhaustion if their resources fall below a death threshold t_δ. A dead agent is completely removed from the population together with all its links.

Each time step, a random number is drawn from a uniform integer distribution covering the unit interval for each agent. If that number is smaller than p_δ, the agent with all its links is removed. This process is a simple way of modeling death from age or accidents and assures that agents have a finite lifetime. Even highly successful agents cannot live forever, die eventually, and leave space and resources for other agents to replace them. Not having random deaths would enable unrealistic settings in which single successful agents could potentially survive forever, even on evolutionary time scales. Random deaths result in a Poisson age distribution with mean at $\mu_\delta = 1/p_\delta$. Therefore, varying the death probability enables adjustment of the death rate impacting the lifetime of generations of agents. We speak of one passed generation of agents if N_t agents died after time t, with N_t being the number of agents at time t. For *ReCooDy*, this rather rough classification of a generation is sufficiently exact to measure evolutionary time with respect to, on average, entirely exchanged populations. Further, from the system perspective, random deaths provide the means to investigate the resilience of successful strategies in *ReCooDy* by examining whether they can propagate and reappear after accidental extinctions.

Death from exhaustion is the direct consequence of unsuccessful strategies in *ReCooDy*'s evolutionary population dynamics. Each time step, all agents with less internal resources than the threshold die, i.e., if $R_a^{(t+\frac{3}{5})} < t_\delta$ agent a dies. It implements the first part of the evolutionary selection process because less successful strategies will experience higher death rates. Still, high death rates can be compensated by high birth rates, which constitute the second part of the selection process explained in section 4.6.

The dedicated death process acts as a resource sink, thus, explicitly introduces another source of resource dissipation into the system besides linking costs and positive investments. The death process constraints the system size by reducing the number of living agents such that population size cannot explode while at the same time increasing the agents' evolutionary pressure.

All agents a that do not die keep their internal resources:

$$R_a^{(t+\frac{4}{5})} = R_a^{(t+\frac{3}{5})}. \tag{4.26}$$

4.6 Birth

Agents procreate asexually and stochastically if they have enough resources. Each time step, an agent creates offspring with probability p_β if its internal resources exceed the global birth threshold t_β, i.e., if the condition $R_a^{(t+\frac{4}{5})} > t_\beta$ is met. To evaluate p_β, a random number is drawn from a uniform integer distribution spanning the unit interval. If that random number is smaller than p_β, the agent creates one offspring. The parent pays a birth cost c_β subtracted from its resource and transfers resources j_β to its offspring. Thus, when giving birth, the parents resources and the offsprings resources change as follows:

$$R_a^{(t+1)} = R_a^{(t+\frac{4}{5})} - c_\beta - j_\beta \qquad \text{(parent)} \tag{4.27}$$

$$R_o^{(t+1)} = j_\beta \qquad \text{(offspring).} \tag{4.28}$$

The offspring inherits its parent's traits with slight variation. Thus, these traits are not passed on perfectly but slightly mutate, which results in trait variation throughout the population. This variation is a necessary requirement for evolution. During their lifetime, agents' traits do not change. Therefore, there is no lifetime development of traits but only change through inheritance. Traits determine the agent's behavior and strategy, as we have seen during all model processes. How an agent's life will manifest exactly will not only depend on its traits but also on the agent's social environment as well as the other agents' strategies. In this sense, we could think about the very simplistic analogy to a genotype-phenotype mapping in which the agent's genotype (their traits) develops a specific phenotype (strategy) within their environment.

Table 4.1 gathers all evolving agent traits together with their respective sampling distributions used for mutations and their respective value spaces. Most traits change by adding a small mutation sampled from the corresponding distributions when they are inherited from parent to offspring. If the new trait would exceed the trait's value space it is capped to the upper or lower limit, respectively. In mathematical notation, a trait x is inherited from parent to offspring in the following way:

$$\begin{aligned} &x_{\text{offspring}} = \max(a, \min(x_{\text{parent}} + \Delta x, b)), \\ &\Delta x \sim U_x, \\ &x \in M_x,\ a := \min(M_x),\ b := \max(M_x), \\ &\forall x \in \{p_s,\ s,\ i,\ p_l,\ t_a,\ t_r,\ p_b,\ p_g\}. \end{aligned} \tag{4.29}$$

All mutations rely on random numbers drawn from uniform distributions. The more natural choice would probably have been to sample from Gaussian distributions. However, these come with the downside of comparatively high computational cost, which could result in significantly slower simulations of *ReCooDy* because a lot of random numbers are required. Further, the central limit theorem assures that for a large number of random numbers the resulting distribution approximates a Gaussian distribution. Even more, in *ReCooDy* it is more important that mutations happen than how exactly they happen, as long as mutations are comparatively small. These mutations lead to a variation of agent traits that will be selected. Here, we assume that the selection mechanism has a more significant impact on the evolutionary trajectory than the exact microscopic distribution of the mutations.

Table 4.1: Evolving parameters, their respective mutation sampling distributions, and value spaces. Distributions with curly brackets indicate discrete uniform distributions while square brackets denote continuous uniform ones.

symbol	trait	distribution U_x	value space M_x
		evolving	
p_s	interaction probability	$U_{p_s}[-0.05, 0.05]$	$[0, 1]$
s	strength	$U_s[-0.1, 0.1]$	$[0, \infty)$
i	investment trait	$U_\iota[-0.1, 0.1]$	$(-\infty, \infty)$
p_l	local linking probability	$U_{p_l}[-0.05, 0.05]$	$[0, 1]$
t_a	threshold for link addition	$U_{t_a}\{-1, 1\}$	$[0, t_a^{\max}]$
t_r	threshold for link removal	$U_{t_r}\{-1, 1\}$	$[0, t_r^{\max}]$
ν_l	link mode add local	$U_{\nu_l}\{0, 7\}$	$\{0, 1, \cdots, 7\}$
ν_g	link mode add global	$U_{\nu_g}\{0, 7\}$	$\{0, 1, \cdots, 7\}$
ν_r	link mode remove	$U_{\nu_r}\{0, 7\}$	$\{0, 1, \cdots, 7\}$
		indirectly evolving via coupled parameters	
p_b	basic extraction probability		$p_b = 1 - p_s$
p_g	global linking probability		$p_g = 1 - p_l$

The link modes ν_l, ν_g, and ν_r do not mutate each time an offspring is created but mutate stochastically with probabilities p_{ρ_l}, p_{ρ_g}, and p_{ρ_r}, respectively. The probabilities define the time scale at which mutations occur. If a mutation happens, the respective mode changes to one randomly selected mode within the corresponding value space. I.e., mutations of the link modes happen according to

$$x_{\text{offspring}} = \begin{cases} x_{\text{new}} & \text{with probability } p_x \\ x_{\text{parent}} & \text{else} \end{cases} \tag{4.30}$$
$$x_{\text{new}} \sim U_x,$$
$$\forall x \in \{\nu_l,\ \nu_g,\ \nu_r\}.$$

4.7 Initialization

When initializing model parameters and their corresponding processes, we define the specific storyline (see storyline 4.8). In this thesis, I focus on the emergence of cooperation in evolutionary systems. Emergence implies an initially neutral population, i.e., a population in which cooperation does not yet exist ($\mu_i = 0$) and in which there are no links between agents ($k = 0$) for all agents. Such a neutral initial population state contrast typical EGT models that usually presume the existence of at least a few cooperators that can invade the population. In *ReCooDy*, cooperation as a strategy needs to evolve in the first place. Further, a suitable interaction structure also needs to evolve and develop during an agent's lifetime. Both strategy and population structure evolve and will self-organize from the structurally included processes and their chosen parameters; Only computer simulation will tell whether cooperation emerges or not because the model's complexity most probably exceeds our human intuition and expectations capacity as well as analytical feasibility (*Holovatch* et al. 2017).

Parameters and their initialization determine the time scales of evolving agent traits and define the environment in which agents live and evolve. The rate and magnitude of the evolving traits and their relation to one another define how fast traits evolve compared to each other. Especially if we choose overlapping mutation time scales, we can expect the evolution of one trait to directly impact the evolution of another one and vise-versa resulting in feedback loops. Thus, the system can be expected to behave differently than if orders of magnitude separate time scales. As a simple example, consider the well-known spatial forest-fire model introduced by (*Drossel and Schwabl* 1992) which exhibits self-organized cauliflower patterns when simulated over long time scales on large grids. The tree growth rate and the velocity of spreading fires operate on similar time scales, which allows for the observed structural self-organization. However, if fires spread instantaneously, burning down whole tree clusters at once, the system exhibits a self-organized critical state (*Bak* et al. 1990), in which these cauliflower patterns disappear. The separation of time scales in these simple forest-fire models resulted in entirely different dynamical structures. This argument not only holds for the mutation rates of traits but can also be expected to hold more generally for processes with overlapping time scales. In this thesis, we choose all parameters such that they all operate in the same order of magnitude of time scales, such that we can assume the system to exhibit self-organized states with structural patterns.

The choice of parameters also defines the agents' environment. For example, the rates of basic and synergistic resources inflow, $\mathcal{A}_b$ and $\mathcal{A}_s$, determine the amount of resource flowing into the system and, therefore, the system's driving force. We use $\mathcal{A}_b$ and $\mathcal{A}_s$ to regulate the system size, i.e., the number of agents and links it can sustain given their strategies, its resilience, and the experienced evolutionary pressure on the agents. With more resources flowing into the system, more agents can sustain a successful strategy, but in principle, it also allows for other strategies to survive in respective niches more easily. More generally, we can expect a higher system driving force via increased resource inflows to enlarge the possibility space of the model, which comes with the tradeoff of higher computational costs.

As for many complex adaptive systems, effective parameters from experimental data is not readily available for *ReCooDy*; Instead, we need to choose and relate parameters to each other and systematically explore parameter regimes. Let us, for example, set the resource intake from basic resources to $j_b = 1$ as a reference value. With a basic and synergistic resource amount $\mathcal{A}_b = 500$ and $\mathcal{A}_s = 1000$, a synergy factor of $r = 2$, and a cost of living $c_l = 0.1$, we can estimate that the system can sustain less than $(\mathcal{A}_b + \mathcal{A}_s - \mathcal{A}_s/r)/c_l = 10\,000$ agents in a population with entirely positive investments. The last term in the sum corresponds to the agents' investments costs. The actual maximal number of agents will be significantly smaller because we omit strength, linking, and birth costs, which act as resource sinks. Hence, simulation results rather show at most $N = 4000$ agents living at a time. Still, the estimated upper limit is a guaranteed safeguard for population size and maximal estimated simulation time. We can further deduce that the agent's evolving strength should not exceed 1 over long times because the basic resource intake is fixed at $j_b = 1$, and more strength would result in a net loss. Such a strategy should not be sustainable and eventually result in death. Additionally,

Table 4.2: Initialized Parameters. For a detailed description of all parameters, look up the symbol in the corresponding Appendix list.

symbol	short description	initialized value
N	number of agents	1000
k	agents' number of links (degree)	0
R	agents' internally stored resources	5
A	agents' age	0
p_s	prob. of synergistic extraction	0.5
p_b	prob. of basic extraction	0.5
j_b	inflow of extracted basic resources per time	1
$\mathcal{A}_b$	amount of basic resources available per time	500
$\mathcal{A}_s$	amount of synergistic recources available per time	1000
s	agents' strength	0
i	agents' investment trait	0
r	synergy factor	$\langle$varying$\rangle$
c_l	cost of living	0.1
c_s	cost of strength	0
p_l	prob. to add links locally	0.5
p_g	prob. to add lings globally	0.5
t_a	threshold for link addition	2
$t_a^{\max}$	upper limit for the link addition threshold	40
t_r	threshold for link removal	4
$t_r^{\max}$	upper limit for the link removal threshold	40
ν_l	linking mode local addition	0.125 prob. each
ν_g	linking mode global addition	0.125 prob. each
ν_r	linking mode removal	0.125 prob. each
κ_l	cost to add local links (source)	1
γ_l	cost to add local links (target)	0
κ_g	cost to add global links (source)	1
γ_g	cost to add global links (target)	0
κ_r	cost to remove links (source)	1
γ_r	cost to remove links (target)	0
p_δ	agent's random death probability	0.01
t_δ	death threshold	0
t_β	birth threshold	5
p_β	birth probability	0.1
c_β	birth cost	4
j_β	transferred resources from parent to offspring	3
p_{ρ_l}	prob. to change the local addition link mode	0.01
p_{ρ_g}	prob. to change the global addition link mode	0.01
p_{ρ_r}	prob. to change the removal link mode	0.01

such parameter relations and derived expectations offer a convenient way of automatically testing macroscopic model expectations within Utopia (see section 3.2.5); Especially, edge-case scenarios can easily be tested, which in their entirety cover a broad range of model dynamics expectations. I carried out a similar choose-and-relate method for all initialization parameters to define a consistent storyline. Table 4.2 shows the resulting parameters used to initialize *ReCooDy* throughout this thesis if not otherwise stated. Of course, a multitude of different parameter combinations would be imaginable, defining different storylines and most probably showing varying dynamical regimes.

Ideally, we would carry out a sensitivity analysis with all parameters; However, it is fundamentally unfeasible due to the underlying curse of dimensionality. If we would sweep all 37 parameters in Table 4.2 with 10 different values each, we would need to simulate 10^{37} separate universe runs to simulate every possible parameter combination—A number vastly bigger than the approximately 10^{22} to 10^{24} stars in the universe (*The European Space Agency (ESA)* 2021). Furthermore, this would not yet include any statistics as each parameter combination would only be simulated ones. Sweeping over all combinations of only 3 or 4 parameters with multiple system realizations, each can become computationally unfeasible depending on the universe simulation time. The required computational time diverges following a power-law distribution with an increasing number of parameters deeming a complete, systematic parameter exploration impossible. Thus, we can only choose a few critical parameters to sweep over and classify emerging dynamical regimes. Still, such low-dimensional sweeps of the most relevant parameters are sensible approaches to systematically exploring dynamics regimes and the system's resilience. The underlying curse of dimensionality forces me to focus on a few specific selected individual exploration paths, leaving many options available for future investigations.

4.8 Summary, Storylines, and Discussion

ReCooDy is an abstract model and way too simple to model any reality with all its details, its intrinsic complexity, and facets. It is especially true when we imagine *ReCooDy*'s population to represent human societies. For example, take the linking mechanisms and think about a real-world system in which entities would always look within the entire population which agent has a specific highest trait or state and, without any further thoughts, directly create a link to the corresponding individual. The target always accepts without question, but can in principle, remove another link. We can easily imagine questions such as "Where do agents get the needed information from to detect their targets?", "Why do they always behave the same following their traits (except for a bit of randomness)?", "What if linking to the best is not the best option in a specific context because everyone does? Shouldn't agents be more intelligent?", and so on. These example questions allude to the common intrinsic issue of modeling assumptions, their level of detail in mathematical modeling, and how many and which processes are needed to correctly investigate a specific real-world system. A modeler needs to decide which processes and degree of detail to incorporate, and more importantly, what kind of knowledge can be extracted from a specific model. The question of how the correct high-level abstraction for a low-level real-world system looks like is a tremendously difficult one, usually connected to a tradeoff between realism and analytically as well as computational feasibility.

Despite it probably being too simple to represent reality adequately in all its level of detail, *ReCooDy* is, to the best of my knowledge, enormously too complex to be analytically feasible with our current mathematical toolset. While most EGT models focus on just one or two (co-)evolving parameter(s) and related processes, *ReCooDy* has nine coevolving parameters. Moreover, it incorporates several intertwined processes and introduces an overarching resource-flow-based modeling approach to the fundamental population dynamics, and assures consistency of the system's driving forces. The combinatorial space of possible system realizations is extremely high-dimensional. However, when simulating *ReCooDy*, the implemented evolution process will choose one specific path along which to progress. Another intrinsic mathematical difficulty lies in the history-dependence of the system because agents' actions accumulate over their overlapping lifetimes, and trait combinations progress even further for generations. Additionally, most processes are governed by probabilities, that we cannot expect to create enough statistics for all imaginable strategy combinations within their local dynamically changing settings. All of the mentioned pose severe challenges for a mathematical all-entailing approach.

Thus, *ReCooDy* is a system that is too simple for reality in all its detail and is too complex to understand systematically in its entirety. So, the obvious questions arise about what we can do with such a model and its purpose. As *Goldenfeld and Kadanoff* (1999) pointed out: In complex system science, we typically learn "lessons" from one system, adapt, and apply them to other ones. We identify and conceptualize the structural properties and processes of real-world systems and model them comparatively. Thus, we intend to create high-level abstractions of low-level systems. It is crucial to include all relevant processes and their structural properties, even if their level of modeling detail is low. More details are essential for more accurate results on a more refined level of investigation, and more exact parameter estimates if these are the aim of the scientific endeavor. With *ReCooDy*, I aim to identify structural properties that describe various real-world systems on a high level of abstraction. As motivated before, with *ReCooDy* we may extract results in a heuristic manner by including and coupling several structural properties and apply these results to real-world systems in order to better understand them on an abstracted high level.

Let us start by summing up *ReCooDy*'s structural process scaffolding and extract *ReCooDy*'s structural properties. A population of agents competes for limited resources. Resources enable agents to survive, interact, and optimize their interaction environment. Interacting with others is a high-risk, high-reward way of getting resources. It allows for both public good creation and destruction. Resources can also be extracted in a more reliant, individual way with a fixed return, but still with the danger of receiving nothing for not being strong enough to compete. A dynamic network defines the interaction structure that agents can optimize locally or even globally for their advantage with a resource cost. The time scales of the dynamic linking and the resource extraction overlap. Multiple agent traits evolve at similar time scales. These traits characterize the included processes, which have a stochastic nature. Agents get entirely removed from the system, either by chance or from exhaustion. Offspring inherit traits of successful agents with some variation. Structurally abstracted, *ReCooDy* models a competitive, evolving, and dynamically changing population of agents that may cooperate, defect, or retract themselves to gain resources in order to survive, optimize their resources extraction mechanism, and pass on their traits. It is an evolving resource-flow-based cooperation dynamics model of dynamic populations.

With these structural properties in mind, let us unfold a few potential storylines to identify structural similarities.

Climate Change Greenhouse gas emissions drive global warming and climate change (*IPCC* 2021). Taking actions to reduce global greenhouse gas emissions and the probability of a system collapse can be identified as a PGG with its underlying tragedy of the commons as done, for example in *Barfuss* et al. (2020). However, the aim of reducing the greenhouse gas concentration in the atmosphere exhibits two distinct behavioral qualities structurally that are not captured in their entirety in the standard PGG but are included in *ReCooDy*'s gPGG interaction: Acteurs (individuals, organizations, institutions) can either cooperate by paying individual costs to actively reduce the atmospheric greenhouse gas concentration or defect by maximizing personal profits and actively emit greenhouse gases, for which all have to pay in the long run. Their positive and negative investments are in first approximation unbound and reflect a continuous spectrum. They can change their behavior towards more cooperative behavior, e.g., directly via planting trees or carbon capture technologies, or indirectly, e.g., by generally reducing consumption, relying on green energy sources, or reducing heat dissipation in buildings. In contrast, actors can become more defecting, e.g., by actively increasing their fossil fuel extraction or consumption, throwing working products away, or traveling more by airplane. Underlying all is the need to remain competitive to survive, improve personal networks, and thrive in a dynamically changing, evolving system with limited non-renewable energy resources in the long run.

Economics EGT is widely applied to economical settings (*Gintis* 2009; *Friedman and Barry* 2016), often assuming a "homo economicus" image. However, more recent research directions aim at extending the scope to emphasize our socio-economical aspect, i.e., recognizing that humans are social beings relying on complex social interaction networks; Thus, we are more accurately described as "homo socialis" and "homos ludens" species (*Gintis and Helbing* 2015; *Helbing* 2015). Using the "homo socialis" as a basis, a bottom-up economy built on a standardized reputation system becomes, at least in theory, imaginable (*Helbing* 2013). Social aspects fundamentally rely on cooperative, benevolent, and altruistic acts, which are structurally captured in the PGG. They can choose to invest in ethical, social, educational, sustainable, and even their competition. On the contrary, in economic contexts, we can identify truly defective behavior, which is not captured in the standard PGG but in *ReCooDy*'s gPGG: A possible actor's strategy is to actively work against the competition through selfish and destructive acts that generate a personal benefit and create collateral damage for others. Further, socio-economical settings frequently exhibit actors that newly enter the population, imitating existing strategies with variation, need to invest resources to develop their network over time, and eventually disappear on time scales of a few to many years. The individual influence and potential investments are strongly heterogeneous and, in first approximation, only bound to the available resources. More influence usually comes with more local or global connections that may be created or removed if not worth the cost of maintaining them. Dependent on the specific setting, we can identify goods as money, power, opportunity, a general measure of well-being, or similar[9].

[9] The gPGG incorporates a more general notion of public goods as the usual definition in the economic contexts, which define public goods as non-excludable and non-rivalrous goods such as clean air,

Bacteria population With a few adjustments, we could adapt the *ReCooDy* model to describe bacteria populations. Bacteria communicate and coordinate themselves via quorum sensing—sending and sensing of chemical signals (*Whiteley* et al. 2017). The effects of quorum sensing are public goods (*Diggle* et al. 2007). However, here we cannot easily identify active destruction of public goods, except perhaps if we extend the scope of the observed system to include multiple bacteria populations that could use the chemical signals differently. Indeed, although bacteria populations are spatially structured, they also seem to form socio-ecological networks of intra-population cooperation and inter-population competition (*Cordero* et al. 2012). Nevertheless, whether a form of true defection can be identified structurally in such settings remains unknown for now. In general, the environment provides the chemical building blocks used to create the chemical signal molecules. Especially with quickly growing populations, we can imagine cases in which the system's carrying capacity gets reached such that resources need to be competed for among agents. In general, to investigate bacterial populations in *ReCooDy*, we need to adjust either the structural processes or at least the chosen parameters; As a start, we would need to configure the model to a changed linking process by excluding the global linking and replace the link development phase with a more immediate social inheritance process. In principle, this can be achieved by just adapting configuration parameters. Nevertheless, bacteria populations exhibit a few structural properties, which *ReCooDy* implements.

Obviously, *ReCooDy* cannot represent any of these storylines in all their detail and complexity. Aiming at it would require knowing effective parameters and the indeed relevant processes at the more macroscopic level. Physical systems, which usually exhibit significantly less complexity and hierarchy than socio-ecological ones, such as water flowing through porous media, already show us that it is even possible for effective descriptions to fundamentally and irrevocably break down (*Roth* 2008). As Roth points out, in such cases, the only way to proceed is via computer simulations as done in most environmental systems. For *ReCooDy*, I structurally included multiple different processes, which we can identify in real-world systems as exemplified above. Thus, we have a high-level abstracted model system that structurally describes a range of possible low-level real-world systems. By construction, we can only extract structural and abstracted knowledge from the model.

Simplified representations of complex systems can help to extract structural knowledge. Examples are: From network science, we know that a preferential attachment mechanism in a network will lead to a scale-free degree-distribution (*Barabási and Albert* 1999); From complexity science, we know that a slowly-driven heap of sand will exhibit a self-organized critical angle with power-law distributed avalanche-sizes (*Bak* et al. 1987). A structurally similar mechanism leads to scale-free distributed burning clusters in a forest fire model (*Drossel and Schwabl* 1992); From chaos theory, we know that a deterministic time horizon exists in which we can determine the system's exact trajectory, which becomes fundamentally impossible for larger times (*Lorenz* 1963; *Strogatz* 2014). Of course, the mentioned examples are profoundly fundamental and had an enormous impact in their respective fields as well as beyond.

knowledge, or national security. See *Samuelson* (1954) for the first introduction of a public good in the economic context as a "collective consumption good" and section 4.2.2 for the defining properties of a public good in the gPGG.

In this work, we will investigate what happens if we make models social dynamics models more realistic and what structural results we can observe in the example system. We explore whether this modeling approach leads to significant deviations from our expectations and published results in the respective field of research. The questions we will explore in the following are more of the type: How does cooperation emerge in the first place and evolve if we base models on resource-flows within a population dynamics context? What if agents can dynamically change and optimize their individual social environments either randomly or through preferential attachment processes? What is the impact of a limited driving force? *ReCooDy* is an endeavor to explore such questions in a more realistic coevolving population dynamics-based EGT setting. Of course, within the scope of this thesis, we will not wholly elucidate all questions in all their generality.

5 The Social Dilemma

The interaction based on the gPGG introduced in section 4.2.2 (section 4.2.2.3) per definition contains social dilemma characteristics microscopically. For low synergy factors r, it models a tragedy of the commons. The standard PGG requires the dilemma condition $r < N$ where N is the number of agents playing the game (see section 2.1.2.2). As we will see in the following, the situation is more intricate for the gPGG, primarily due to the network topology. The fact that multiple sub-interactions occur each time agents interact, and that they partition their investments among the individual sub-interactions, effectively results in a changed, intricate dilemma setting for the agents dependent on local context.

5.1 Tragedy of the Commons

Let us first look at a single gPGG interaction and elaborate on its dilemma characteristics. Thus, we focus on the microscopically implemented interaction. Let us assume a fixed interaction group. The payoff of an agent a for a single interaction around vertex v is given by equation 4.15, which we use and slightly rearrange:

$$\begin{aligned} P_{v,a} &= \frac{G_v}{n_v} - \frac{\iota_a}{n_a} \\ &\overset{(4.14)}{=} \frac{r}{n_v} \sum_{b \in \mathcal{N}_v} \frac{\iota_b}{n_b} - \frac{\iota_a}{n_a} \\ &\overset{\mathcal{K}_v = \mathcal{N}_v \setminus \{a\}}{=} \frac{r}{n_v} \sum_{b \in \mathcal{K}_v} \frac{\iota_b}{n_b} + \left(\frac{r}{n_v} - 1 \right) \frac{\iota_a}{n_a}. \end{aligned} \tag{5.1}$$

The left summand represents the payoff agent a receives due to the investments of all other participating agents. The right summand equals the payoff a receives from only its own investment ι_a. When we fix the first and only focus on the latter, we immediately see that a positive change in investment ($\Delta\iota_a > 0$) only creates lower personal payoff $\left(\Delta P_{v,a} \overset{!}{<} 0\right)$ if the following condition is met: $r < n_v$. To indeed create synergies, we further required $r > 1$. Thus, we obtain the well-known tragedy of the commons condition (see section 2.1.2.2) implemented in the public goods game by construction. Increasing ones' investment results in a lower payoff as long as synergies are not exceedingly high.

Let us focus on another agent $b \neq a$. For simplicity and w.l.o.g., let us assume all other agents' investments to be zero. From agent a's investment alone, b receives a payoff $P_{v,b} = (r\iota_a)/n_a^2$. Thus, b will always receive a higher payoff as a if the latter increases its investment. It is also intuitively clear because, with the increased investment, a creates more public goods shared among all. Therefore, even if $r > n_v$, other agents will profit more from a personally increased investment because they profit from the shared goods while not paying for their creation. The PGG still implements a competitive dilemma that is, however, of a weaker kind as in the tragedy of the commons.

In summary, in a microscopic single gPGG interaction, if $r < n_v$, an agent receives a lower payoff from synergies for higher personal investment. For higher synergies, $n_v < r$, the agent receives a higher payoff from a higher investment but will still always create more payoff for others. We observe a tragedy of the commons for $1 < r < n_v$ and still a weaker competitive social dilemma for higher synergy factors $n_v < r$ in the microscopic gPGG.

5.2 Personal Dilemma

To explore the social dilemma in the network setting, let us take an agent's perspective and explore the effect of an investment mutation, $\tilde{\iota}_a := \iota_a + \epsilon,\ \epsilon \in \mathbb{R}$, on the resulting payoff difference $\Delta P_a = \tilde{P}_a - P_a$ in an otherwise constant network and population. Let us take the payoff equation (equation 4.16), assume fixed neighborhoods n_x and constant neighbors' investments, and use the sum's linearity to calculate the payoff difference:

$$\begin{aligned}
\Delta P_a &= \tilde{P}_a - P_a \\
&\overset{(4.16)}{=} \left[\sum_{v\in\mathcal{N}_a}\left(\frac{r}{n_v}\sum_{b\in\mathcal{N}_v}\frac{\iota_b+\epsilon}{n_b}\right) - (\iota_a+\epsilon)\right] - \left[\sum_{v\in\mathcal{N}_a}\left(\frac{r}{n_v}\sum_{b\in\mathcal{N}_v}\frac{\iota_b}{n_b}\right) - \iota_a\right] \\
&\overset{\text{linear}}{=} \left[\sum_{v\in\mathcal{N}_a}\left(\frac{r}{n_v}\sum_{b\in\mathcal{N}_v}\frac{\iota_b}{n_b}\right) + \sum_{v\in\mathcal{N}_a}\left(\frac{r}{n_v}\sum_{b\in\mathcal{N}_v}\frac{\epsilon}{n_b}\right) - \iota_a - \epsilon\right] \\
&\quad - \left[\sum_{v\in\mathcal{N}_a}\left(\frac{r}{n_v}\sum_{b\in\mathcal{N}_v}\frac{\iota_b}{n_b}\right) - \iota_a\right] \\
&= \sum_{v\in\mathcal{N}_a}\left(\frac{r}{n_v}\sum_{b\in\mathcal{N}_v}\frac{\epsilon}{n_b}\right) - \epsilon \\
&= \left(r\sum_{v\in\mathcal{N}_a}\frac{1}{n_v}\sum_{b\in\mathcal{N}_v}\frac{1}{n_b} - 1\right)\epsilon \qquad (5.2)
\end{aligned}$$

To get a strict tragedy of the commons setting as presumed in the well-mixed default PGG (see section 2.1.2.2), we require that a positive investment mutation results in a negative payoff difference. It guarantees that in a fixed social environment, a high-investment agent indeed has a smaller payoff than a low-investment agent from the personal investment. Thus, for $\epsilon > 0$, the strict tragedy of the commons condition is:

$$\begin{aligned}
& 0 \overset{!}{>} \Delta P_a \\
\Leftrightarrow\quad & 0 \overset{!}{>} \left(r\sum_{v\in\mathcal{N}_a}\frac{1}{n_v}\sum_{b\in\mathcal{N}_v}\frac{1}{n_b} - 1\right)\epsilon \\
\overset{\epsilon>0}{\Leftrightarrow}\quad & 1 \overset{!}{>} r\sum_{v\in\mathcal{N}_a}\frac{1}{n_v}\sum_{b\in\mathcal{N}_v}\frac{1}{n_b} \\
\overset{n_x>0}{\Leftrightarrow}\quad & r_{\mathrm{c}_a} \overset{!}{>} r \qquad \text{with } r_{\mathrm{c}_a} := \left(\sum_{v\in\mathcal{N}_a}\frac{1}{n_v}\sum_{b\in\mathcal{N}_v}\frac{1}{n_b}\right)^{-1}. \qquad (5.3)
\end{aligned}$$

If this agent-specific condition is met within its effective social environment an increase in personal investment results in a decrease in payoff and vice versa. Agents that have (high) positive investments within their mesoscopic social environment and satisfy the condition (equation 5.3) truly experience a strict personal social dilemma because if they increase their investments, they will receive a lower payoff.

Let us change our perspective and focus on agents with negative investments. If an agent grabs more resources ($\epsilon < 0$) we expect the agent to increase its payoff difference $\Delta P_a \overset{!}{>} 0$ such that grabbing indeed is a selfish action with personal profit. We observe that compared to the derivation above, here we have both signs of the preconditions flipped such that, in the end, we retrieve the same condition as in equation 5.3. Thus, the strict dilemma condition for the unified goods creation and destruction processes is the same.

We notice that, for a single unconnected agent, the condition in equation 5.3 recovers the trivial expectation that synergies only occur if $r > 1$. Therefore, the agent will always profit from increasing its investment because it alone receives all the created goods.

We can further derive that r_{c_a} is monotonically increasing with the number of sub-interactions and also the number of agents in the next-neighborhood. The more links there are in the agent's social environment, the larger r_{c_a} gets. Moreover, we notice that for a constant synergy factor, the personal dilemma gets worse from an agent's point of view the more intra-connected and populated its social environment becomes. However, we only consider the individual view from personal investment changes so far and disregard the beneficial effects of having cooperative neighbors.

Equation 5.3 defines the condition for an agent's experienced personal dilemma that specifies whether it receives a payoff lower than its additional investment from all the interactions within its effective social environment. It emerges on the mesoscopic scale from the microscopically defined gPGG interaction rules.

5.3 Competition Dilemma

A requirement for the strict tragedy of the commons is that others receive more payoff than the investing agent if its investments are positive (see section 5.1). Here, we will explore what effect the population structure, which defines multiple interactions, has on this dilemma requirement.

Let us assume that, an agent changes its investment, $\tilde{\iota}_a = \iota_a + \epsilon$ with $\epsilon \in \mathbb{R}$ and $\epsilon > 0$, in an otherwise constant setting. I.e., its social environment and all of the neighbors' investments stay constant. The goods created in the sub-interaction around vertex v due to the investment change then are

$$\Delta G_v = \tilde{G}_v - G_v \overset{(4.14)}{=} \frac{r\epsilon}{n_a}. \tag{5.4}$$

With it, the change of the agent's payoff in one sub-interaction is

$$\Delta P_{v,a} = \tilde{P}_{v,a} - P_{v,a} \overset{(4.15)}{=} \left(\frac{r}{n_v} - 1\right)\frac{\epsilon}{n_a}. \tag{5.5}$$

In contrast, another participating agent b receives

$$\Delta P_{v,b} = \tilde{P}_{v,b} - P_{v,b} \overset{(4.15)}{=} \frac{r\epsilon}{n_v n_a} \tag{5.6}$$

because b receive a share of the additionally created goods while not needing to pay for an increased investment. Naively, we could conclude that agent a always has a payoff disadvantage compared to another participating agent b because $\Delta\ P_{v,a} - \Delta\ P_{v,b} = -\epsilon\ /\ n_a < 0$ with $\epsilon > 0$, thus recovering what we found in the single-interaction setting in section 5.1. However, the network topology again leads to a more complicated situation.

Agent a participates in multiple sub-interactions. Hence, a's accumulated total payoff difference is

$$\Delta P_a = \sum_{v \in \mathcal{N}_a} \left(\tilde{P}_{v,a} - P_{v,a}\right) \overset{(5.5)}{=} \sum_{v \in \mathcal{N}_a} \left(\frac{r}{n_v} - 1\right) \frac{\epsilon}{n_a}. \tag{5.7}$$

An agent b, which is part of a's social environment, only receives a share of the goods from the shared sub-interactions with a. Therefore, the total payoff change for a neighboring agent b from an investment change of a is

$$\Delta P_b = \sum_{v \in \mathcal{N}_a \cap \mathcal{N}_b} \left(\tilde{P}_{v,b} - P_{v,b}\right) \overset{(5.6)}{=} \sum_{v \in \mathcal{N}_a \cap \mathcal{N}_b} \frac{r\epsilon}{n_v n_a}. \tag{5.8}$$

With it, we can derive the condition for when agent b receives a smaller change in total payoff than a from a personally increased investment of a, $\epsilon > 0$:

$$\begin{aligned}
& 0 \overset{!}{>} \Delta P_b - \Delta P_a \\
\overset{(5.7),\ (5.8)}{\Leftrightarrow}\quad & 0 \overset{!}{>} \sum_{v \in \mathcal{N}_a \cap \mathcal{N}_b} \frac{r\epsilon}{n_v n_a} - \sum_{v \in \mathcal{N}_a} \left(\frac{r}{n_v} - 1\right) \frac{\epsilon}{n_a} \\
\Leftrightarrow\quad & 0 \overset{!}{>} \sum_{v \in \mathcal{N}_a \cap \mathcal{N}_b} \frac{r\epsilon}{n_v n_a} - \sum_{v \in \mathcal{N}_a} \frac{r\epsilon}{n_v n_a} + \sum_{v \in \mathcal{N}_a} \frac{\epsilon}{n_a} \\
\Leftrightarrow\quad & 0 \overset{!}{>} - \sum_{v \in \mathcal{N}_a \setminus (\mathcal{N}_a \cap \mathcal{N}_b)} \frac{r\epsilon}{n_v n_a} + \sum_{v \in \mathcal{N}_a} \frac{\epsilon}{n_a} \\
\overset{\epsilon>0,\ n_a>0}{\Leftrightarrow}\quad & 0 \overset{!}{>} - \sum_{v \in \mathcal{N}_a \setminus (\mathcal{N}_a \cap \mathcal{N}_b)} \frac{r}{n_v} + \sum_{v \in \mathcal{N}_a} 1 \\
\overset{\substack{\mathcal{N}_a = \mathcal{K}_a \cup \{a\} \\ \mathcal{N}_b = \mathcal{K}_b \cup \{a\} \\ a \in \mathcal{N}_a,\ a \in \mathcal{N}_b}}{\Leftrightarrow}\quad & 0 \overset{!}{>} - \sum_{v \in \mathcal{K}_a \setminus (\mathcal{K}_a \cap \mathcal{K}_b)} \frac{r}{n_v} + n_a \\
\Leftrightarrow\quad & r \overset{!}{>} \hat{r}_{\mathrm{c}_{a,b}} \qquad \text{with } \hat{r}_{\mathrm{c}_{a,b}} := \frac{n_a}{\sum_{v \in \mathcal{K}_a \setminus (\mathcal{K}_a \cap \mathcal{K}_b)} \frac{1}{n_v}}.
\end{aligned} \tag{5.9}$$

This condition defines the competition dilemma of a regarding neighbor b: If $r > \hat{r}_{\mathrm{c}_{a,b}}$ is satisfied, agent a receives a higher change in total payoff when increasing its investment than agent b. Instead, if $r < \hat{r}_{\mathrm{c}_{a,b}}$, agent a experiences a competition dilemma regarding b because increasing its investment results in a higher total payoff change than its neighbor'. For each possible agent-neighbor pair equation 5.9 defines the personal competition dilemma condition of the former.

We notice that the number of neighbors that a does not share with b, encoded in $\mathcal{K}_a \setminus (\mathcal{K}_a \cap \mathcal{K}_b)$, and the actual number of agents participating in each sub-interaction impact whether a experiences a competition dilemma. As a rule of thumb, if a and b

share a lot of neighbors, $\#\mathcal{K}_a \approx \#(\mathcal{K}_a \cap \mathcal{K}_b)$, an increased investment of a will more probable result in comparatively higher total payoffs for b. Thus, shared neighbors tend to increase the personal competition dilemma. However, if they do not share many neighbors, $\#\mathcal{K}_a \gg \#(\mathcal{K}_a \cap \mathcal{K}_b)$, and if agent a increases its investment, it can receive a higher payoff than its neighbor b.

5.4 Instructive Example

Let us focus on an example to illustrate the different social dilemma qualities. We take the central agent in Figure 4.2 and calculate r_{c_a} and $\hat{r}_{\mathrm{c}_{a,b}}$ with respect to all its neighbors. That way, we can elaborate on the agent-specific dilemma situation. For the personal dilemma, using equation 5.3 and inserting the respective numbers yields $r_{\mathrm{c}_a} = 623/450 \approx 1.38$. If r exceeds this value, replacing a with a higher investment agent on the same vertex results in more payoff for the latter. Thus, only if $r < 1.38$, that specific agent experiences a personal dilemma.

Let us now focus on the competitor dilemma; thus, on how all its neighbors' payoff would change dependent on r. Using equation 5.9 and inserting the respective numbers, we get (i) an undefined result for the bottom left single neighbor, (ii) $\hat{r}_{\mathrm{c}_{a,b}} = 24/5 = 4.8$ for the right neighbor, and (iii) $\hat{r}_{\mathrm{c}_{a,b}} = 40/7 \approx 5.7$ for the top left neighbor. Thus, if a increases its investment, neighbor (i) will always have a higher total payoff than a, independent of r, neighbor (ii) will have a higher total payoff only if $r < 4.8$, and neighbor (iii) only if $r < 5.7$.

Agents experience different qualities of a social dilemma or even no dilemma at all, strongly dependent on their social environment and the synergy factor r. The simple example illustrates that the population structure leads to intricate social dilemma settings with different qualities—reducing the gPGG to one overarching public goods game dilemma is a too simplistic point of view already in this simple example setting.

5.5 Offspring Benefit

The benefit of cooperative offspring is key to understanding the emergence of cooperation in *ReCooDy*. Here, agents' strategies are passed on to their offspring (see section 4.6), but their social environment is not directly inherited. Instead, an offspring starts its life with a single connection to its parent and can build up connections during its lifetime. If we want to build a deeper understanding of *ReCooDy*'s social dilemma, we need to explore how the parent's payoff changes when it gives birth to an offspring.

Let us take the minimal example of two connected interacting agents (agent 0 and 1) and calculate the expected payoffs analytically that come with one of them creating offspring. For simplicity, we do not consider limited resources, and without loss of generality, we focus on agent 0 as the parent. Using equation 4.15 we can directly write down its payoff before giving birth:

$$P_0 = 2 \cdot \frac{r}{2}\left(\frac{\iota_0 + \iota_1}{2}\right) - \iota_0 = \frac{r}{2}\left(\iota_0 + \iota_1\right) - \iota_0. \tag{5.10}$$

Now, if agent 0 creates an offspring, which we denote as agent 2, the new payoff in the next interaction updates according to

$$\begin{aligned}\tilde{P}_0 &= r\left(\underbrace{\frac{1}{2}\left(\frac{\iota_0}{3}+\frac{\iota_1}{2}\right)}_{\text{1 centered}}+\underbrace{\frac{1}{2}\left(\frac{\iota_0}{3}+\frac{\iota_2}{2}\right)}_{\text{2 centered}}+\underbrace{\frac{1}{3}\left(\frac{\iota_0}{3}+\frac{\iota_1}{2}+\frac{\iota_2}{2}\right)}_{\text{0 centered}}\right)-\iota_0\\ \tilde{P}_0 &= r\left(\frac{4}{9}\iota_0+\frac{5}{12}\iota_1+\frac{5}{12}\iota_2\right)-\iota_0. \qquad (5.11)\end{aligned}$$

I indicated around which agent the respective sub-interaction is centered. The parent's payoff thus changes by

$$\begin{aligned}\tilde{P}_0 - P_0 &= r\left(\frac{4}{9}\iota_0+\frac{5}{12}\iota_1+\frac{5}{12}\iota_2\right)-\iota_0-\left(\frac{r}{2}\left(\iota_0+\iota_1\right)-\iota_0\right)\\ &= r\left(-\frac{1}{18}\iota_0-\frac{1}{12}\iota_1+\frac{5}{12}\iota_2\right)\\ &\overset{\iota_0\approx\iota_2}{\approx} r\left(\frac{13}{36}\iota_0-\frac{1}{12}\iota_1\right). \qquad (5.12)\end{aligned}$$

In the last step, we assume that the offspring's investment is inherited from the parent with expectation value $\langle\iota_2\rangle = \iota_0$ and small mutations with mean equal to zero (see the inheritance process in section 4.6). Hence, $\iota_0 \approx \iota_2$.

We observe the linear dependency of the parent's payoff change with its investment. Creating offspring creates more additional positive payoff for higher investment parents. Thus, more cooperative agents profit most from creating offspring in contrast to more defective agents that harm themselves by creating defective offspring.

Interestingly, the payoff difference is also proportional to the negative investment of the other agent connected to the parent. However, the impact is lower as the propertionality constant is significantly lower. Recognising its influence, we can calculate the condition for when a cooperative parent indeed profits from its offspring's investment:

$$\tilde{P}_0 - P_0 \overset{!}{>} 0 \quad \overset{(5.12)}{\Leftrightarrow} \quad \frac{13}{3}\iota_0 > \iota_1. \qquad (5.13)$$

Therefore, a positive investment parent only profits from its offspring if the other connected neighbor is not vastly more cooperative, as defined by the derived condition. What first seems surprising makes sense if we recognize that the interaction structure and the number of agents participating in each subinteraction changes with the new offspring. The offspring receives a part of the parent's previous share from the goods created by the additional high-cooperative neighbor.

From the opposite perspective of a defective parent, creating defective offspring is harmful because the parent has to pay for a part of the additionally destructed goods from the offspring. Still, the parent's payoff can increase from creating offspring if it has a much more defective neighbor characterized by the condition in equation 5.13. In that case, the offspring takes on a fraction of the destructed goods of the much more defective neighbor because of the changed interaction structure.

If we take a relative perspective to classify the agent, we observe that a much less defective parent than its neighbor is a comparatively much more cooperative agent. Therefore, we again see a manifestation of the emerging property that more cooperative agents profit from their offspring while more defective agents harm themselves from creating offspring.

This simple example illustrates a generalizable system property: More cooperative agents within a social environment profit from their offspring while more defective agents harm themselves. First, they have the direct benefit from the passed on cooperative trait and the created goods from the offspring. And second, if they are much less cooperative compared to their neighbors, they harm their profitable interacting structure, with which they profit from their neighbors, by having to share the neighbor's created goods with the offspring, too.

We should not forget that a benefit for the parent comes with a potential penalty for the cooperative offspring. For low synergy factors, and high absolute investment values, a cooperative offspring will typically have a disadvantage over a more defective offspring: The former shares the created goods with the parent and the parent's neighbors but needs to pay for the investment, while the latter will share the created bads with them and directly grabs resources. However, the exact individual scenario again depends on the local network configuration, which I will not derive here.

A generalization to arbitrary social environment would help us quantify the effect. However, qualitatively, we can expect to see the same properties.

We have to keep in mind that, in *ReCooDy*, creating offspring requires resources (see section 4.6 for birth costs). Thus, the offspring benefit only turns into a net resource gain for the parent if the general payoff difference $\tilde{P}_0 - P_0$ is big or often received. The latter requires the offspring to survive and not remove the link to the parent. Still, even if the parent receives a net resource loss from creating offspring, the net loss will usually be even greater if both have highly negative investments. The importance of the absolute received payoff further supports the observation that more cooperative agents profit from offspring while more defective ones do not.

5.6 Summary and Discussion

In this chapter, I analyzed the social dilemma, its microscopic implementation, and its emergent mesoscopic characteristics resulting from the interaction structure.

In section 5.1, I investigated a single gPGG and retrieved the standard public goods game dilemma condition $1 < r < n_v$ for a synergy factor r and n_v interaction agents (see section 2.1.2.2 for comparison). However, even for $r > n_v$, a competitor dilemma exists of the kind that an agent never receives more payoff than its competitors from a personal investment because it must pay for the investment while the created goods are shared.

As we have seen, the social dilemma situation becomes much more intricate with population structure and the resulting multiple sub-interactions. In section 5.2, I derived an agent's personal dilemma condition (see equation 5.3) and in section 5.3 the agents competition dilemma with respect to a specific neighbor (see equation 5.9).

In a personal dilemma, if an agent increases its investment, it receives less payoff than before. In a competition dilemma with a specific neighbor, if an agent increases its investment, it receives less payoff than its neighbor from the additionally created goods. Both conditions depend on the agent's entire social environment, and the latter also differs for each neighbor. Generally, the more connections there are within an agent's social environment, i.e., its neighborhood and also next-neighborhood, the stronger are both his experienced dilemmas.

The competition dilemma is of a weaker form than the personal dilemma. Only if an agent experiences both dilemmas defined by their respective conditions (equation 5.3 and equation 5.9), we should classify the public goods game interaction on the networks as a strict tragedy of the commons dilemma. We see that the emerging PGG condition on a network is completely different than for a single interaction; Most importantly, it is agent-specific and significantly depends on the local network realization. Thus, the social dilemma is an emergent property on the mesoscopic scale.

We can easily imagine settings in which there is no personal dilemma but still a competitor dilemma regarding one or several neighbors, as I have exemplified in section 5.4. We should not characterize such a situation as a strict public goods dilemma (or tragedy of the commons) because the individual increases its payoff by increasing its investment. However, in the evolutionary context, in which the success against competitors is relevant, we can still observe a competition dilemma when there is no personal dilemma because others profit even more from the increased investment.

Even more, agents can experience a competition dilemma with a subset of neighbors but not with all of them (see section 5.4). Combination of the personal dilemma condition (equation 5.3) and the competition dilemma condition (equation 5.9) for each possible pair of agents lets us anticipate the emerging intricacy of individual dilemma qualities resulting from the population structure.

Taking the macroscopic perspective by averaging the degree on the network level and demanding $r < \bar{k} + 1$ as the public goods game condition hides the emerging intricacy on the mesoscopic level, which agents actually experience. Even more, it is misleading as some agents can indeed experience no dilemma condition at all, and others may only experience a weak form. Therefore, demanding $r < \bar{k} + 1$ as the effective public goods game condition on networks is at least highly questionable, if not entirely wrong.

The situation gets even more complicated when we have settings that are not governed by simple birth-death-like or imitation-based evolutionary update rules in which the network topology stays constant but instead constantly developing network topologies. In such situations, as implemented in *ReCooDy*, it is easily imaginable to have agents that profit from high investments at the beginning of their development phase but experience harsh dilemma situations near the end of their lives. It is not clear whether we can average over such varying settings and scenarios and effectively state that agents experience a social dilemma, although, in the microscopically defined game, it clearly is a social dilemma. The results indicate that an effective macroscopic description captured in a single number does not exist because of the qualitative differences of the individual dilemma regimes.

Furthermore, for *ReCooDy*'s dynamics, the impact of offspring is very important. As derived in section 5.5, offspring creation is beneficial for cooperative agents but not for defective ones. Cooperative offspring creates additional goods for the parent, too, while defective offspring creates additional bads. Further, with the new agent, the parent's interaction structure also changes, and with it, the payoff fractions from the individual sub-interactions. Here, the parent's relative strategy compared to its neighbors is important. Again, comparatively more cooperative agents profit more from created offspring than comparatively more defective agents. Overall, we notice an intricate situation in which the absolute benefit depends on the parent's strategy as well as its neighbors and their strategies. These observations rely on the minimal example and are not derived in all generality. However, already from this simple example, we can qualitatively extract this statement because we could expect that for more connected parents, the offspring benefit shrinks as potentially more agents would receive a share of the additionally created goods from the offspring. Overall, we observe that cooperation is a self-supportive strategy in contrast to defection, which is self-harming when spreading.

It is not trivial to analytically extract general expectations of the interaction dynamics, especially in dynamically changing social environments on the macroscopic level. We could, for example, dive even deeper than before and look at more generalized population settings of offspring creation. The arguments and analytical considerations that we made so far indicate that handling *ReCooDy* in an entirely analytical way is very challenging. Still, analytical approaches such as presented here are helpful to develop the foundational, intuitive expectations in simple settings, which we can then apply to explain the macroscopic observations later. Nevertheless, we will mainly focus on simulation results and their interpretation in the next chapter because these show the dynamics in their entirety and do not look at just a fraction of the relevant processes.

In the final discussion (chapter 7), I will further elaborate on the social dilemma from a more comprehensive point of view.

6 Evolution of Cooperation and Defection

In this chapter, I will present simulation results that investigate the emergence of cooperation and defection in *ReCooDy* (see chapter 4) from an initially neutral and unconnected population. A strong focus lies in the results from the interplay of all implemented processes operating on similar time scales:

- the option of extracting high-risk, high-reward synergistic resources
- the possibility to survive without these high-risk resources
- the limited amount of resources provided by the environment
- the dynamic development of the interaction structure
- the agents' evolvable social environment modification potential.

How will the population self-organize and evolve if resources are limited and each action requires agents to pay resource costs? The imaginable space of possibility exceeds our intuitive understanding. Therefore, we will investigate and inspect simulation results, look at the observed phenomenology, and deduce hypothetical explanations of the observations. When possible, we will dive into these hypotheses and test them with further experiments as well as assure consistency with the analytical investigations (see chapter 5) and the overall expected process interaction consistency. As *Holovatch* et al. (2017) motivated, here, we will use computer simulations to investigate a system in which comprehensive analytical solutions are not obtainable. The results presented in the following represent a selection of the most critical findings and are a first step in exploring *ReCooDy*'s full capabilities.

Section 4.7 contains the initialization details for the population, the environment, and the chosen parameter regime. The accompanying Table 4.2 presents an overview of all initialization parameters used to create all simulation results throughout this chapter if not otherwise stated.

I implemented *ReCooDy* as a Utopia model (see chapter 3) and simulated it to obtain all the results presented in this chapter.

6.1 Overview

Synergy is the fundamental principle determining the evolution of cooperation and defection in *ReCooDy*. Synergies effectively transform agents' investments into public goods and bads (see section 4.2.2). Thus, how often and how much an agent invests determines its level of cooperation. We measure it directly through an agent's expected investment trait $\mu_i = i \cdot p_s$. We recall that it combines the agent's probability of extracting synergistic resources instead of basic ones with the magnitude of investment if interacting. Thus, it quantifies how much an agent is expected to invest on average within a time step. Large positive values correspond to high levels of cooperation and large negative values to high levels of defection (see section 4.2.2.4 for a more detailed classification). However, we should notice that there are different types of cooperation as high μ_i can either mean frequent but comparatively low investments or occasional but high investments. Therefore, if necessary, we will additionally inspect i or p_s additionally. Still, usually we explore the evolution of cooperation and defection mainly by observing the evolution of μ_i.

Figure 6.1 shows the expected investment μ_i in a trait evolution plot for all agents that lived throughout a simulation in birth order for varying synergy factors r in a two-dimensional histogram. To complement, Figure A.1 shows the corresponding i and p_s evolutions separately. Figure 6.1 provides a visual first impression of *ReCooDy*'s dynamics regimes for the chosen parameters and dependent on r, which we will explore throughout this chapter.

Minute synergy factors ($r = 1.01$ and $r = 1.1$) exhibit barely visible changes in μ_i compared to higher synergy factors and within the observed time frame. For $r = 1.01$, the expected investment μ_i remains around zero with positive and negative values for the first 1.5×10^6 agents (approximately 2.9×10^3 generations[1]). From that point on, only positive μ_i remain with a slight tendency to increase. It is more clearly visible in the i evolution (see Figure A.1). For $r = 1.1$, the evolution towards positive values happens earlier (around agent 3×10^5, i.e., after approximately 60 generations). The following trend of rising μ_i is more apparent, reaching values up to $\mu_i \approx 2$ at the end of the simulation. Even though low synergy factors maximize the personal social dilemma, as we recall from chapter 5, both simulations appear to show a slow emergence of cooperation with defective strategies dying out. Without any doubt, we need to verify that these observations are not mere fluctuations but systematic and seek explanations of the phenomenology. We investigate this regime of emerging cooperation in section 6.2.

Increasing the synergy factor ($r = 1.4$) reveals a transition regime with intricate dynamics. Agents quickly evolve exclusively positive μ_i with values spred within the range $0 < \mu_i < 8$. Around agent 1.1×10^6, a sudden transition happens showing a bifurcation into a high local density cooperator branch and a coexisting loner branch, which are stable for the next $\approx 3 \times 10^6$ agents. Then, the loners evolve into profiteers, i.e., start interacting more (see also p_s in Figure A.1), which eventually results in the collapse of the cooperator branch and the death of all benefactors around 4.5×10^6. Zooming into the region directly after the collapse reveals another bifurcation resulting in coexistence of cooperators ($\mu_i \approx 1$) and loners ($\mu_i \approx 0$). It is stable for only around 0.3×10^6 agents. Suddenly, the cooperators start evolving, increasing μ_i high levels of cooperation, $\mu_i \approx 8$ similar to before. Shortly after μ_i becomes maximal around agent 5.4×10^6, the loner branch again bifurcates. However, this time defectors emerge that become more and more defective with increasing time. At that point, the observed time-span ends, although we can already suspect the model to exhibit more dynamic patterns emerge and eventually collapse. Moreover, more than twice the total number of agents lived throughout the simulation compared to other r regimes. It indicates that the resources flowing into the system are distributed to sustain in total more agents. Thus, agents evolve to share resources among more individuals than in the other r regimes during specific phases. To understand all observations, we first need to understand the emergence regime. Afterward,

[1] Inspection of the data reveils that the number of agents is approximately constant in this regime around $N \approx 520$.

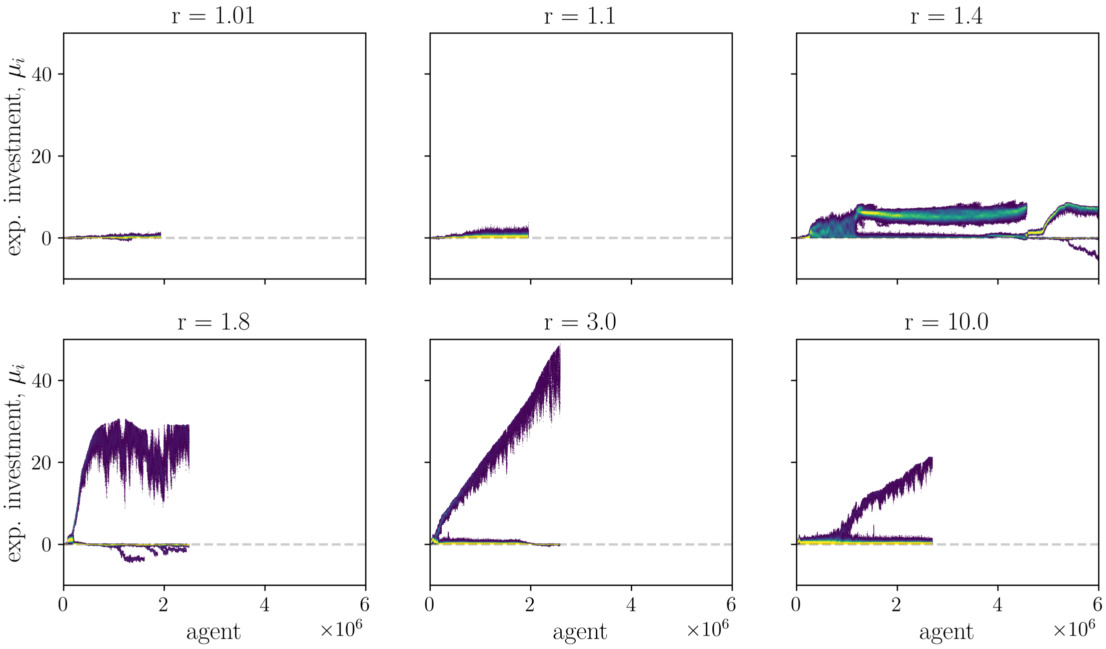


Figure 6.1: Trait evolution plots of the expected investment μ_i for varying synergy factors r that exemplify different dynamical regimes.

A trait evolution plot is a two-dimensional histogram of a quantity shown on the y-axis and agents on the x-axis, in their order of birth, which allows visualizing the evolutionary quantity change with respect to fixed-sized agent generations. Thus, it shows the evolution of a quantity in trait space, which corresponds to a non-linear time evolution. A trait evolution plot directly shows the total number of agents that lived throughout a simulation in the maximum value on the agent-axis that can be used for comparison between simulations. It shows the color-encoded local agent and trait density, respectively, from low densities (purple) to high (yellow) ones. Bins with counts equal to zero are always shown as white bins.

Each time evolution plot has 900 bins in both dimensions with bin counts higher than 300 shown as yellow bins. Each simulation ran for $T = 10^5$ steps. Figure A.1 complements this figure as it shows the evolution of i and p_s individually.

we explore this intricate transition regime in section 6.3, for which we already have observed the evolution of cooperation, defection, and the coexistence of varying strategy combinations that are metastable on different time scales.

Further increasing the synergy factor ($r = 1.8$, $r = 3$, and $r = 10$), results in a coexistence of cooperators and loners after an initial strategy bifurcation. For $r = 1.8$, the cooperator branch appears to remain approximately constant ($\mu_i \approx 20$), however, with signifant noice. If we look at i and p_s separately (see Figure A.1), we observe that the investments are focused while p_s exhibits variation. The higher the investments are, the more the variation of p_s will increase the spread of μ_i because we multiply higher investment values with approximately equally fluctuating probabilities p_s. From the loner branch occasionally defective strategy branches emerge (e.g. around agent 1×10^6 and 1.8×10^6) which however eventually die out. For a higher synergy factor of $r = 3$, the cooperator branch increases continuously in a linear-like manner. Interestingly, for even higher synergy factors of $r = 10$, thus, more beneficial cooperation regimes, the slope of the linear increase appears to be smaller. In section 6.4 we investigate this high synergy regime.

From these first simulation results, we already see that *ReCooDy* exhibits intricate phenomenology and intuitively unexpected dynamical behavior such as the emergence of cooperation even for minute synergy factors ($r = 1.01$), metastable coexistence of varying strategies, low-probability and high-impact events, and potentially lower levels of cooperation in high-synergy environments. To better understand how the observed phenomenology can result from the microscopic interactions, we investigate collective multiverse results of multiple system realizations as well as representative single universe runs. This way, we get more profound statistical results and their corresponding microscopic explanations.

6.2 Emergence of Cooperation

Without synergy ($r = 1.0$), agents' mean expected investments are approximately zero, except for a few (6) outlier realizations with highly negative values that start evolving after $t = 3.5 \times 10^5$. The latter are better visible in Figure A.2 showing the full value range. In general, we neither observe a strong selection towards cooperative strategies nor towards defective ones, except for the outliers. In more detail, investments usually evolve in a random-walk-like neutral way. The probability of extracting synergistic resources stays around zero (both are not shown here but revealed by data inspection). The evolutionary mechanism selects for agents that do not interact. However, due to mutations in p_s, they will sometimes interact by chance. When interacting, neither cooperation nor defection is usually strongly selected for if there are no synergies. With no positive and negative synergies, there is neither a personal gain in creating nor in destructing resources (compare section 4.2.2.3). Still, in a few cases, defection appears to be profitable. Investigating these systems in more detail would be an exciting endeavor. However, it is not the focus of this thesis because we will focus on systems that indeed provide synergies. It remains an open pathway for future work. Overall, statistically, we do not observe the systematic evolution of cooperation and the evolution of defection only in some realizations for no synergies.

For very low synergy factors ($1.01 < r < 1.17$), $\bar{\mu}_i$ evolves positive values that monotonically increase with time as well as r. The probability of synergistic resource extraction resources stays around $p_s \approx 0$ with fluctuations in the order of the mutations as inspection of the data reveils. Thus, the population seldomly interacts to extract synergistic resources. However, if they interact, they invest resources to create shared goods rather than selfishly grabbing, thereby destroying resources. Cooperation emerges. Here, the slow but steady emergence and evolution of cooperation relies on infrequent interactions with few agents. Interacting often with many agents comes hand in hand with a high risk of life-threatening exploitation because payoffs are small and need to be shared with others. Instead, when agents seldomly interact they can heavily rely on the basic resources for survival. If agents do not crucially depend on the outcome of interactions to compete and survive, it facilitates the evolution of positive investments. Another key to understanding why cooperators emerges rather than defectors comes from the interaction and the interaction structure itself: Although increasing individual investments usually results in smaller individual payoffs when interacting with others, giving birth to a cooperative offspring almost always creates a benefit for the parent if there are no defective neighbors (see chapter 5). Even more, in cases in which agents extract synergistic resources with no interacting neighbors, they gain a few resources from their investments $P_a = (r-1)i_a$. But recalling the cost of living $c_l = 0.1$ together with the evolving strength cost c_s (see section 4.3), we immediately see that agents initially, when investments are small, cannot sustain themselves only from synergistic resource extraction; They require contributions from others, too, as well as the safety from basic resources. However, occasionally extracting synergistic resources can be profitable enough to be selected for above a reached investment threshold. Still, agents cannot evolve frequent interactions because of the huge risk of being exploited in this r regime. Below, in section 6.2.1, we investigate the case $r = 1.01$ in more detail and see how the network itself self-organizes and whether agents indeed intend to minimize their connections, as we would expect for now. Summing up, we observe that cooperation emerges already for minute synergy factors; Evolution selects for selfless creation rather than selfish destruction in this *ReCooDy* regime.

Slightly higher synergies ($1.18 < r < 1.2$) show on average higher $\bar{\mu}_i^T$ but with a few clusters of final states, as well as several outliers. Outliers exhibit mostly positive $\bar{\mu}_i^T$ but in three case also extremely negative $\bar{\mu}_i^T$ up to ≈ -65 (see Figure A.2). Figure A.2 visualizes the whole value range to better illustrate the extreme scenarios. For $r = 1.2$, the time development of $\bar{\mu}_i$ first shows rising values that, however, reach a maximum and start shrinking again after $t \approx 3.5 \times 10^5$. We also observe a standard deviation from the different realizations, which is not present for lower sr, and which slightly increases with time. The observed variance is systematic due to the extreme outliers as well as the internal clustering structure exhibited in the final $\bar{\mu}_i^T$ distribution. We observe a transition of the system into the next regime, for which the microscopic organization is crucial to understand the observables (see section 6.3). These synergy factor results are included in the figure as references that reappear on the lower value range of the higher synergy regime (see later in Figure 6.4). For now, we recognize that averaged quantities are not sufficient to understand the overall behavior so far because simulations can yield significantly different system realizations, as the outliers indicate.

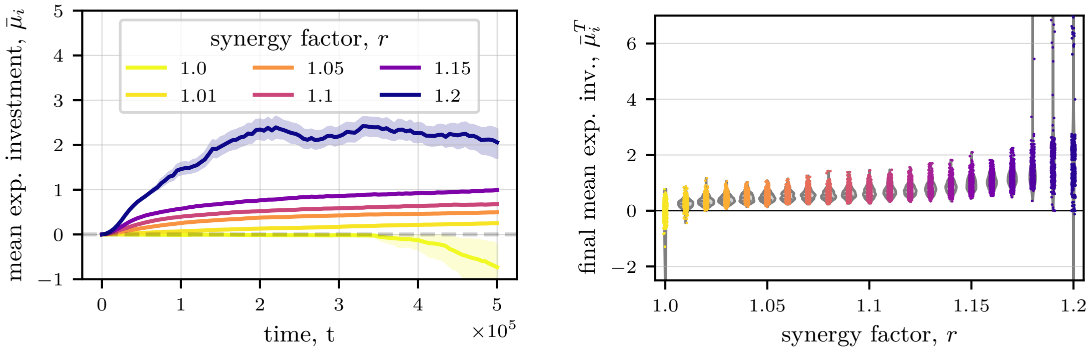


Figure 6.2: Time evolution of the mean expected investment $\bar{\mu}_i$ (left) and its final distribution $\bar{\mu}_i^T$ after $T = 5 \cdot 10^5$ time steps ($\approx 10^3$ generations) for varying synergy factor r (right) showing the emergence of cooperation regime. On the left, lines represent the mean and their shaded areas the standard deviation of $\bar{\mu}_i$ calculated over simulation runs. On the right, each of the 512 dots represents the final population-averaged expected investment $\bar{\mu}_i^T$ of a single system realization. Color encodes the synergy factor r from low values (yellow) to high ones (blue). The violin plots in the background show the distributions' kernel density estimates with width scaled to equal areas. The right figure shows a zoomed $\bar{\mu}_i^T$ value range. Figure A.2 shows the entire value range for completeness.

6.2.1 Emergence of Infrequent Cooperation in Sparse Effective Networks

Let us focus on the simulation results for a minute synergy factor of $r = 1.01$ that statistically already shows the emergence of cooperation, as seen before. Figure 6.3 visualizes simulation results of universe and multiverse runs for $r = 1.01$.

Rising investments. The evolution of investment i of a single model run shows increasing values over long times (top left in Figure 6.3). However, the increase is not strictly monotone and fluctuates, showing multiple phases of linearly increasing and decreasing i with each lasting in the order of 10^7 agents (≈ 2000 generations) long and changing i values within the range of 5 units. Zooming further into the structures reveals that branches of agent strategies frequently emerge that intend to separate themselves from the entire population. However, they die out quickly. If we would look at higher synergy factors within this dynamics regime (e.g., $r = 1.1$), we would see the rising investment more clearly with less and smaller decreasing evolution phases. All these observations indicate that the selection on i is very weak and acts over many times and generations; Still, investments and, therefore, cooperation rises.

Infrequent interactions. The corresponding evolution of the probability of extracting synergistic resources p_s (top right in Figure 6.3) shows that throughout the simulation, the majority of agents do not extract synergistic resources (yellowish area). The emerging horizontal line reflects the magnitude of p_s mutations ($O(0.05)$; see also Table 4.2). Only comparatively few have twice, thrice, or four times the p_s mutations, which we expect to see by chance from the random mutation process (greenish and purple areas above the yellow line). Thus, agents evolve to rarely interact with others, i.e., to rely on basic resources primarily.

Slight tendency to specialize in one resources. The middle left plot of Figure 6.3 visualizes the phase space of the expected investment μ_i and the expected strength $\mu_s := s \cdot p_s$. The latter describes the expected resources an agent effectively uses to extract basic resources over its lifetime. Most μ_s values are within the range 0.4 and 0.9. Lower values do not evolve because agents mainly compete for basic resources. After all, $p_s \approx 0$ and strength decide which agents indeed succeed in getting resources and which ones do not. On the other side, higher values than 0.9 are not sustainable because agents extract at most $j_b = 1$ resource per time while they need to pay a cost of living $c_l = 0.1$ and an evolving cost for having high strength $c_s = s$. Therefore, the evolved value range of μ_s corresponds to our expectations from the model formulation. Importantly, the phase-space distribution is skewed such that agents with high μ_i tend to have low μ_s and vise-versa. Thus, agents have a slight tendency to specialize themselves either in basic resources or synergistic resources. However, not entirely because living from synergistic resources alone yields too little to survive in this low-synergy regime. Nevertheless, on average higher expected investments allow for slightly reduced strength costs.

Sparse networks with power-law-like distributions. Let us shift our focus to the underlying network structure itself. In the bottom left of Figure 6.3, we see the degree distribution that appears to follow a power-law with a steep decrease over only 1.5 orders of magnitude. The plot accumulates results of the final network, which evolved after $T = 10^6$ time steps, from 512 universe runs with varying seed. The stacked colors encode the degree distributions of the underlying individual networks of each run. I applied a least-squares fit to extract the distribution's exponent: -4.09 ± 0.03. The observations show that most agents develop few links, but a few agents become hubs in the network. These hubs are still small because of the high exponent of the distribution. A combination of processes can explain the observations. First, the offspring inherits just a single link to their parent. All additional links have to build up during an agent's lifetime with a maximal addition rate of one link per time step. From the time snapshots, we, therefore, can expect many low-degree agents. Second, adding and removing links comes with a cost (here $\kappa_l = 1$, $\kappa_g = 1$, $\kappa_r = 1$) for the initiating agent. Agents that initiate linking actions can accumulate a significant resource cost and, thus, can have a disadvantage compared to receiving and more isolated agents. With low synergistic resource inflow ($r = 1.01$) and limited resources initiating agents can have a selective disadvantage, which helps explain why degree counts shrink rapidly with the increasing degree. Third, separated connected clusters can reconnect via global linking. The reconnection probability scales with the component size, introducing preferential attachment of larger connected clusters. We know that above a given threshold, a similar process can lead to an approximated scale-free connected component size distribution in the distribution's tail (*Yules* 1924). Further, percolation systems such as the simple forest fire model exhibit scale-free distributions reliant on the preferential connection of large clusters (*Bak* et al. 1987). And forth, link addition and removal exhibit trait-based preferential attachment and removal linking mechanism, respectively, and preferential attachment in the context of network growth

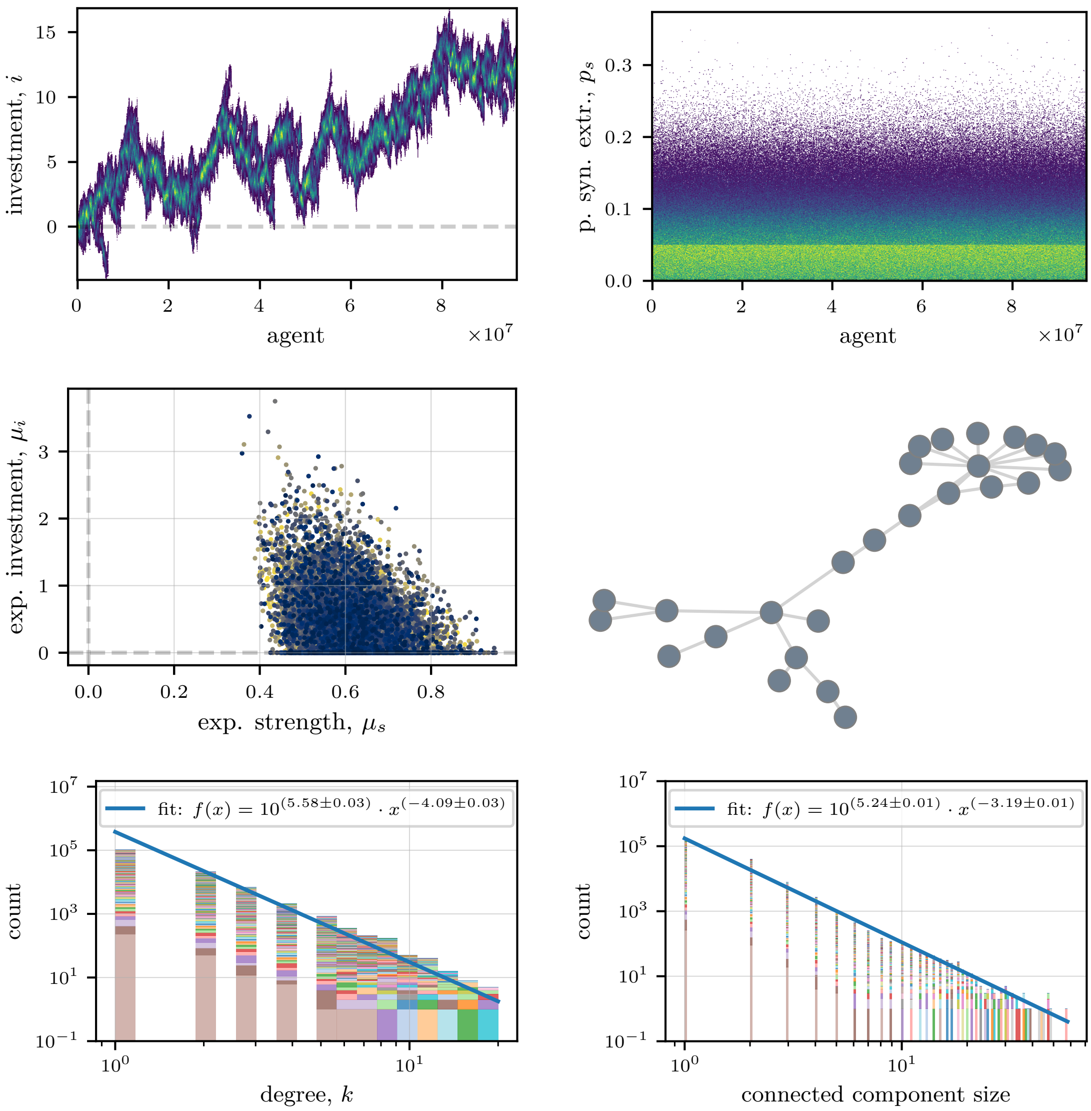


Figure 6.3: Simulation results for a minute synergy factor of $r = 1.01$ showing the slow but steady emergence of infrequent cooperation in sparse networks. It shows trait evolution plots of the investment trait i (top left) and the probability of extracting synergistic resources p_s (top right) with 900 bins in each dimension and bin counts higher than 100 for i and 20 for p_s shown in yellow. The middle left shows the phase space of the expected investment μ_i and the expected strength μ_s for the last 10^6 agents with dots representing single agents and color encoding time from early (yellow) until the simulation end (blue). The middle right shows a final largest connected component example for a population. The bottom left visualizes the final degree distribution, and the bottom right the final connected component size distribution over 128 simulation runs, each with color encoding different simulation runs. Blue lines show least-squares fit results of a power-law function. All simulations ran for $T = 10^6$ time steps for the single runs and $T = 2 \times 10^6$ for the distributions. See the caption in Figure 6.1 for the explanation of a trait evolution plot.

can result in scale-free degree distributions (*Barabási and Albert* 1999). Agents with desirable traits within their local and global social environment preferably receive links from others. With trait-based preferential attachment linking, we can expect a power-law degree distribution.
Complementing the power-law degree distribution, we observe a power-law distribution of connected component sizes (see bottom right Figure 6.3). A connected component is an induced subgraph, for which all vertices are connected via paths. There are no additional connections to other vertices of the original graph. The figure consists of the same simulation data as for the degree distribution before: 512 model realizations and the connected components calculated for the last time step $T = 10^6$. An example largest connected component of a corresponding single universe run is shown in Figure 6.3, too. Its number of vertices/agents gives the size of the connected component. The figure visualizes the sizes of the connected components and their accumulated count. Color encodes the results of single system realizations. Again, we observe a power-law-like distribution over approximately 1.5 orders of magnitude. Actually fitting a power-law (blue line) to the data using a least-squares method yields the corresponding exponent -3.19 ± 0.01. The same mechanisms that explain the power-law degree distribution also explain the connected components size distribution: the lifetime dependent development of the social environment, the resource costs to add and remove links, and most notably, the trait-based preferential attachment linking mechanism.

6.2.2 Summary

Cooperation emerges and evolves almost always for minute synergies—defection does not. Although we observe high positive investments evolving over long times, the probability of interacting and actually investing resources to extract synergistic resources and the number of interaction partners both evolve to small values. Further, the network structure evolves to be sparse with most agents having no or just a few neighbors following a power-law-like degree distribution with steep slope (exponent: -4.09 ± 0.03) and a power-law-like connected cluster distribution (exponent: -3.19 ± 0.01). We can explain the power-law-like distributions through the preferential attachment mechanism partly inherent in the linking mechanism (see section 4.4). In total, mostly isolated or low-connected agents evolve in this emergence regime. Together with the low p_s values, we recognize that populations consist of minimal, sparse, and generally low-connected effective networks. Thus, evolution selects for agents that minimize their personal social environments and thus their personal social dilemma (see the analytical considerations in chapter 5). Still, creating offspring usually increases an agent's personal payoff even for such minute synergies if the parent is a cooperator and as long as there is no very defective neighbor.

Specializing solely in synergistic resources, however, is not a surviving strategy. Even though agents tend to specialize either in basic or synergistic resource extraction, the risk is too high and the reward too low within this low synergy dynamics regime to survive. Overall, we observe the emergence of cooperators already for minute synergies that infrequently interact with few agents and profit from creating cooperative offspring.

6.3 Transition Regime

Figure 6.4 shows simulation results for synergy factors in the range $1.2 < r < 2.0$. The top shows the time evolution of the mean expected investment $\bar{\mu}_i$ and the total number of agents N for four selected synergy factors. The mean and standard deviation are calculated over the 512 different system realizations generated by varying the random number generator seed. The bottom shows the corresponding final distribution of the mean expected investments of single system realizations depending on the synergy factor r in the whole value range and a zoomed-in range. We notice that for $r = 1.2$ (yellow line) the expected investment μ_i increases with time until $t \approx 2 \times 10^5$. Increasing the synergy factor to $r = 1.4$ shows an initially steeper increase in $\bar{\mu}_i$ and already at $t \approx 5 \times 10^4$ the mean expected investment $\bar{\mu}_i$ reaches a maximum and starts continuously decreasing, thereby reaching ever more negative values with increasing standard deviation. Further increasing the synergy factor to $r = 1.8$ completely changes the dynamics, again. After the initialization phase with rising $\bar{\mu}_i$, the population reaches a roughly constant mean expected investment of $\bar{\mu}_i \approx 2$ with a slight tendency of decreasing values throughout the following time. The standard deviation is no longer visible, indicating that differences between system realizations for equal r are minimal on the observable scale. For $r = 2.0$, the population exhibits the same general $\bar{\mu}_i$ evolution, however, without the tendency of decreasing values.

When we shift our focus to the final distribution of expected investments $\bar{\mu}_i^T$ (bottom left and right plot of Figure 6.4) we start to understand what leads to decreasing $\bar{\mu}_i$ over time together with increasing standard deviations for low r. For synergy factors in the range $1.2 < r < 1.5$, we observe single system realizations with exceedingly negative $\bar{\mu}_i^T$ with values ranging from approximately -10 to -145. The kernel density estimate represented by the shaded violinplots enables us to approximate the $\bar{\mu}_i^T$ distribution. For $r = 1.3$, most very defective populations arise, i.e., the probability of highly defective populations emerging is highest. Furthermore, most defective populations are extremely defective. Synergy factors in the range $1.6 < r < 2.0$ do not exhibit populations with highly negative $\bar{\mu}_i^T$, i.e., highly defective populations anymore.

Looking at the bulk of final states (bottom right of Figure 6.4), we notice that for the observed synergy factors $1.2 < r < 2.0$ most final mean expected investments $\bar{\mu}_i^T$ are positive, i.e., usually cooperative populations evolve. Interestingly, for $r = 1.2$, a few simulations show highly positive outliers with up to $\bar{\mu}_i^T \approx 12$. Moreover, the results show clustering into subgroups of similar final values, which we can observe in the violinplots and the grouped dots, dependent on r. Unintuitively, the lowest synergy factor in the observed range ($r = 1.2$) results in the highest observed $\bar{\mu}_i^T$ values, as we can see in the medians and quartiles inside the violins. Higher synergy factors result in lower median $\bar{\mu}_i^T$ up to $r = 1.8$. For $r = 1.8$, $\bar{\mu}_i^T$ appear to bifurcate, showing two density peaks at $\bar{\mu}_i^T \approx 1.5$ and $\bar{\mu}_i^T \approx 2.0$ (Figure A.3 exhibits the bifurcation more clearly in the respective investment trait). Further increasing the synergy factor to $r = 1.9$ and $r = 2.0$ lets only the more positive branch survive, effectively increasing the median again. However, medians at $r = 1.9$ and $r = 2.0$ are lower than for $r = 1.2$. Most surprisingly, within the observed synergy factor range $1.2 < r < 2.0$, the most cooperative populations arise on average for the lowest synergies.

Furthermore, during most of the time, the lowest synergy factor ($r = 1.2$) sustains the highest number of agents during most of the simulation time (see Figure A.3). However,

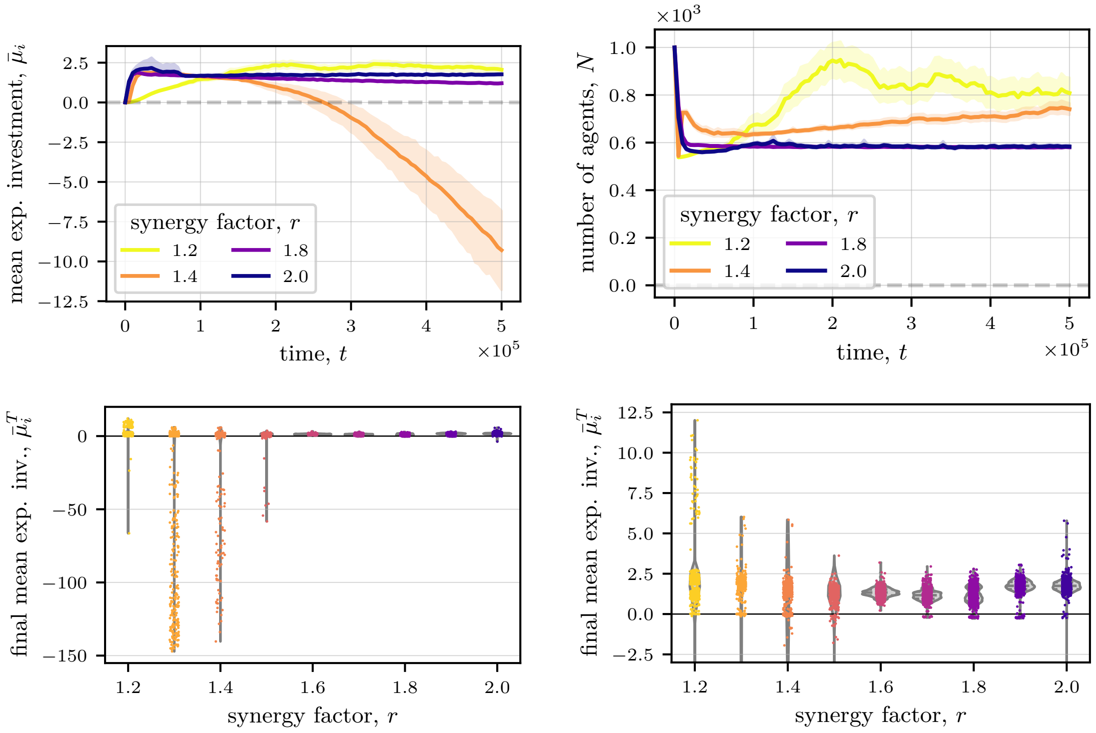


Figure 6.4: The transition regime. The top row shows the time evolution of the mean expected investment $\bar{\mu}_i$ (top left) and the number of agents N (top right) for different synergy factors r. The mean and standard deviation (shaded area) are calculated over system realizations. The bottom row shows the final mean expected investment $\bar{\mu}_i^T$ after $T = 5 \cdot 10^5$ time steps in the whole value range (bottom left) and zoomed (bottom right). Each dot represents the final population average of one system realization, 512 per r. Color encodes rising synergy factors from low (yellow) to high values (blue). The violin plots in the background show the kernel density estimates of the distributions with width scaled to equal areas. Lines within the violins show the median and quartiles of the corresponding distribution. Figure A.3 complements this figure by showing the time evolution of the number of agent and the trait evolution plots of $\bar{i}^T$ and $\bar{p}_s^T$.

with increasing defection for $r = 1.4$ also the number of agents increases. In general, these results indicate that if agents extract more resources either through cooperation or through defection, the system can sustain more agents.

All of the observations indicate that the system exhibits intricate dynamics dependent not only on the synergy factor but also on the specific realization. Just changing the sequence of random numbers for a specific synergy factor can lead to entirely different system states at a given time. In other words, the system's evolved state, i.e., its history, significantly determines the evolutionary pathway it will follow, at least for the observed timeline. The results indicate that looking at averaged quantities can be misleading because it hides and oversimplifies the underlying structure observed in the final distributions. We can only explain the observations so far if we explore the microscopic organization of single realizations of representative runs from the observed clustered final states in more detail, which we will do in the following.

6.3.1 Rich Dynamics

In the following, we will focus on simulation runs with a synergy factor of $r = 1.4$ as the representative example for the diverse dynamics characterizing the transition regime.

Figure 6.5 shows selected universe runs that present a phenomenological overview for simulation runs with a synergy factor of $r = 1.4$ and varying seeds. Each subplot shows the expected investment μ_i plotted against the corresponding agent. The ordering corresponds to changes in the observed phenomenology. Color encodes local density from low (purple) to high (yellow) by having both quantities binned into 500 bins each and displayed as a two-dimensional histogram. Bin counts that exceed 500 are capped and shown in yellow.

Coexistence. We usually observe strategy coexistence that seems to be metastable on varying times scales. The simulation for seeds 45 and 42 show that strategies can bifurcate directly after initialization into a cooperative branch at $\mu_i \approx 9 \pm 3$ and a neutral branch at $\mu_i \approx 0$ that coexist during the whole simulation time. Agents specialize either in basic or synergistic resource extraction. Here, cooperators not only invest positive amounts but also frequently interact as inspection of the p_s evolution reveals. In contrast to the previous emergence of cooperation regime (see section 6.2 for comparison), agents can and do sustain themselves from synergistic resource extraction alone. The benefit from synergies outweighs the risk of interacting with these specialized agents. However, neither too high nor too low expected investments relative to the other cooperators emerge; The expected investments do not deviate more than ± 3 from the cooperators' average. It seems that evolving higher investment first yields comparably higher returns making the strategy evolutionary successful until investments rise so high that agents already extract all synergistic resources. In these specific population settings, evolving even higher investments comes with the risk of not receiving resources anymore such that investments reach the observed maximum. The neutral agents (loners with $\mu_i \approx 0$) usually do not extract synergistic resources. Instead, they compete for the basic resource through strength, which can sustain a limited number of agents. Frequently, defective strategy branches emerge from the neutral branch with mainly decreasing μ_i that survive for up to 10^3 generations ($\approx 6 \times 10^6$ agents) but typically less before dying out. Here, defection is a metastable strategy on medium time scales that frequently dies out, however.

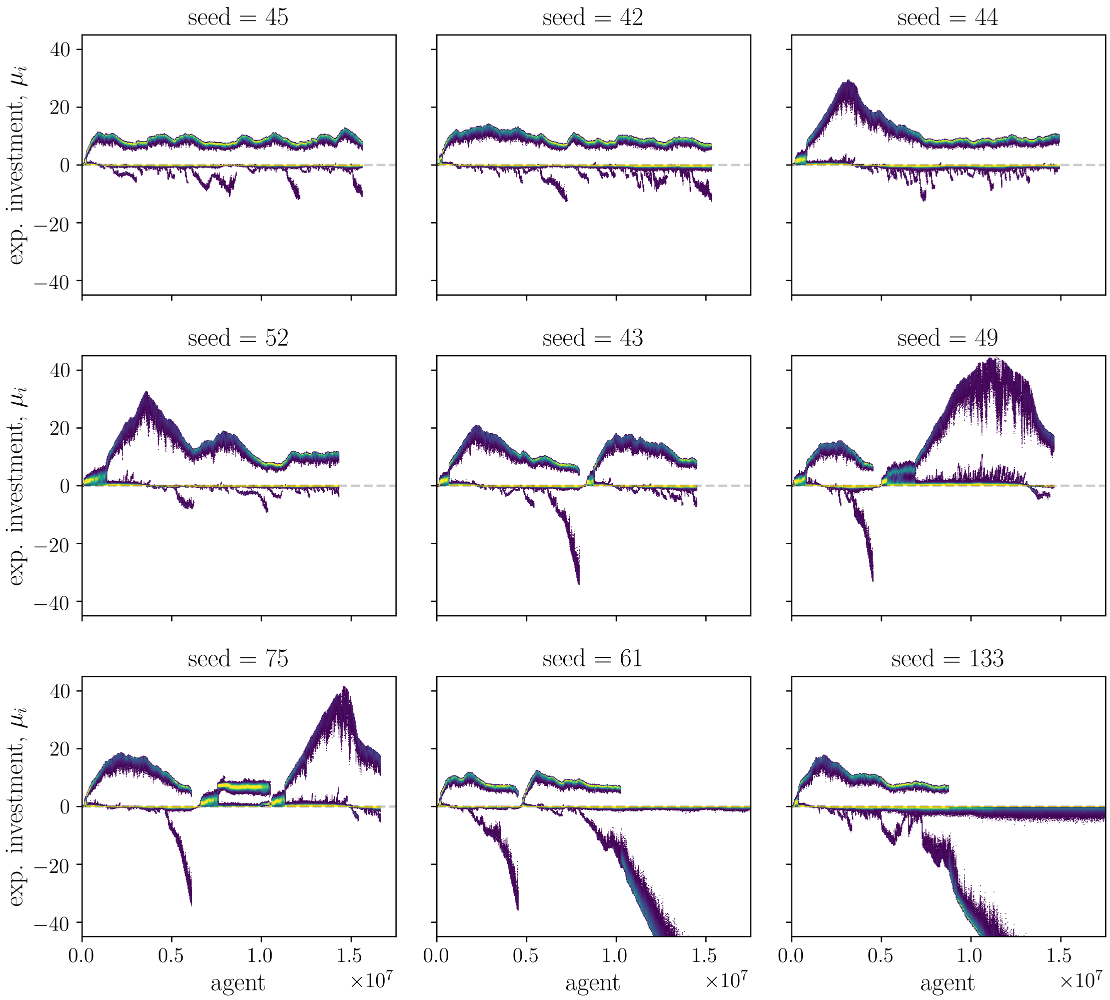


Figure 6.5: Trait evolution plots of the expected investments μ_i that exemplify the rich dynamics of the transition regime. Simulation results for $r = 1.4$ and nine selected seeds representing typical dynamical phases are shown. Each plot contains 500 bins in each dimension with counts higher than 500 shown in yellow. Each simulation ran for $T = 10^5$ steps. See Figure A.4 for the entire value range and Figure A.5 for the corresponding time evolution of the number of agents and the final degree distribution. See the caption in Figure 6.1 for the explanation of a trait evolution plot.

Mountain-shape-like dynamic structures. The simulation result for seed 44 illustrates a recurring dynamics pattern: Agent's expected investments μ_i rise in an approximately linear way to very high values (here $\mu_i \approx 30$) before they reach a maximum and start decreasing again until they reach values around $\mu_i \approx 9 \pm 3$. This mountain-shaped-like dynamics pattern is probably an overshoot event that the agent's linking traits could cause if evolving such that investing too much becomes an evolutionary disadvantage. However, the precise cause would need to be explored further. The mountain-shape-like dynamic patterns not only occur directly after initialization (seed 44) but can also have double-peaks (seed 52), follow directly after a synchronous collapse of a cooperative and a defective branch (seeds 43 and 49), follow after the collapse of a highly cooperative branch (seed 75), and last longer while reaching high values (seed 49).

High-density cooperation. Seed 75 shows another dynamics pattern around agents 1.5×10^6 to 2.0×10^6 exhibiting very high cooperator density. Not only does the cooperator density increase significantly but also the total number of agents more than triples ($N \approx 2200$ compared to $N \approx 600$ (see Figure A.5)). The number of links increases dramatically, as well as we can see in the degree distribution. Agents evolve linking strategies that modify their social environments and enable more agents to live at a time. More agents can live from the extracted synergistic resources despite their increased linking cost; thus, they partition resources more equally among all living agents. However, this dynamical phase is only metastable and collapses after approximately 3×10^7 agents, again. Below in section 6.3.2, we will explore this regime and its collapse in more detail.

Ever growing defection. The simulation results for seeds 61 and 133 show a phase of ever-growing defection at the end. Highly negative values are not shown to better visualize the other dynamical phases but Figure A.4 shows the complete value range for completeness. We can see that after the collapse of a cooperative branch, the defective branch evolves ever lower μ_i over time. We will explore this transition into ever more defection in section 6.3.3 and address its triggers, its stability, and whether it is the final system state.

Investments shrink with emerging defection but Cooperators increase. In general, we observe the tendency that if defection emerges and grows through a decreasing μ_i branch, its values decrease in the cooperative branch. Furthermore, if we focus on the color-encoded local density, we observe increased local densities in the cooperative branch, the lower the values of μ_i get. We see this pattern for all seeds. The number of agents stays roughly the same around $N = 600$ during these dynamical phases (see Figure A.5). Thus, if defectors emerge, the relative number of cooperators increases but the contributions of cooperators decrease. We can explain the correlated decrease of the defective and cooperative branches through the nature of the interactions: defectors profit most when the investment difference is high. Thus, they have an advantage if they lower their investments, as long as they have cooperators with which to interact. In contrast, cooperators lose less from interactions with defectors if the difference in investment is low. Thus, cooperators lowering their μ_i as a response to lower μ_i from defectors could explain the correlated decrease in expected investments.

The increased number of cooperators for lower μ_i is most probably related to the limited amount of extractable synergistic resource $\mathcal{A}_s$ per time step. It is partitioned among all interacting agents according to the success of their interactions (recall the finite resources described in section 4.2.2.5). Inspection of the data reveals that indeed all synergistic resources are extracted within at the respective times. The payoff individuals may receive scales with the agent's investment. Thus, with less investment per agent, potentially more agents receive a share of the limited resources. Moreover, the number of defectors is very small compared to the number of cooperators (all purple versus yellowish bins), indicating that the effect from decreasing μ_i of the defectors is smaller than for cooperators. The observations indicate that the presence of a few defectors indirectly increases the number of cooperators by letting them evolve smaller investments such that more agents profit; More agents profit but with lower shares from cooperatively extracted resources.

Slight Profiteers of rising cooperation. Interestingly, during the phases of linear-like increases in μ_i, a few agents emerge with low but positive μ_i. They appear to be slight profiteers that occasionally profit from interactions with cooperators. However, as soon as agents extract all synergistic resources within a time step, this strategy does not appear to be successful anymore, most probably because it yields too little payoff. When $\bar{\mu}_i$ drops again after a maximum in the cooperative branch, a few agents emerge with low but negative μ_i, as described before for seeds 45 and 42. In rising cooperation times, a minority of profiteers emerges and survives, while at decreasing cooperation times, a minority of exploiters emerges and survives. Whether this is a general obervation or an artifact and why exactly this is the case requires further investigation.

Collective starting phase. At the start of the described dynamic patterns, we usually observe a trigger phase, in which the whole population evolves low but positive μ_i for $O(30)$ generations (roughly 2×10^5 agents). It is before they bifurcate into a cooperative and a neutral branch that, for example, triggers the mountain-shape-like dynamic pattern. Here, agents already evolve positive investments, but they cannot survive on synergistic resources alone because the overall investments and resulting payoffs are too small. They first need to evolve high-enough investments to evolve $p_s \approx 1$, i.e., interact frequently and be able to survive from the resulting payoffs. In section 6.3.2 we will explore and explain the bifurcations following after the initial collective phase in more detail.

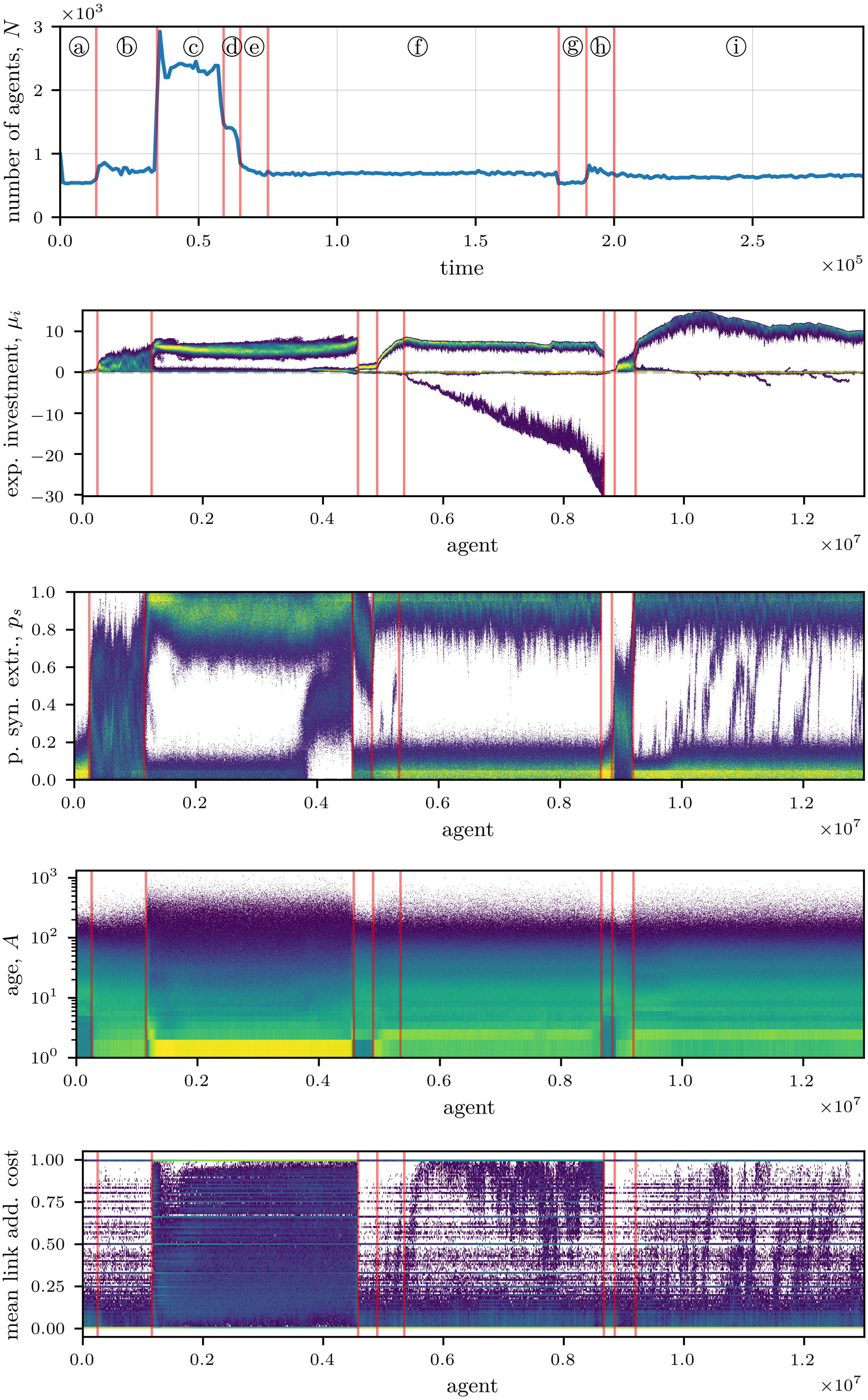
6.3.2 Dynamical Phases and their Transitions

We have already seen that simulations yield a small set of typical dynamical phases that emerge, survive temporarily, and transition to the next phase mostly through sudden collapses. Here, we will take a representative single simulation run and explore such dynamics regimes, their observables, and sudden transitions in more detail by investigating more evolved traits. Figure 6.6 shows simulation results from a single model realization. The subplots show the number of agents N over time and two-dimensional histograms of the agents' expected investments μ_i, their probability to extract synergistic resources p_s, their final ages A, the lifetime-averaged cost paid for link addition and removal, respectively, and the link modes, which determine the targetted quantity for global and local link addition and link removal, respectively. The counts of the link addition mode are weighted with the probabilities p_l and p_g, respectively, to account for the fact that within a time step, links are either added in the local or global social environment, never both. I.e., they show the weighted average counts for local or global link addition. Color encodes the local density of states in the trait space from low (purple) to high (yellow) values. The local density is linearly scaled for the expected investment and all link modes, and logarithmically scaled for the age and the accumulated age-normalized linking costs. The horizontal grey line in the mean link removal cost plot shows an average link removal cost of 1, i.e., one link removal action per time step. Vertical red lines indicate transitions between dynamical regimes. We will now go through these individual dynamics regimes and their transitions, describe the observed phenomenology, try to derive explanations for the observations, and extract results.

Initialization phase ⓐ The simulation begins with a spin-up and initialization phase. The system holds $N \approx 500$ agents corresponding to the amount of basic resources $\mathcal{A}_b$ available per time step and the extraction of $j_b = 1$ per agent. The expected investments start at $\mu_i = 0$ and slowly rise. If we inspect the investments i and probabilities to extract synergistic resources p_s individually, we notice rising investments but with $p_s \approx 0$. Most agents have minimal mean cost for link addition and removal and preferably do not link to others. We know that initially, all agents are neutral, i.e, $p_s = 0$ and $i = 0$ for all agents. Investing in linking will be a waste of resources because the rare interactions to extract the synergistic resources combined with the low investments will not be able to compensate for the linking costs. Thus, agents that do not waste resources on linking do have an evolutionary advantage; Evolution selects for agents that rarely link from the initially linking population. With such low investments, agents cannot sustain themselves only from the synergistic resource alone. The interactions cannot generate enough payoffs to compensate for the agent's cost of living and strength. However, in the rare occasions that they interact by chance, induced by mutations of p_s, investing positive investments is the more successful strategy. Importantly, agents do not vitally depend on high success in the interactions because of the rare interactions, which in turn lowers the potential risk of getting exploited from defectors. In this way, the initialization phase resembles the dynamics of the emergence regime (see section 6.2) as similar arguments apply.

More frequent interactions ⓑ As investments slowly but steadily increase, the system reaches a threshold with a rapid transition to a new dynamical phase characterized by slightly more agents, $N \approx 600$, and expected investments with values approximately an order of magnitude higher than before. The latter is a direct result of rapidly increasing p_s with values spanning the whole range $0 < p_s < 1$ but most found in $0 < p_s < 0.8$. The investments themselves, however, evolve slower towards higher values after an initial instantaneous selection for high investments directly at the transition (see Figure A.6). After the transition, we observe, on average, a slight decrease in the age A as well as a vast increase in the number of agents that die immediately (recall the logarithmic color-scale). The cost for link addition slightly decreases, and nearly no agent pays a cost to remove links. The few agents that add links do so preferably to a random target agent within their social environment. As already said, link removal does not occur; thus, the link removal mode evolves randomly despite the high density of apparently payoff-focussed link removal mode for the observed times.

The observations indicate that the system exceeded a threshold allowing the transition into a dynamics regime of more frequent interactions and slowly increasing cooperation


×10^3
number of agents, N
3
2
1
0
ⓐ
ⓑ
ⓒ
ⓓ
ⓔ
ⓕ
ⓖ
ⓗ
ⓘ
0.0
0.5
1.0
1.5
2.0
2.5
time
×10^5
exp. investment, μ_i
10
0
−10
−20
−30
0.0
0.2
0.4
0.6
0.8
1.0
1.2
agent
×10^7
p. syn. extr., p_s
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
agent
×10^7
age, A
10^3
10^2
10^1
10^0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
agent
×10^7
mean link add. cost
1.00
0.75
0.50
0.25
0.00
0.0
0.2
0.4
0.6
0.8
1.0
1.2
agent
×10^7

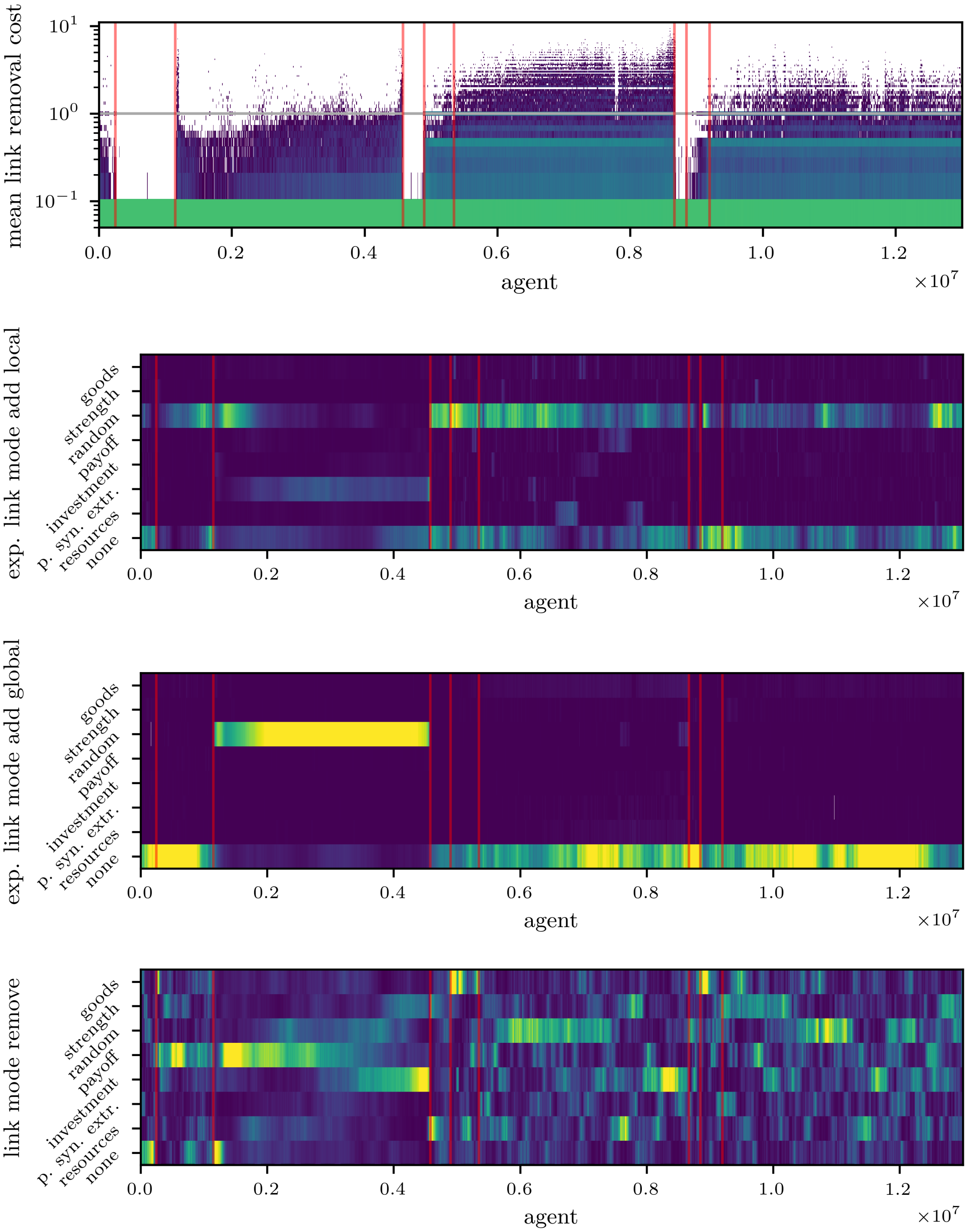


Figure 6.6: Trait evolution plots of the most important quantities and the time evolution of the number of agents (top) that show examples of dynamical phases in the transition regime in detail. Circled letters represent dynamical phases referred to in the text. The simulation ran for $T = 5 \times 10^5$ steps. Figure A.6 complements this figure by showing further trait evolution plots. See the caption in Figure 6.1 for the explanation of a trait evolution plot.

throughout the population. The system can sustain ≈ 100 more agents from the synergistic resources alone. The payoff from synergistic resource extraction outweighs the risk of investing and getting exploited for a successful subset of agents. Other more unfortunate agents die immediately, as the age distribution indicates due to the high exploitation risk of investing. If agents invest directly after their birth by chance, it adds an additional resource dissipation term that can trigger an immediate death if they are not successful compared to others. For the more fortunate agents, the investments reach high enough values to produce payoffs that allow survival and eventual reproduction. We observe a manifestation of the high-risk nature of synergistic resource extraction section 4.2.2. Here, we do not yet observe the high-reward nature directly because agents do not entirely rely on synergistic resources yet and payoffs only allow survival without agents actively tailoring their social environment. Still, agents receive a reward from extracting synergistic resources, which elevates the system's carrying capacity to sustain ≈ 100 more agents. Living from synergistic resources alone seems not to be possible in this phase because we do not observe cooperators wholly specialized in synergistic resources with $p_s \approx 1$. Still, agents start interacting more frequently, thereby extracting more synergistic resources and increasing the system's carrying capacity. However, the risk of interacting with the relatively low payoffs still prevents entire specialization.

Fully cooperative population ⓒ As the maximal expected investment slowly increases over time, a rapid transition towards a complete cooperative dynamical phase happens around agent 1.15×10^6, when the system reaches a threshold. This phase exhibits a large population with approximately thrice the previous number of agents ($N \approx 2300$ v.s. $N \approx 750$) that prevails for approximately 1430 generations ($\approx 3.43 \times 10^6 / 2.4 \times 10^3$) or 5×10^5 time steps. The transition comes with a bifurcation of expected investments into a cooperative branch, reaching around $\mu_i \approx 6$ with huge local strategy densities, and a neutral branch around $\mu_i \approx 0$ with at most $\mu_i \approx 2$. The transition takes some 10 generations with overshooting values until the cooperative branch gets focussed. During the entire phase, the cooperative branch evolves a broader μ_i distribution that decreases in the first half on average but then starts increasing again. Before the next transition and shortly after μ_i starts rising on average, we observe an increased local density of the neutral branch. Indeed, if we inspect the p_s evolution, we see that all formally neutral agents, i.e., the ones completely specialized on basic resources with $p_s \approx 0$, all quickly evolve values around $p_s \approx 0.4$ with a continuing increasing trend until the regime collapse happens. Thus, the neutral branch evolves into a profiteer branch. When we look at the age distribution, we see that most agents die directly after getting born. A few agents, however, reach significantly higher ages (recall the logarithmic scaling). During the fully cooperative population phase, we see significantly higher mean link addition costs as well as moderate mean link removal costs for a small subset of agents. However, most still not removing links at all. Interestingly, links are mainly created to globally selected targets chosen at random. However, at the beginning of this dynamical phase, a subset of the population chooses random targets within the local social environment. Still, it switches to select for agents with high p_s, which maximizes directly before the collapse of cooperation. At the beginning of the fully cooperative population phase, links are rarely removed, followed by focusing on removing links to agents that receive a small payoff. In the second half of this dynamical phase, agents start to increasingly remove links to agents with low investment, which maximizes shortly before the dynamics collapse. Overall we see other link modes also emerge and die out. However, from the mean link removal costs, we can derive that most agents do not reach their link removal threshold, thus, do not indeed remove links. Such a situation makes the link removal mode meaningless, i.e., we mainly observe neutral evolution overlaid with a signal from the agents that indeed remove links.

How can we interpret and explain these observations? This dynamics regime starts after a small subset of agents exceeds a μ_i threshold that enables enough return to allow agents to specialize on the synergistic resources entirely. With the specialization, they can reduce their resources spent on strength if they evolve low strength through consecutive small mutations, as they indeed do, as inspection of the data reveals. Thus, they avoid competition for the basic resource. As we already know, cooperating parents usually profit from cooperative offspring (see section 5.5). But they need to pay for resources to create offspring. In this simulation, the total birth cost—creating offspring and transferring resources— equals 7. With $\mu_i \approx 6$ and $r = 1.4$, the parent of an interacting offspring receives an estimated payoff of $6 \cdot 1.4/2 = 4.2$ just from the offspring's investments into the shared interactions if there is only one link to the parent, which is the case after birth. Thus, in this scenario, the parent's net cost of giving birth is less than 3 after birth. If the offspring survives a second interaction, the parent's birth cost is already compensated, effectively yielding a profit. In general, the higher μ_i gets, the lower the personal birth cost gets.

As the age distribution indicates, the offspring generally experiences a cruel regime in which it usually immediately dies because their theoretical payoff is too low to allow them actually to receive a share of the available synergistic resources. They are most of the time not successful enough in the interactions to survive. Still, some cooperators get the chance to survive the first few time steps and optimize their social environment enough to survive with the extracted synergistic resources. Random deaths, given by p_δ, occasionally open niches for new agents to take over and become successful. Surprisingly, this link optimization mostly happens randomly. Randomly choosing target agents is the only link mode that does not include a trait-based preferential attachment mechanism. When preferential linking occurs, multiple agents will link to the same (few) agents, effectively overwhelming them. The link mode is heritable; thus, if a link mode temporarily is successful, the offspring will most likely have the same link mode, which will increase the number of agents that will link to the same small subset of agents. Overwhelm can happen because a high individual degree can easily diminish payoffs (see section 5.2 and section 5.3). Agents can counteract such an invasion of incoming links by removing bad links directly afterward. Indeed agents remove links in this regime, as we observed. A few agents even remove more than one link per time step (agents above the grey line in the mean link removal cost plot). They first cut links to low payoff agents but later with low investment agents; Both modes encode optimizations for increased individual interaction outcomes. Receiving a lot of links and afterward removing bad links evolves as a strategy to optimize the individual social environment.

Removing links, however, is costly. Directly before the collapse of the cooperative branch, link removal costs slightly increase, which increases resource dissipation. Furthermore, local link addition, targetted at agents with high p_s, increases drastically. Recalling the emerging profiteer branch, we see that this change in linking mode also increases the probability of an agent to form links to a profiteer. The transition of neutral agents to profiteers and their rising influence, combined with the effectively increased probability of linking to harmful profiteers instead of beneficial cooperators eventually leads to the collapse of cooperation. The effects accumulate, resulting in exceeded thresholds, the invasion of cooperators, and their instantaneous death. Due to the cooperators' self-supporting nature, the death of a few key agents could result in weaker remaining cooperators and an eventual cascade effect, resulting in their extinction. The dynamics regime of a fully cooperative population exhibits the high-risk, high-reward nature of cooperation that eventually results in its collapse.

Low cooperation population ⓓ With the collapse of the highly cooperative branch, the number of agents drops to around $N \approx 1400$, and the former profiteers become the new benefactors that keep their expected investments for around 10^4 time steps (≈ 1400 generations) at low values of $\mu_i \approx 1$. Looking at the probability of synergistic resource extraction p_s we observe that the neutral branch $p_s \approx 0$ reappears and the cooperative branch slowly evolves to lower p_s. The age distribution resembles the initial one but with most agents dying after the second time step. Agents rarely spend resources on link addition and, except for three outliers, do not remove links. In each generation, it seems that the cooperators with comparably low p_s survive for a bit longer and create slightly more offspring, which results in the slow selection for lower p_s agents. Cooperation is metastable, and the nearly constant investments are sufficient to sustain many agents. However, due to the shrinking p_s, fewer agents interact per time, and fewer agents share the extracted resources, which eventually leads to the next dynamical phase. In this dynamics regime, a large population still sustains itself from cooperation; however, evolution selects agents that interact less often, which eventually leads to a transition into the next phase.

Rising cooperation ⓔ When p_s falls below a threshold, $p_s \approx 0.4$, the system transitions immediately into a dynamical phase of rising cooperation, i.e., increasing μ_i. The number of agents drops to $N \approx 600$, indicating that the extracted synergistic resources can sustain less than half of the population of the previous dynamical phase. Interestingly, the probability of extracting synergistic resources of the cooperative agents jumps back to $p_s \approx 1.0$. Now, we see more agents modifying their social environment, especially by removing links to agents that are located on nodes producing few goods. Thus, there are fewer cooperators that extract all the synergistic resources and specialize in them. Investing more and starting to optimize the social environment are both profitable strategies as soon as there are fewer cooperating competitors left.

Emerging defection ⓕ When the cooperative branch reaches its local maximum of μ_i, a defective branch emerges out of the neutral branch containing only a few defectors per generation that evolves increasingly low μ_i until its sudden collapse. This dynamics regime happens on comparably long time scales, lasting around $t = 10^5$ time steps or approximately 4×10^4 generations. If we look closely at the p_s distribution directly after the transition, we see a small branch emerging from low probabilities near $p_s \approx 0$ to high ones near $p_s \approx 1$ in just a few generations. These agents are the emerging defectors as data inspection reveals. A few defectors can successfully survive from synergistic resources alone by exploiting the cooperators. The population contains enough high-investment cooperators to cope with the destructed goods and to remain stable. However, multiple

defectors harm each other through resource destruction, effectively decreasing the resource gain from grabbing resources and keep the number of defectors as a result low. A defector with lower investment outcompetes a defector with comparably higher investments because it grabs comparably more resources, thereby increasing its personal payoff and competitive advantage. Selection happens for evermore defective agents within the defective branch. Eventually, the defectors get so exploitative that the cooperators cannot sustain them and themselves anymore, resulting in a collapse of both the cooperative and the defective branch. The defectors do not grab enough resources to survive on their own but require the aid of cooperators to survive.
When we look at the mean link addition cost, we observe a branch of rising values reaching average costs of 1, which widens its distribution with increasing times. Indeed, we can identify this subpattern as belonging to the defective branch. The defectors have enough resources to invest in link addition, which usually profits them because their destructed goods are shared, and they can keep their individual profit (see creating bads in section 4.2.2.2). In the mean link removal costs, we see that the agents, in general, remove links. A subset of agents even removes more than one links per time step on average. Link removal costs grow over time in the upper extreme reaching up to 10 links removed per time step on average for single agents directly before the collapse of the dynamics regime. What we observe is that agents optimize their social environment. Especially high-achieving cooperators invest a lot of resources in removing links, thus, optimizing their social environment to counteract the pressure from the emerging defectors. In summary, in this dynamical phase, the cooperators can sustain a small group of self-limiting defectors over a long time until the latter eventually cause a collapse of both cooperation and defection through their exploitation. Defection emerges naturally if stable cooperation exists.

Collapsed population resembles intial population ⓖ Directly after the collapse of the cooperative and defective branches, the population resembles the initial population in all key traits. Agents do not interact but slowly start growing positive investments. Linking does not happen frequently; There is no link removal and minimal link addition. Agents rely entirely on the basic resource and start evolving anew, as they did at the beginning of the simulation.

More frequent interactions ⓗ As before, agents quickly evolve more frequent interacts for synergistic resource extraction, $p_s \approx 0.3$. However, they cannot yet live from synergistic resources alone. The expected investments μ_i slowly evolve higher values. As soon as a few agents reach a threshold of μ_i, the transition to the next dynamical phase happens via a bifurcation into one cooperative and one neutral branch. Qualitatively, we observe the same dynamical phase as previously when interactions became more frequent. However, there are a few differences. Here, only approximately half the number of agents until the next following transition happens; thus, it lasts only half as long. On average, the spread and the mean of μ_i are lower. And a few agents already start to remove links, which does not happen previously. We observe structurally the same dynamics pattern, however, with differences in the specific realizations. The distribution of all agent traits is different compared to the initial population because the (linking) traits already evolved and self-organized once. In contrast, they were artificially set at the beginning. We can expect these historical remnants to have an impact on the exact realization. We see structurally similar, recurring dynamics patterns that transition to other dynamics regimes when the system-defining states and traits reach threshold values.

Cooperative population ⓘ The last observable dynamics regime is a metastable coexistence of a cooperative and a neutral branch. In contrast to the earlier cooperative dynamics regimes, here, we observe fewer cooperators but with higher expected investments, $\mu_i \approx 10$. Now and then, small groups of defectors emerge with similar characteristics as for the previous dynamics regime of the emerging defective branch; the probability of synergistic resource extraction p_s and the mean link addition costs both increase for the emerging defector subpopulation. However, within the observable time, the defective branches all die out eventually. They do not evolve highly negative expected investments. In general, we observe a state of coexisting cooperators and neutral agents with frequent unstable emergence of defectors.

Overall observations and results Overall, we observe multiple qualitatively different dynamics regimes that are metastable on varying time scales. However, none of the evolved strategies is safe from the invasion of an eventually evolving defective strategy. Only for the last cooperative dynamical phase can we not yet make a statement. However, in section 6.3.3, we will see that it will eventually collapse, too. Often, transitions from one dynamical phase into another one are triggered by sudden population collapses. These collapses are the result of exceeded thresholds making it impossible for agents to survive with the extracted resources. Such Exceeded thresholds can easily trigger a death cascade because cooperators rely on each other to extract enough resources to survive. They are self-supportive. As soon as key positions in the self-organized network break down, it could lead to the sudden cascaded death of many cooperators. Further, within the observed dynamical phases, defectors could not survive on their own because someone else needs to pay for the destructed goods such that enough profits remain for them to survive. Therefore, a collapse of cooperation can lead to a cascaded collapse of defection, too. The latter would only be preventable if defectors would have low enough investments such that their personal benefit outweighs the loss due to the destructed goods. Indeed, as we will see in section 6.3.3, such a scenario also happens occasionally. The simulation results show that transitions rarely happen, typically only after hundreds or thousands of generations. Between those transitions, the dynamical phases are metastable. The scarcity of the transitions implies that they require specific microscopic configurations that first must evolve and manifest themselves.

6.3.3 Long-term Evolution – Defective Attractor

Until now, we focussed on medium time scales with simulations of around $T = 5 \times 10^5$ time steps with typically $\approx 1.5 \times 10^7$ agents and some ten thousand generations. They predominantly exhibit metastable cooperative strategies with sudden transitions between regimes. Now, we will shift our focus and look at simulations that run for approximately two orders of magnitude longer. Figure 6.7 shows the results of a multiverse run with 128 different system realizations (seeds) that differ only in their random number sequence. The orange line denotes the time $t = 5 \times 10^5$, the typical final time in the previous section. We observe that the model lives through three consecutive phases denoted with circled numbers. The figure also includes a representative single universe run that exemplifies all three phases (top). In the following, we will explore these phases.

Phase ①: cooperative populations. Let us focus on the middle right plot. The first phase exhibits, on average, primarily cooperative populations with $\bar{\mu}_i \approx 2$. We explored this phase previously in section 6.3.2 and know that it exhibits intricate dynamics with several different metastable phases on low to medium time scales with sudden transitions between them. All of the intricacies are hidden within the population-averaged $\bar{\mu}_i$ quantity and its variation. As the orange line indicates the final time of the mentioned section, we notice that agents can live through many different medium time-scale cooperative dynamical phases before they eventually transition to the next phase ②. We observe a single system realization that survives even for more than 1.5×10^7 time steps. Still, no cooperative population survives on the observed long time scales; Eventually, all leave their metastable cooperative attractors and transition to a phase of growing defection.

Phase ②: ever growing defection. We see a sudden transition towards an ever more defective population best seen in the middle left and top subfigures. We can recall that already in Figure 6.5, we observed for seeds 61 and 133 two such collapses of the cooperative branch that triggered the evolution towards a growingly defective population. Here, first of all, we observe that the growing defection phase is metastable over long simulation times because even after 2×10^7 time steps, the majority of systems (107/128) are still in it (see bottom subfigure). With increasing time, μ_i decreases its rate of decreasing values. Thus, the more defective the population becomes, the less evolutionary pressure the agents experience on average per generation to become even more defective. With limited available resources, the more defective agents get, the fewer defectors can survive with their strategy within one generation. More defective agents grab larger individual amounts leaving less for others. If the number of defectors per generation shrinks, the rate of the μ_i decrease within the top trait-space-based subfigure decreases. For now, we have mainly described macroscopic observations gained from the 128 system realizations. In order to explain and interpret them adequately, we need to explore the microscopic system constellation in more detail and look at more observables. Below in section 6.3.3.1, we will do so while focussing on what triggers the transition from a cooperative population state to an ever more defective one. Despite the long-term metastability of phase ②, several systems (21/128) eventually collapse to phase ③ exhibiting a defective attractor. We could suspect that for longer simulation times, most, if not all simulations, showing ever more defection to collapse into a phase of moderate defection. As we learned so far, in *ReCooDy*, transitions seem to depend on specific system constellations exceedings thresholds, which irreversibly change the dynamics. Therefore, we could expect an eventual collapse of an increasingly defective population, even if it takes a lot of time because it already happened in several realizations.

Phase ③: defective attractor A moderately defective population characterizes the last phase within the observed time frame, reached by 21 of the 128 model realizations. On average, populations exhibit a mean expected investment of $\bar{\mu}_i \approx -5.5$, which is moderately defective compared to phase ②. The value remains approximately constant with fluctuations of roughly ± 1 but no noticeable overall direction. Even more, we do not

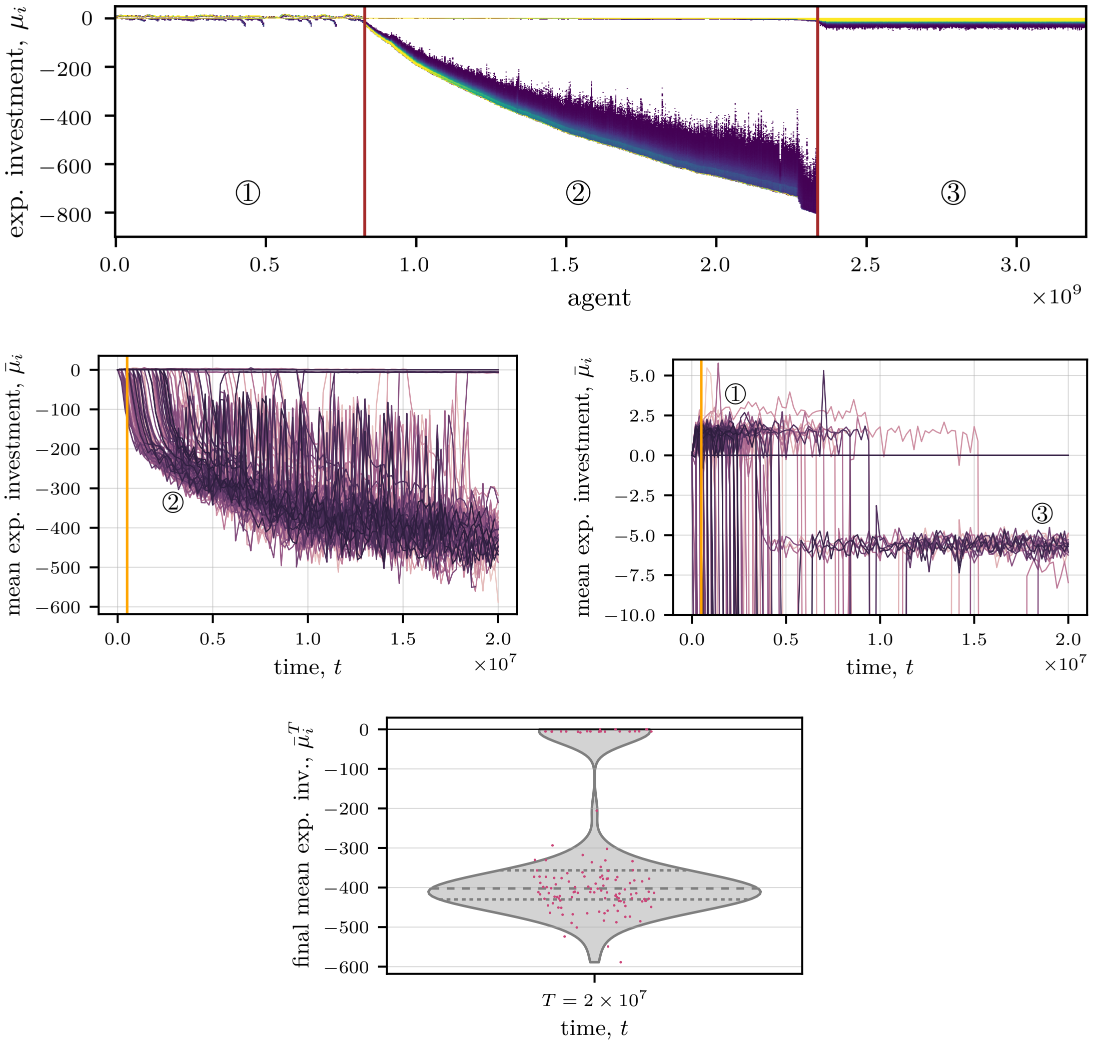


Figure 6.7: Long-term evolution for a synergy factor $r = 1.4$. The top shows the trait evolution plot of the expected investment μ_i for a single example system realization run for $T = 10^7$ time steps. It has 1000(500) bins in the x(y) direction with counts higher than 10^3 shown in yellow. The middle and bottom rows show simulation results of 128 system realizations through varying seed. It shows the time evolution of the mean expected investment $\bar{\mu}_i$ in the entire value range (middle left) and zoomed (middle right). Each line represents one universe run. The bottom shows the final mean expected investment $\bar{\mu}_i^T$ distribution. The orange line marks $t = 5 \times 10^5$, the simulation time used previously (Figure 6.5). The red lines mark the transitions in the example runs. There are three regimes: ① Cooperative populations that ② eventually become hugely defective before ③ collapsing into a regime of moderate defection. All initially cooperative populations die out on the visible time-scale, but only part of the extremely defective populations collapse into moderate defective populations (bottom). These transitions are highly improbable events with a huge impact. No moderately defective population evolves cooperation anew. See the caption in Figure 6.1 for the explanation of a trait evolution plot.

observe any following phase, i.e., on the simulated time frame, the moderately defective system attractor is stable. If we zoom even further into the middle right subfigure showing ③ we notice that this phase is predominantly reached as a result of the collapse of phase ② and rarely preceded by phase ①. In section 6.3.3.2, we will take a single simulation run to explore this last transition, what causes it, and how we can explain and interpret the observations.

We could suspect observing even more phases for longer simulation times, but at this point, we are limited by computational resources. Further, we would probably need to reach at least one order of magnitude higher simulation times, increase the number of simulations, or both. I did not observe a single simulation that showed another sequentially reached system attractor, and the agents' traits do not change significantly within the last phase (see also section 6.3.3.2). Therefore, we may classify the moderately defective system attractor as the final reached phase.

We notice that cooperative populations survive only on medium time scales but never for the observed 128 system realizations in the long run. Even more, cooperation never reemerges from the final moderately defective system state in contrast to the previous dynamical phase for which cooperation frequently reemerged after its collapse (see section 6.3.2).

6.3.3.1 Trigger Towards Growing Defection

What triggers the first transition towards a phase of ever more defective populations – the transition between phase ① and ② in Figure 6.7? This transition typically shows similar characteristics when looking at different system realizations (see Figure A.7). Therefore, I focus on a single representative simulation run in the following to understand what triggers the transition towards defection in detail. Figure 6.8 shows, from top left to bottom right, the trait evolution plots of the expected investment μ_i, the final age A, the lifetime accumulated costs for link removal, the lifetime accumulated cost for link addition, the probability to add local links, the global link addition mode, the local link addition mode, and the link removal mode at the transition point. All subfigures belong to the same simulation with seed 61. As before, color encodes local state density from low (purple) to high (yellow); The color scale is logarithmic for the accumulated quantities, and the age with values higher than 10^3 shown in yellow. Color scales linearly for the rest with bin counts larger than 30 (μ_i), 2×10^3 (for both link addition modes), or 10^3 (for the link removal mode) shown in yellow.

We first note that the collapse of cooperation and the transition towards growing defection actually happens in two sharp transitions, which are shown as vertical dark red lines. Let us first focus on the first transition, the sudden collapse of the cooperator branch with a surviving defector branch, as we can see in the upper left expected investment evolution plot. With the collapse, the defective branch becomes yellowish, meaning that the local defector density increases and, therefore, the number of defectors increases. In more detail, both the number of agents with defective investments i increases and the number of agents with high probabilities to synergistically interact p_s. This sudden collapse of cooperation is followed by a short phase of rising expected investments, which quickly turns into a first rather unsteady state of shrinking μ_i.

We can see that the agents' age distribution changes drastically with a collapsing cooperative branch (top right). At cooperative times, the vast majority of agents die after

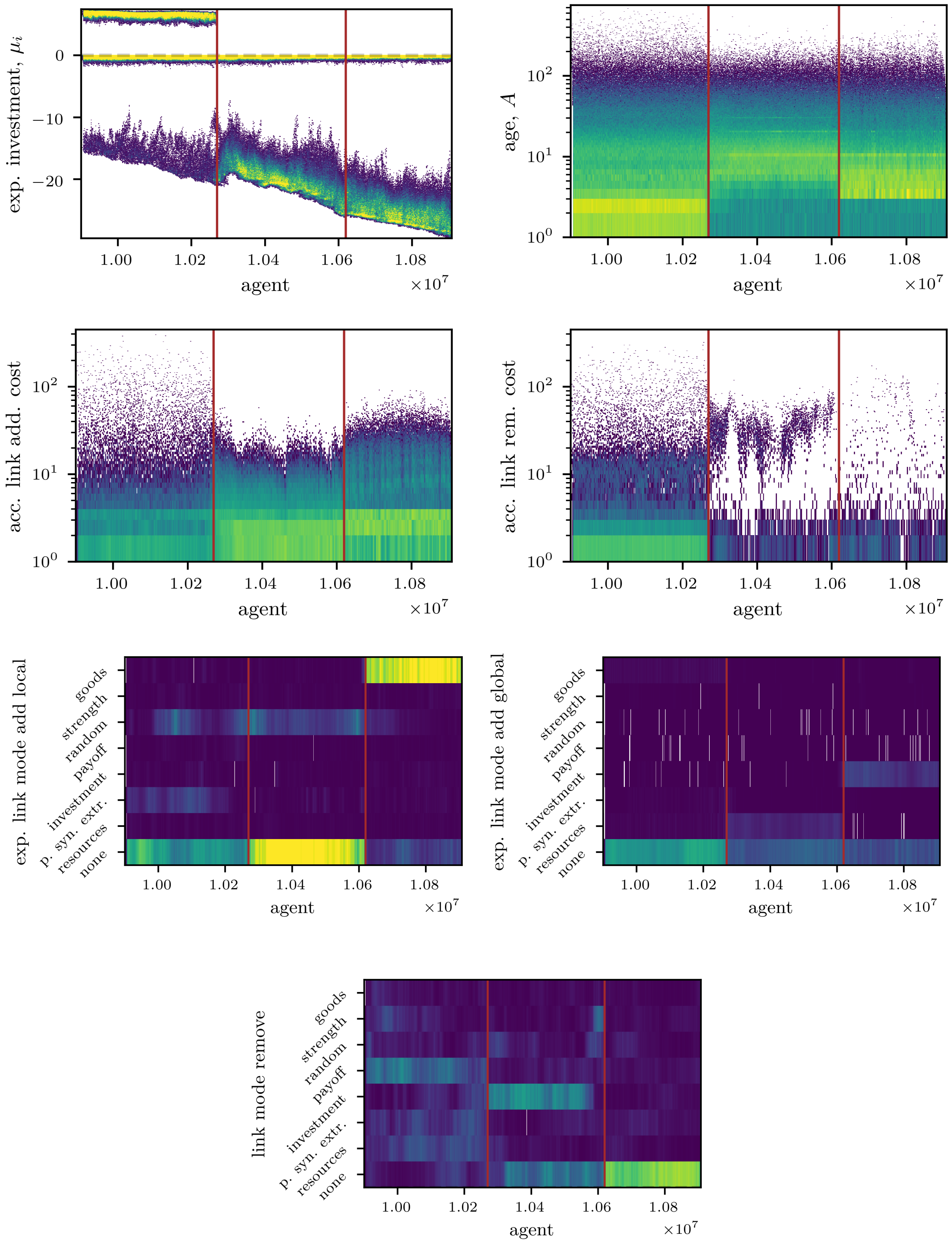


Figure 6.8: Trait evolution plots of the most important quantities that show the transition into growing defection. It happens in two main steps (red lines) that both significantly alter agent traits and the population structure. The simulation ran for $T = 5 \times 10^5$ steps. Figure A.8 complements this figure by showing the mean link addition and removal costs and the agents' accumulated number of links. See the caption in Figure 6.1 for the explanation of a trait evolution plot.

one or two time steps already—recalling the logarithmically color-coded local density. Thus, most agents born into the cooperative environment do not survive even for a few time steps; The phase is highly competitive. However, a few highly successful agents outlive most by up to two orders of magnitude. The situation changes at defective times. Here, most agents live approximately one order of magnitude longer (order of 10 time steps). Most survive the first few time steps with their initial resources. A few agents still outlive the absolute majority of others, but they do so for less time compared to the previous cooperative phase. If we recall the interaction properties (see section 4.2.2), we recognize the manifestation of the high-risk, high-reward nature of cooperation. Cooperation is a successful strategy only if others cooperate and the population structure promotes cooperation. If agents are born into a competitive social environment, having to invest resources in order to have the possibility to receive a payoff can result in immediate exhaustion and death, especially due to limited resources that are shared among successful agents only. Agents that live longer potentially have developed better tailored social environments, have already accumulated resources, and can survive better.

In contrast, defection can be beneficial for an individual as long as enough resources are available in the environment that can be extracted by multiple agents and if defectors have more cooperative neighbors that pay for the destructed goods. Especially at an early age, defection is the less risky strategy. Defective offspring do not experience a direct resource dissipation caused by positive investments. Rather, grabbing resources via negative investments comes with the potential immediate payoff that usually outweighs the destructed resource cost because the latter is shared at least with the parent (see also section 5.5). Here, we see that bads are shared, and profits are taken.

The individual resource dissipation via destructed resources caused by an agent's destruction gets even smaller for the agent if it has more links. Although, this comes with a potential pitfall because if the agent links to comparably more defective agents, it will need to pay the resource cost caused by their destruction. Also, the agent needs to pay a one-time cost for each link. Thus, for the transition towards defection to be stable, the benefit of grabbing resources needs to outweigh the costs of link formation. Translated to the observed quantities, this means that the expected investment needs to reach a negative threshold to allow for the defective branch to take off, which explains why, for different seeds, the observed transition always happens roughly in the same range of $-15 < \mu_i - 20$ (see the upper left figure in Figure 6.8 and more examples in Figure A.7).

If we shift our focus to the linking process, we observe that after the collapse of cooperation, agents add more links and stop removing links. The second row of Figure 6.8 shows the agents' accumulated costs for adding (left) and removing (right) links. Figure A.8 additionally shows the corresponding lifetime-averaged costs. Let us first focus on the first transition marked by the first vertical red line. For link addition, we observe that most agents spend only a few resources to add links before the transition, whereas a few agents spend a vast amount. When we look at how agents determine their linking partners, i.e. their link modes for global and local linking with weighted counts to account for the probability to either create local links p_l or global links p_g per time step (see section 4.4), we see that they target local links to agents with high probabilities of extracting synergistic resources p_s. Most agents with high p_s are cooperators but some are also defectors, the ones on the defective branch. Directly before the collapse of cooperation, we see a change in their local link addition preferences (3rd row, left): Agents start to choose their targets at random instead of based on high p_s. This trait change decreases the probability to target more cooperative agents because the chance increases to link to non-interacting agents, reducing the number of effective interaction partners. We can expect two consequences: First, cooperators are less likely to get exploited by defectors and second, cooperators support each other less. Both effects are enhanced because with an ever more defective branch, the cooperative branch also decreases and gets less populated due to a changed distribution of the limited resources (recall results from section 6.3.1). With the observations so far, we can extract a potential explanation for the collapse of cooperation: The cooperators' increased risk to get exploited triggers the decline of the p_s link addition mode because it evades the increasingly defective agents. However, the support between cooperator also diminishes as a result resulting in the eventual cascaded death of cooperators if a threshold is exceeded.

Also, after the first transition, agents that add local links, i.e., do not have "none" link addition mode choose their linking targets predominantly at random. Others start adding global links to targets with many resources (3rd row, right). As a counterreaction, a few agents cut a lot of links within their lifetime (2nd row, right), thereby focusing on cutting links to agents with low investment (4th row).

As the agents die out that remove a lot of links from low-investment others, the second transition happens (right red line), triggering a more steady evolution towards more and more defection. The expected investment evolves more steadily towards negative values (top left), the age distribution shifts to many more agents dying at just a few time steps approximately after their initial resources exhaust (top right), and the linking structure changes radically (bottom three rows). Apparently, a more hostile environment emerges for agents to live in, in which the initial resources can safely sustain agents for less time (3 steps instead of ≈ 5 as in the intermediate regime), but then instantaneously results in the death of the vast majority of agents (see 1st row, right and recall the logarithmic color-scaling). If we focus on the link addition and removal we see that nearly all agents stop removing links (see 4th row and 2nd row, right) with just a few agents left removing many links per time step (better seen in the normalized mean link removal costs in Figure A.8). Agents predominantly add links locally to agents on vertices with the most goods or rather the lowest bads. As expected, they try to avoid vertices, in which they would need to pay higher shares of destructed goods and try to link to vertices with fewer destructed goods. Another subset of agents connects globally to agents with high investments. In a defective regime, these are agents that grab fewer resources, thus, destruct fewer goods. For a defective agent that adds a link, the agent with the highest investment will destroy fewer goods in the interactions. However, the proportion of agents linking to the globally highest investment agent is comparatively small. If it gets too large, the target agent will rapidly die of exhaustion because it needs to pay for all the destructed goods from the incoming agents and additionally for removing links. If an agent creates a link to another agent that quickly dies of exhaustion, the cost paid for link addition will not yield a profit. Therefore, the source agents will no longer have a positive return from this linking strategy, which makes the strategy unattractive for many agents. This self-regulatory mechanism could explain the moderate density of agents globally linking to agents with the highest investments.

Indeed, in summary, the average number of links increases, and the most connected agents get even more links (see the number of total links evolution in Figure A.8). Defectors only profit from defection if others pay for their destruction. Therefore, defectors have an incentive to create links. As we already observed, they surpassed the threshold for them to survive from synergistic resources alone. Within a connected network of defectors, more defective agents will be more successful in the evolutionary system as they can grab more resources and partition more of their destructed goods than less defective agents.

Earlier in Figure 6.7, we have seen that the regime of rising defection remains metastable over long times reaching highly negative expected investments of $\mu_i \approx -800$ in this specific simulation. How does such a situation look microscopically? Let us, for simplicity, focus on the extreme monopoly-like case, in which a single agent grabs all available resources within a time step. All connected agents share the costs of the synergistically destroyed resources and cannot take any resources themselves because nothing is left, whereas the most defective agents gain an enormous amount of resources. Let us say that this agent gains 800 resource units in one time step. In the given parameter setting and with these resources, it can theoretically survive for $8,000$ time steps, create or delete 800 links, or have $800/7 \approx 114$ offsprings. Thus, a single win in the interactions could be enough to sustain the agent a whole lifetime, creating many defective offsprings. Even more, being successful requires changes in the population structure to become unsuccessful the next time. Further, the more offspring an agent generates and the more connections the agent generates, the lower its individual cost for the destructed resources.

Still, the phase of rising defection eventually collapses into the final state of infrequent defection. In the following, we will explore its transition.

6.3.3.2 Eventual Collapse into Infrequent Defection

Let us explore what happens at the transition at which the ever more defective population collapses into a phase of moderate defection. Figure 6.9 shows the simulation results for one system realization that eventually collapses and zoom into the relevant part. It shows the trait evolution plot of the expected investment μ_i (top left), the probability of extracting synergistic resources p_s (top right), the investment i (middle left), the expected strength μ_s (middle right), and the mean link addition and removal costs averaged over an agent's lifetime (bottom left and right). The corresponding complete effective investment evolution is visualized in Figure 6.7 (top). We observe that the transition happens in two separate steps, as indicated by the dark red vertical lines that determine three distinct dynamical phases.

First transition Before the first transition, we observe a steady decrease of the expected investment μ_i within the defective branch. Inspection of the whole preceding range reveals that not only the values of the investment but also the density of p_s decrease continuously. Thus, fewer agents interact but are more and more defective, meaning that fewer agents share the extracted synergistic resources, which makes the system more unstable. Moreover, we see that agents invest their resources in adding and removing links to optimize their social environment to partition their created bads better. In section 6.3.3.1, we already inspected this regime at an early phase.

Directly after the first transition (left vertical red line), we see that the local defector density decreases significantly, and the local density of agents with $p_s \approx 0$ increases substantially. The surviving defectors stop removing links but still invest their resources to add links, even though a lot fewer agents do so. The defectors evolve expected strengths $\mu_s > 1$ reaching up to $\mu_s \approx 9$. It is astonishing because the maximal return agents receive from basic resource extraction is $j_b = 1$, leaving them with a net loss of resources each

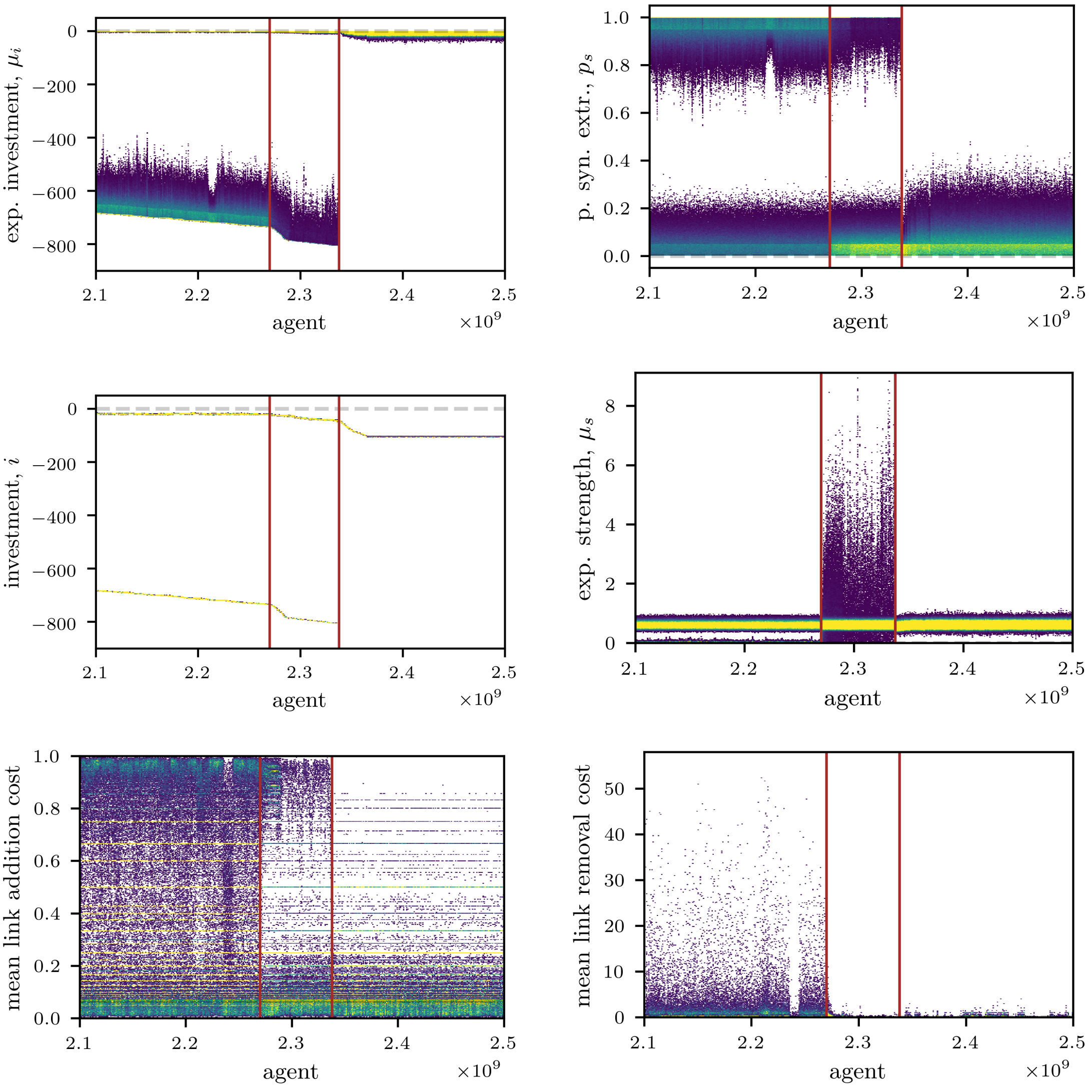


Figure 6.9: Trait evolution plots of the most important quantities showing the final collapse of growing defection into moderate defection. It has 500 bins in each dimension, except for the bottom two rows with 100 bins in y-direction. The transition happens in two stages (red lines) that significantly alter the agents' traits. The entire simulation ran for $T = 10^7$ steps. It is the same as in Figure 6.7 (top). See the caption in Figure 6.1 for the explanation of a trait evolution plot.

time they extract basic resources. Even more, if they do not extract basic resources, they still have to pay a cost c_s for their strength. Even though they win each time against their direct competition for these resources, they have guaranteed losses. This setting is metastable for approximately 5×10^7 agents or roughly 5×10^4 generations. Nevertheless, it eventually collapses in an instant event.

A possible explanation of these observations is as follows: At the point where the transition happens, around $\mu_i \approx 700$, only one agent gets to extract all the synergistic resources within one time step because the sum of created bads and actively grabbed resources exceeds the available resource amount $\mathcal{A}_s$. The lower μ_i evolves, the more resources a single agent receives, further increasing the resource gap between the one successful agent within a time step and its neighbors that have to pay for the additionally destroyed resources. The successful agent will eventually, with a probability of 0.5, create a more defective offspring. The parent potentially will have less success than some offspring, depending on the exact local network configuration, leaving it empty-handed. With the accumulated resources from the previous success, the parent can survive and pay for the bads created by its offspring for some time. Still, the probability for a crucial part of the network to collapse increases with the increasing success gap between successful and unsuccessful agents, too. The death of a few crucial agents at key positions in the network could result in a cascade of deaths because fewer agents have to pay for the destructed goods.

The few defective agents evolve a completely unreasonable strategy of exceedingly high μ_s that lets them outcompete most others on rare occasions, in which they extract basic resources but leaves them with a guaranteed net loss of resources. Their high success in synergistic extraction lets them spend unreasonable amounts on optimizing basic resource extraction, too. They receive so many resources in the synergistic extraction that they can evolve unnecessary and unreasonable resource spendings without crucial implications. Nevertheless, the increased spendings without return eventually lead to an instantaneous collapse of the defective branch.

Collapse into the final defective attractor The right vertical red line in Figure 6.9 indicates the last transition; the collapse of the frequently interacting defectors branch as previously observed for several simulation runs in Figure 6.7. The mean expected investment stays constant at around $\bar{\mu}_i \approx -5.5$, the probability of extracting synergistic resources is predominantly around $p_s \approx 0$ and the investment trait i stays constant over time around $i \approx -104$. After the transition, the expected strength μ_s stays slightly below 1 again, which does not anymore create an inevitable net resource loss as before the transition. Agents have low mean link addition and removal costs; they do not invest in changing the network structure.

Overall, we see that the coexistence of strategies breaks together. Agents are not specialized either in basic or synergistic resources anymore. Instead, all of them predominantly extract basic resources and only infrequently extract synergistic ones. The investment does not change anymore. Therefore, there is a mechanism counteracting the pressure of evolving ever more defection. For such highly negative investments, the individual agents reach the limit of available resources per time, here $\mathcal{A}_s = 10^3$. Only a few agents indeed receive resources if a single agent tries to extract in the order of a tenth the total amount. That agent will also produce collateral damage through negative synergies for which others need to pay. If no other agent pays for the destructed resources, the extracting agents will indeed have a net loss of resources. Defectors need others to pay for the destructed goods, but only a few defectors extract vast amounts of resources. Therefore, the final attractor contains the minimal set of defectors that can successfully extract resources in their infrequent attempts. Below, in dynamical phase ③ of section 6.4.2, we will look at the final defective attractor in higher synergy regimes in more detail by using the evolved traits to estimate the number of successful agents and roughly quantify this regime.

Interestingly, we already encountered a setting in which agents evolve not to interact in order to avoid vital dependency on the interaction outcome for minute synergies ($r = 1.01$), which we explored in the first results (section 6.2). In both regimes, they cannot survive from the extracted synergistic resources alone. For low synergy factors, the amount of created goods is not enough, while here, the population evolved into a hostile defective state, from which it cannot evolve away without external changes, effectively not allowing for a vital dependency on synergistic resources.

6.3.4 Summary and Discussion

The transition regime, especially in the regime $1.2 \leq r \leq 1.5$, which we focussed on, exhibits rich phenomenology with diverse dynamics, strategy coexistence, various strategy bifurcations, and a final defective attractor in the long-term evolution. Only changing the sequence of random numbers can significantly alter the system's trajectory. However, the population self-organizes into structurally similar dynamical phases that emerge and are metastable on time scales typically ranging from some 10 up to some 10^4 generations in the initial dynamical phase and to 10^6 for the long-term increasingly defective metastable state. Typically, agents specialize either in basic or synergistic resource extraction resulting in the emergence and metastable coexistence of at least two distinct strategies. Usually, the majority of agents specialize in basic resources. However, cooperation emerges within a subset of the population. Initially, cooperators infrequently interact. Thus, they are not yet crucially dependent on the outcomes of the interactions. Evolution selects for more cooperative agents, and as soon as the cooperative agents receive enough resources as a result of their interactions, they start to rely on and specialize in the synergistic resources entirely. Frequent cooperation emerges as a strategy, and agents self-organize either into cooperators or loners. Here, accessing a formerly inaccessible resource drives the evolution of cooperation and elevates the system's carrying capacity such that more agents live communally.

Once cooperation evolves, defection becomes profitable and emerges naturally as a response because there are cooperators to exploit that evolved to vitally depend on the interactions. Frequently, defector emerge but cannot survive because they harm each other, and cooperator can evolve defense mechanisms. Still, at times defection emerges with ever more defective agents evolving over generations. They need to evolve ever more defective strategies to remain competitive against other defectors in a Red Queen dynamics[2]. However, defection is a self-harming strategy because it creates defective,

[2] van *Valen* (1973) introduced the Red Queen hypothesis in the context of coevolving species that mutually deteriorate the others' fitness. He was inspired by what the Red Queen said to Alice in wonderland in the inspiring book *Through the Looking-Glass, and What Alice Found There* by *Lewis Carroll* (1871): "Now, here, you see, it takes all the running you can do, to keep in the same place." In this thesis, I use the qualitative concept behind the Red Queen's statement: A replicator's necessity to constantly adapt

thus, harmful offspring from the parent's perspective (see section 5.5). Therefore, the Red Queen dynamics eventually leads to a collapse of either all defectors, all defectors and all cooperators, or all cooperators. The first is most frequent and does not have a great impact on the entire population dynamics. The second occasionally happens and leads to a new starting dynamical phase, usually of emerging cooperation, thus driving the observed recurring dynamics cycles. The last happens rarely but has a tremendous impact as it shapes a different future. Defectors can only survive on their own without cooperators if they evolve low-enough investments, i.e., if they grab enough resources to sustain themselves and outweigh the resource losses caused by destructed goods. Once there are only defectors without cooperators, they need to evolve ever more defection to outcompete the other defectors. The Red Queen drives the dynamics. Cooperation cannot reemerge in such a hostile population because cooperators would easily get exploited.

A collapse typically happens shortly after agents change their linking behavior. Two examples, that we have observed, are agents that evolved to preferably add local links to interacting agents or to remove links to low-investment agents. Both changes in linking strategy introduce a preference in their linking action that targets a small subset of agent strategies disproportionally. Both aim at selfishly optimizing an agent's payoffs from the interactions. However, such selfish optimizations target more cooperative agents that are forced to spend a lot of resources on removing links and also on the more defective neighborhood. Eventually, these selfish optimizations result in cooperators being overwhelmed and dying out. cooperators support each other. Once a critical number of connected cooperators dies out, a cascade of deaths can be triggered as their self-support collapses.

From a different perspective, within the dynamical phase in which the system flourished with stable cooperation and four times the number of agents compared to previous times, agents preferably add links to random global neighbors because random linking is the only introduced linking mechanism that does not inhere a linking preference. Thus, it creates less network heterogeneity. It contradicts the simulation results of *Santos* et al. (2008) who found that heterogeneity in static networks promotes the evolution of cooperation, however, is in agreement with behavioral psychology experiments, which mostly indicate that heterogeneous networks do not promote cooperation in human interactions (see section 2.1.4). Moreover, the observation that random linking increases cooperation contrasts the findings of the model from *Akçay* (2018), who found that random relocations of links hinder the emergence of cooperation as random global linking increases the well-mixedness of the system compared to local linking. The results obtained here indicate that if the alternative to random linking is preferential linking with the risk of exhausting target agents, random global linking can benefit the evolution of cooperation.

In the very long-term evolution, defectors experience a final transition into infrequent defection, which, however, happened only for several systems (21/128) on the observed time frame. In these situations, agents evolve minimal interaction probabilities, thus, mainly rely on basic resource extraction instead of synergistic ones. The risk of relying on synergistic resources alone significantly grows as the available resource amount has to be shared among only a few agents each time. All other interacting defectors have to pay for the collateral damage without receiving a profit. defectors need others to pay for their destruction. Eventually, defectors evolved infrequent extraction of synergistic resources.

in an ever-changing biotic and abiotic environment in order to "keep in the same place".

This state of infrequent defection appears to be an attractive system state from which agents apparently cannot evolve away.

The actual trigger of this last transition showed agents evolving a “stupid” strategy that yields guaranteed losses during basic resource extraction. As we looked at one specific simulation run, we observed highly defective agents that rarely extract basic resources evolving extremely high expected strengths up to 8 times their maximal resource gain. Thus, each time they extract basic resources, they outcompete others but will never profit from them. Instead, they waste resources. They can evolve such high strength because these defectors gain vast amounts of resources from the synergistic resource extraction that allows them to spend otherwise not used resources also on invading agents competing for basic resources. Nevertheless, the additional resource losses harm them in the long run and eventually trigger the collapse of the frequently interacting defective agents.

More generally, in the transition regime, we typically observe instantaneous collapse events, which raises the question of their origin. By construction, *ReCooDy* contains several thresholds such as for birth, death, or all linking (see chapter 4). Also, the linking mechanism is modeled in a simple form as agents always choose the best-suited target with respect to a given trait or state. Errors do not occur. Thus, single agents can become targets of many, which can result in quickly exceeded thresholds. Furthermore, the outcome of the interactions crucially depends on the actions of others. Thus, combined with the mentioned thresholds, we can expect cascades of agents exceeding thresholds that can result in the death of many. *ReCooDy* is a threshold system in which highly improbable events can trigger radical reorganization, as observed in the simulations.

The simulation results also show the general relevance of history. Formerly attractive system states can be left irrevocably, as the long-term evolution shows. Even more, we observe the system reaching a final attractive state, from which it cannot escape.

We cannot exactly predict when a transition between dynamical phases happens nor which dynamical phase will be reached after the transition. Still, we observe a set of possible dynamical structures and typical time horizons for the individual phases. These are typical qualitative properties of chaotic and complex systems[3].

Furthermore, the simulation results show that most offspring die immediately during cooperative dynamical phases, while during more defective dynamical phases, most offspring survive for longer times. What we already anticipated in the analytical considerations in section 5.5 turns out to be observable in the simulations: Cooperative parents profit because their offspring effectively pays for the investment creating the parents’ benefit. In contrast, when defectors create offspring, the parent effectively pays for most of the offspring’s destruction while the offspring profits. By construction of the gPGG on networks, cooperators act selfishly towards their offspring, while defectors act selflessly towards their offspring, which reveals yet another facet of the social dilemma.

[3] See (*Strogatz* 2014) for an excellent introduction to chaos theory.

6.4 High Synergy Regime

In this section, we will extend the previously investigated range of synergy factors to higher values. Thus, we concentrate on interactions that generate a lot of synergies while keeping all other model parameters constant as in the previous sections. Figure 6.10 shows simulation results of 512 seeds for 17 different synergy factors in the range $2 < r < 10$, which we will describe, interpret, and explain in the following. It shows the mean expected investment $\bar{\mu}_i$ time evolution and its final distribution after $T = 10^5$ time steps for different synergy factors r. It naturally extends the synergy factor range investigated previously in Figure 6.2 and Figure 6.4.

Cooperation emerges, defection as well. We observe that for all synergy factors, positive mean expected investments $\bar{\mu}_i$ evolve immediately. However, the general time evolution is qualitatively different for $r = 2$ than for $r > 2$. It belongs to the dynamical regime previously observed and described in Figure 6.4 and is included here for comparison. For $r = 2$, we observe $\bar{\mu}_i \approx 1.8$ after the initial phase, which does not change significantly with increasing time. For $r > 2$, simulations rapidly reach positive $\bar{\mu}_i$ and from there slowly but steadily increase further during the first 10^5 time steps. However, the variance of $\bar{\mu}_i$ increases, and eventually, $\bar{\mu}_i$ starts decreasing again on average, reaching values approximately half as high as for $r = 2$. For $r = 10$, we see only a slowly increasing tendency without the decrease. If we focus on the final distributions at time $T = 5 \times 10^5$ (right) we observe that for each synergy factor, the majority of simulations reach positive values within $0 < \bar{\mu}_i^T < 2$. However, for all synergy factors $r > 2$, we also see several simulations with negative $\bar{\mu}_i^T$ separated through a gap from the on average positive values. Further, there are a few outlier simulations for some synergy factors with extremely high $\bar{\mu}_i^T$ as indicated by the violinplots and clearly shown in Figure A.9. All of these observations indicate that, as in the previous transition regime in section 6.3, the system contains different dynamical attractors such that a changed sequence of random numbers can lead to significantly different system realizations. On the observed time frame, most model realizations show, on average, cooperative populations, a few exhibit extreme magnitude of cooperation, and some show defective populations. We cannot explain the observed dynamical regimes only from these averaged observable. Instead, we need to explore the microscopic system realization again, which we will do in section 6.4.1 for the representative $r = 3$ case.

More synergy comes with less mean cooperation. With increasing synergy factors r, we observe the tendency of lower mean expected investment (see right plot in Figure 6.10). We see this decrease in the right plot if we follow the bulk of the violin plots indicating the median of the $\bar{\mu}_i^T$ distribution. Of course, the tendency is slight and the measure rough because, arguably, it does not adequately cope with the underlying intricacy of the system. Still, it hints at a tendency that we actually could explain using a macroscopic explanation. The higher r is, the higher the shares of goods each agent would theoretically receive from the same investment. However, a distribution issue arises with the fixed amount of resources flowing into the system each time step. Either fewer agents receive their expected resource share, or the same number of agents evolves lower expected investments to receive effectively the same resource share as in the case of lower synergies. Both would decrease the population's mean cooperation, as observed in the figure. Therefore,

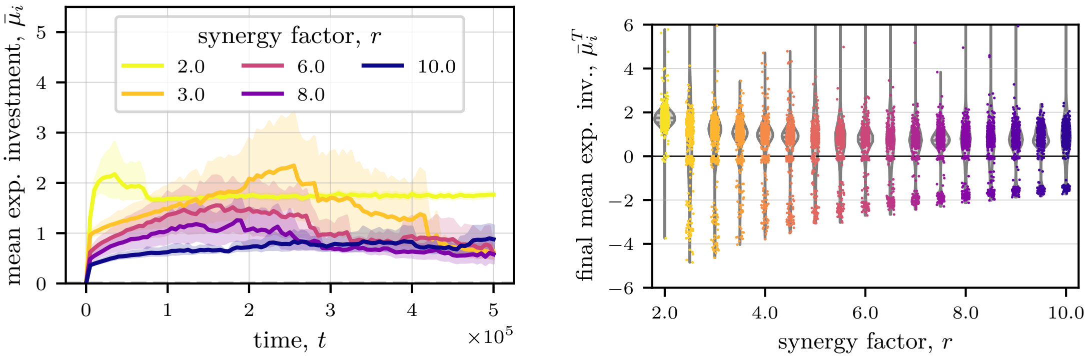


Figure 6.10: Time evolution plot of the mean expected investment $\bar{\mu}_i$ (left) and its final distribution $\bar{\mu}_i^T$ at time $T = 10^5$ (right) for different synergy factors r showing the high synergy regime. On the left, lines represent the mean and their shaded areas the standard deviation of $\bar{\mu}_i$ calculated over simulation runs. On the right, each dot represents the final population average of a single system realization (512 per r). Color encodes rising synergy factors from low (yellow) to high values (blue). The violin plots in the background show the kernel density estimates of the distributions. The width of each violin is scaled according to equal areas. Lines within the violins show the median and quartiles of the corresponding distribution. The right plot shows a zoomed value range for $\bar{\mu}_i^T$. Figure A.9 shows the full value range with extreme outliers. Dead populations are shown as dots at $\bar{\mu}_i^T = 0$.

partitioning the limited resources among fewer and lower investment agents can explain the tendency of less cooperation for higher synergy factors.

More synergy comes with less severe defection. With increasing synergy factors r, we further observe that the final defective populations have comparatively higher values of $\bar{\mu}_i^T$. Thus, overall defective populations are less defective for higher synergy factors. It again is a manifestation of the distribution issue. Partitioning the limited resources among fewer agents with less negative investments can explain the tendency of less extreme defective populations for higher synergy factors.

6.4.1 Dynamical Phases and their Transitions

Figure 6.11 shows the expected investment μ_i evolution of selection of representative example simulation runs with $r = 3$ ordered according to their final mean expected investment $\bar{\mu}_i^T$ from low to high values after $T = 5 \times 10^5$ time steps. In the following, we explore the observed structural patterns.

Rising cooperative branch. First, we notice that all simulations start with a cooperative branch that emerges and thrives towards higher and higher expected investments μ_i with roughly linear growth. These branches of ever-higher μ_i eventually collapse. In the example runs, this collapse happens at some realization-specific exceeded threshold approximately within the range $80 < \mu_i < 160$. If we recall the analytical considerations in chapter 5, we recognize that for $r = 3$, the social dilemma is weak in many microscopic local configurations. Further, the personal benefit of creating cooperative offspring also increases with r (see section 5.5). Therefore, the emergence of cooperation and its evolution towards ever higher μ_i is not surprising because agents indeed benefit from their investments. With higher investment, the benefit increases, too, making more cooperative agents more successful than their less investing competitors. Again, we observe Red Queen dynamics towards increasing μ_i. The cooperative branch evolves towards ever higher μ_i while the number of cooperators constantly decreasing. It is in contrast to the previous transition regime (section 6.3), in which cooperation usually approaches a constant value of μ_i without growing further. For higher synergies, the cooperative branch collapses before reaching a perhaps existing but unreachable asymptotic state.

How could we explain the instantaneous collapse of a cooperative branch? Higher investments result in more goods created for local social environments. However, the amount of available synergistic resources is limited ($\mathcal{A}_s = 10^3$), i.e., not all agents indeed receive a payoff. Therefore, with higher expected investments, fewer agents receive more of the available goods. However, cooperators rely on the support of each other to create public goods. After all, goods are only created within a social environment if there are investing agents. Additionally, less cooperators with higher investments are easier to exploit by profiteers or exploiters. Thus, the cooperative population evolves towards an increasingly unstable state, eventually resulting in a collapse of the cooperative branch.

The observations exhibit yet another facet of the social dilemma because due to limited resources, cooperators can indeed act incredibly selfish, taking all the goods and leaving nothing for others. We will discuss the implications further in the overall discussion in chapter 7.

Dying populations. Seed 110 in Figure 6.11 exemplifies a simulation run for which the whole population dies out. As before, we observe an increase in μ_i only that here the growth rate is roughly an order of magnitude faster than before and that there is no subpopulation of agents left that specializes in basic resource extraction. All agents quickly evolve the strategy to specialize in synergistic resource extraction entirely with quickly growing investments. As explained in the paragraph above, such a rising cooperative branch eventually collapses. With no other agents left, the entire population dies out.

Highly cooperative populations. In Figure 6.11, seed 111 shows a system realization for which the whole population evolves high expected investments compared to other realizations. This simulation represent a positive outlier in the violin plot for $r = 3$ in Figure 6.10 (more clearly visible in the appendix in Figure A.9). Interestingly, the initial phase resembles the one for an eventually dying population (see paragraph directly above). The important difference is that, here, before collapsing, the entire population evolves minimal probabilities of extracting synergistic resources p_s. We can observe it indirectly through the suddenly emerging low values of μ_i around agent 10^5. Inspection of p_s reveals it more clearly (not shown). Thus, the population evolves to predominantly extract basic resources and, therefore, prevents its collapse as seen in seed 110. From there on, a new cooperative branch emerges. It evolves much faster compared to other simulations because agents already evolved high investments in their initial phase. The baseline of investments belonging to the loners does not change significantly over the following time (see also the evolution of the investment in Figure A.10). There is no strong selection for or against higher or lower investments for the loners. Thus, the agents' initial evolutionary phase, in which they collectively evolved high investments, significantly shapes the course of the whole simulation further on.

Defective populations. In Figure 6.11, the first row of simulations (seeds 71, 78, and 70) show defective final populations ordered from strongly defective to slightly defective. At the final phase of the simulations, there is no strategy separation. Instead, the entire population evolves towards a state of more and more defection. Agents evolve more and more negative investments while having low synergistic extraction probabilities $p_s \approx 0$, as revealed by looking at both quantities separately (see Figure A.10 for the former). Thus, an agent infrequently extracts synergistic resources, but if it does, it grabs many resources while creating a lot of collateral damage. From another perspective, we may say that if the agent is occasionally forced to interact with others, being more defective appears to be the more successful strategy in the evolved population setting.

6.4.2 Long-Term Evolution

Until now, we focussed on the medium-term evolution of *ReCooDy* in the high synergy regime. We observed recurring dynamical patterns dominated by either emerging and eventually collapsing cooperative branches or increasingly defective populations. Here, we will look at the long-term evolution and investigate how the system unfolds in the long run and which dynamics pattern predominantly occurs.

Figure 6.12 shows simulation results of 128 different model realizations over $T = 2 \times 10^7$ time steps. We observe that most model realizations evolve through three dynamic phases in the long run marked by numbers within the figures: They ① evolve cooperative

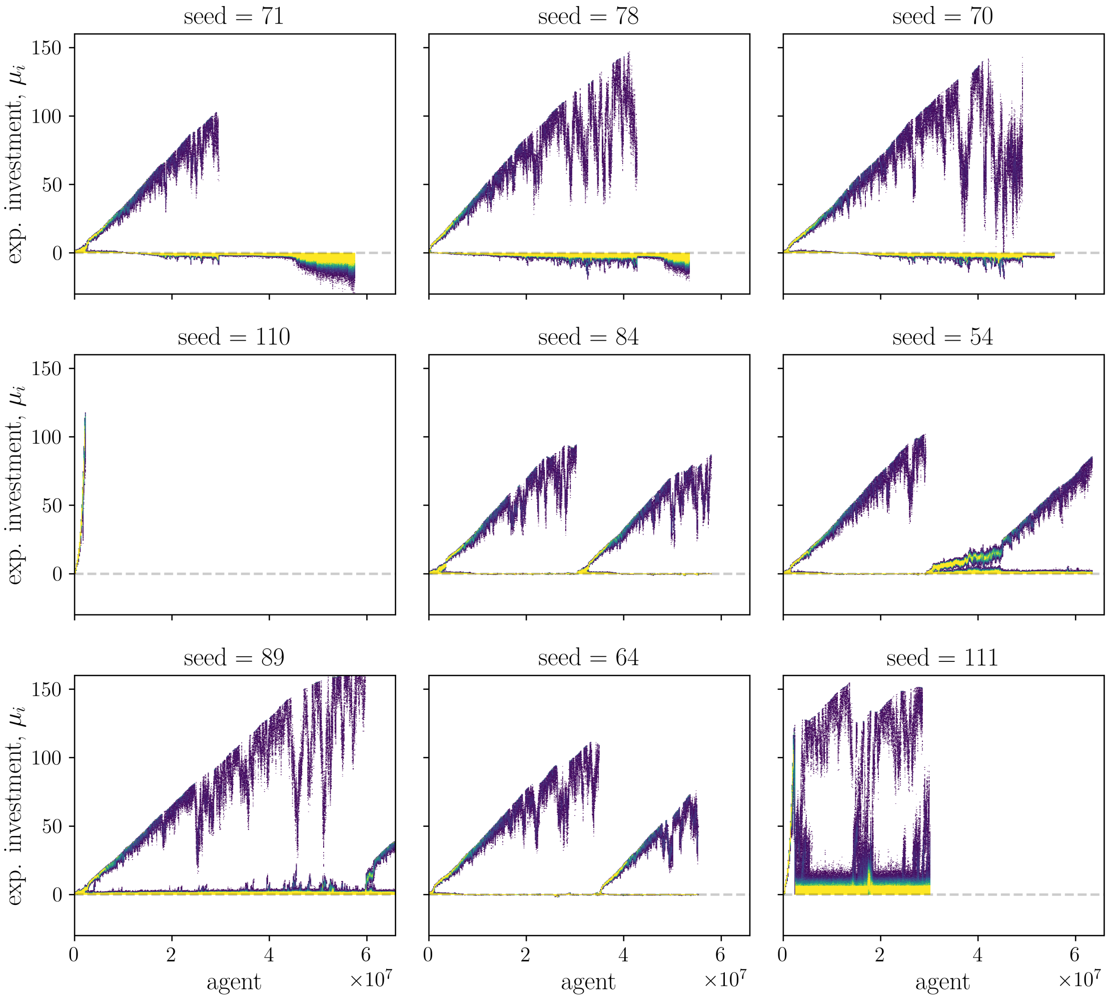


Figure 6.11: Trait evolution plots of the expected investment μ_i for $r = 3$ and varying random number generator seed that exemplify dynamical phases in the high synergy regime. Each plot has 900 bins in both dimensions with counts higher than 200 shown in yellow. The seed ordering follows lowest to highest final mean expected investment values from top to bottom and left to right. Each simulation ran for $T = 5 \times 10^5$ simulation steps. The populations with seeds 71, 78, and 70 show $\bar{\mu}_i^T < 0$, 110 shows $\bar{\mu}_i^T = 0$, and 84, 54, 89, 64, and 111 show $\bar{\mu}_i^T > 0$. Seed 111 is a positive outlier in the respective violin in Figure 6.10; Figure A.10 complements this figure by showing the trait evolution plot of the corresponding investment trait. See the caption in Figure 6.1 for the explanation of a trait evolution plot.

population in the beginning, ② collapse into a slightly defective population state from which they can either re-evolve cooperation or ③ transition into a final state of defection. Figure 6.13 shows the expected investment μ_i and the investment traits i evolution for two example runs. Let us look at each dynamical phase individually in the following.

Phase ①: cooperative populations. All 128 populations initially evolve positive expected investments $\bar{\mu}_i$. They become cooperative from the initial neutral population state. We already observed this phenomenology for single runs in Figure 6.11 but now for more system realizations. After the first collapse of the cooperative branch, a few model realizations reach slightly defective average states ($\bar{\mu}_i < 0$), which trigger a transition into the completely defective state (see phase ② below). However, we see that most model realizations live through several phases of rising and collapsing cooperation because they show $\bar{\mu}_i > 0$ over more than 10^6 time steps. Still, the absolute majority of cooperative populations transition into defective ones within the first 10^7 time steps, with most in the first half. However, after $t = 1.5 \times 10^7$, no system transitions into a defective state anymore. Within the dynamical phase of rising cooperation, we occasionally observe outstandingly high spikes of $\bar{\mu}_i$ (see ⓪ in Figure 6.12). These model realizations correspond to the previously observed realizations, in which the entire population evolves cooperation (see for example seed 111 in Figure 6.11). Now, we observe that these phases do not necessarily only occur at the beginning of a simulation but can evolve after a collapsed cooperation event, too. On the observed time scale, 8/128 model realizations remain on average cooperative until the end of the simulation. We see a slight tendency that these are model realizations with higher $\bar{\mu}_i$ than the cooperative populations that collapsed within the observed time frame. It indicates that populations that evolved high investments as a whole during their lifetime, i.e., that lived through spike events (see ⓪), are much less likely to reach a negative expected investment threshold that can lead to the irreversible evolution towards defection (see ② and ③ below). Instead, these populations indeed have a higher extinction probability, as we can, for example, see in two simulations around time 1.5×10^7 that reach $\bar{\mu}_i = 0$. Therefore, on even longer time scales, we could expect the on average highly cooperative populations to go extinct eventually and not transition into a defective state.

Phase ②: moderate defection transient This dynamical phase is occasionally reached after a collapse of the cooperative branch under the precondition that the surviving population is defective (see bottom left figure in Figure 6.12). Populations reach a metastable state with $\bar{\mu}_i \approx -0.2$, which is never reached through consecutive small mutation steps beforehand but only instantaneously after a collapse of cooperation. Once a population reaches this moderate defection transient, it stays and only slowly evolves away on moderate time scales within the order of a few 10^5 times (2×10^6 agents or approximately 3×10^3 generations). The system's trajectory qualitatively behaves as if on a critical point with slow repulsion. There is no strong selection on the investment trait. Still, phase ② is only a transient phase because trajectories eventually leave the critical point and either evolve cooperation again, which we can see in the trajectories that evolve towards positive values before quickly shooting up again, or the defective state of phase ③, which we can observe in the rapidly decreasing $\bar{\mu}_i$.

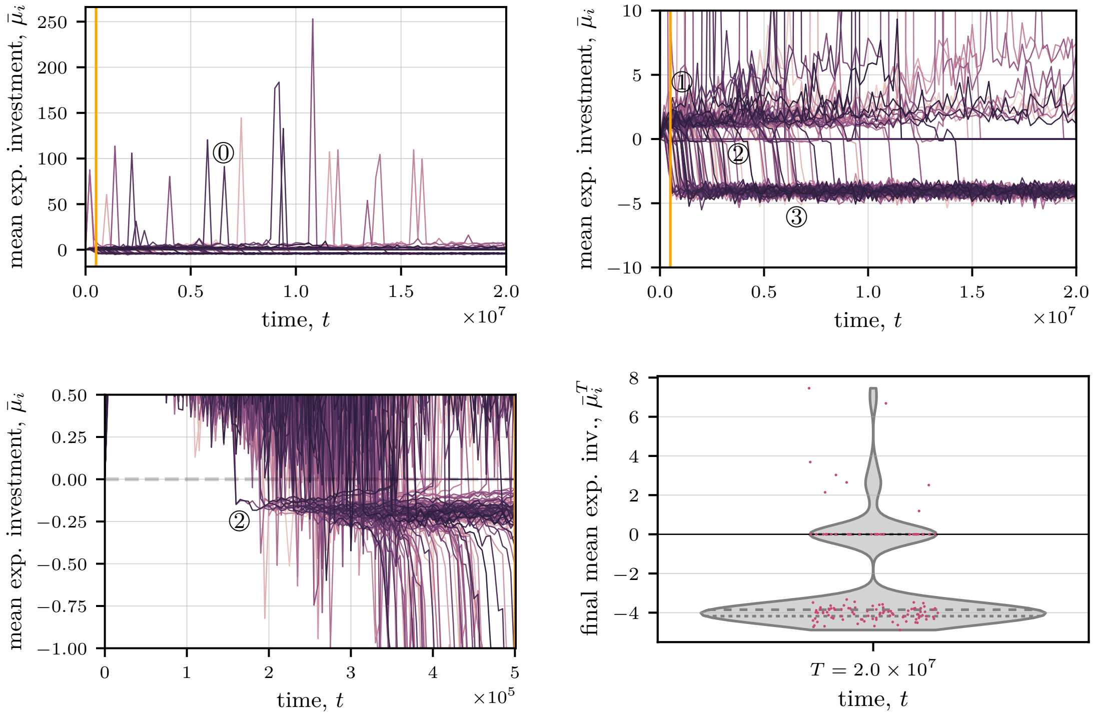


Figure 6.12: Long-term evolution of the high synergy regime. The time evolution of the mean expected investment $\bar{\mu}_i$ for synergy factor $r = 3$, simulation time $T = 2 \times 10^7$, and 512 different seeds in the entire value range (top left), zoomed into $\bar{\mu}_i$ (top right), and further zoomed into $\bar{\mu}_i$ and t (bottom left) together with its final distributions $\bar{\mu}_i^T$ (bottom right) are shown. Each line represents one system realization. The vertical orange line represents the zoomed time $t = 5 \times 10^5$. Single dots represent $\bar{\mu}_i^T$ of a single system realization with dead populations visualized as $\bar{\mu}_i^T = 0$. The violin in the background shows the kernel density estimate of the distribution. The plots show the long-term evolution of the high synergy regime moving from ① cooperative populations through ② a metastable population state of defection either back to a cooperative population or ③ to the final defective attractor state.

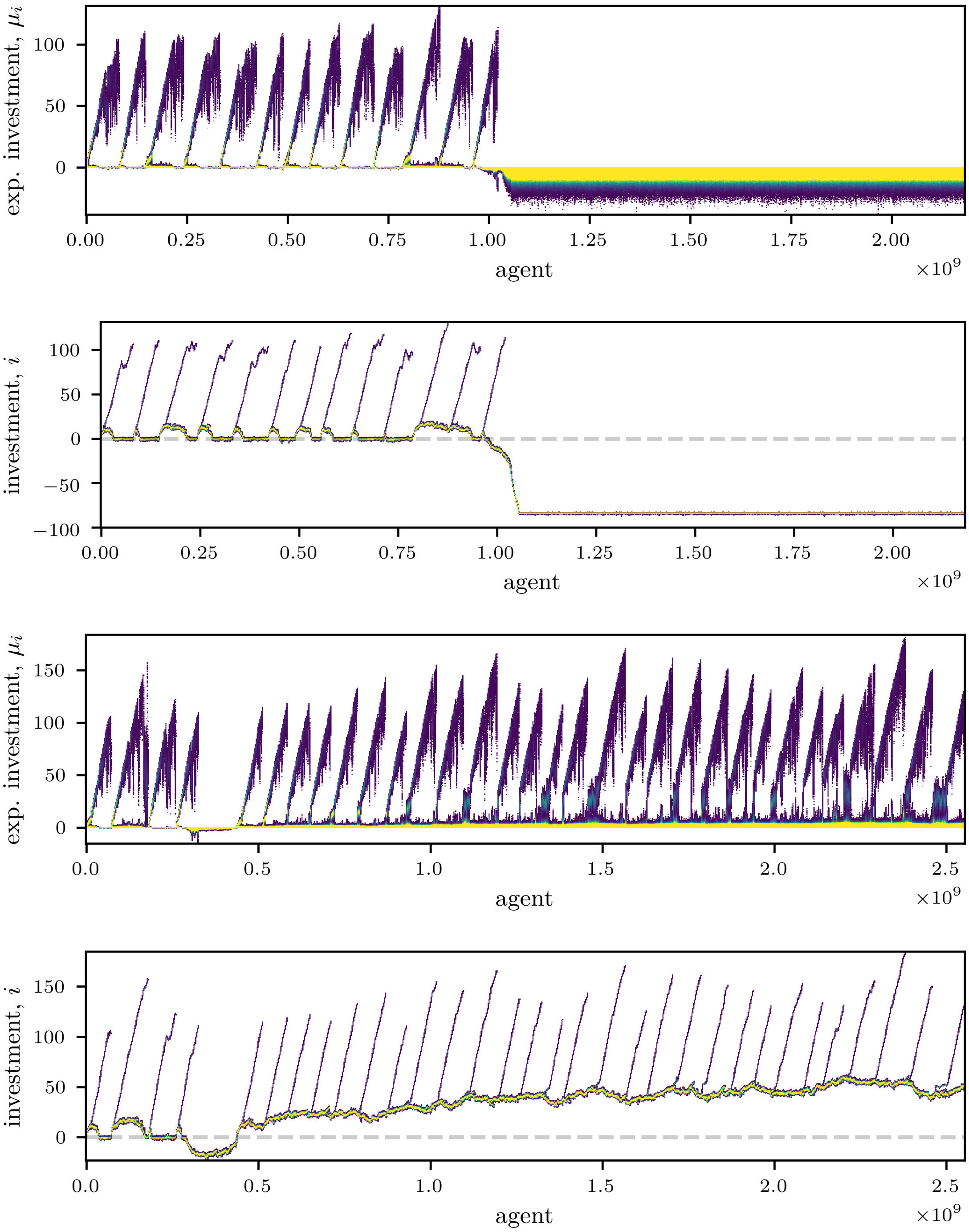


Figure 6.13: Example realizations showing the long-term evolution for $r = 3$ with a defective final attractor (top two rows) and continuously cooperative populations (bottom two rows) as shown in Figure 6.12. Trait evolution plots of the respective expected investments μ_i and the investment traits i are shown. Each plot has 1000(500) bins in the x(y) direction with counts higher than 200(2000) shown in yellow for $\mu_i(i)$. See the caption in Figure 6.1 for the explanation of a trait evolution plot.

Phase ③: final defective attractor. Eventually, the absolute majority of simulation runs end in a seemingly stable state of defection, as we can see in the final $\bar{\mu}_i^T$ distribution (bottom right and upper right plots in Figure 6.12). The mean expected investment remains around $\bar{\mu}_i \approx -4.5$ until the end of the simulation. Indeed, as seen in the single system realizations, the probability of synergistic resource extraction resources is minimal. Inspection of the data reveals that $\bar{p}_s \approx 0.05$ and that the investment reaches the highly negative value of $\bar{i} \approx -80$. Approximately $N \approx 550$ agents coexist in this dynamical phase taking in all of the available synergistic resources $\mathcal{A}_s$ per time step. Using these numbers, we can develop an intuition of how the system roughly behaves on average within a time step. At any time during this dynamical phase, $N \cdot \bar{p}_s \approx 28$ agents intend to grab the synergistic resources. The effective interaction network is minimal. As we know, synergistic resources are limited ($\mathcal{A}_s = 10^3$) such that a single defector that actually extracts resources grabs and destroys a total of $|i| + |i| \cdot r \approx 320$ resources (see equation 4.17). Therefore, approximately only three agents actually receive synergistic resources, and all others do not get anything but potentially need to pay for the collateral damage if they interacted. The lower the investment would be, the fewer agents would receive resources; thus, the fewer agents could survive from the synergistic resource intake. Why does *ReCooDy* not evolve a single surviving agent but instead a minimal set of agents? A single defective agent that extracts synergistic resources on its own will have a net loss in resources because it needs others to pay for its destruction. With $r = 3$ and using equation 4.15, we notice that a single separated defector would receive $P_{v,a} = (r-1)i_a = 2i_a < 0$. The damage caused by negative synergies, which exceeds the amount of grabbed resources, thus needs to be shared with others in order for the defector to be successful. There seems to be a minimal set of successful defective agents required such that some defectors indeed can profit from their defective strategy. This argument could explain why the system does not evolve lower investments but gets trapped in the defective attractor.

Extreme Synergy

Here, we investigate how the model dynamics change if we go to extremely high synergies. Let us choose the synergy factor of $r = 10$ and explore the model dynamics. From the analytical considerations in chapter 5, we know that for such high synergy factors, the social dilemma that the agents' experience is very weak or even not present at all, especially because they can evolve to optimize their social environment accordingly. However, the limited resource distribution dilemma, which we encountered early in this section, increases because higher synergy factors make individual agents receive huge resource shares, leaving other agents empty-handed. The individual's social dilemma is weak, but the competition for limited resources is strong.

Let us look at simulation results for $r = 10$. Figure 6.14 shows the long-term evolution of 128 system realizations that differ only in their random number sequence, i.e., the seed. it contains the mean expected investment evolution (left) together with its final distribution (right) and two instructive example expected investment evolutions (middle and bottom). In Figure 6.10, we already encountered the regime of $r = 10$ on medium time scales.

The system realizations show that there are two qualitatively distinct trajectories, one resulting in a cooperative population and one resulting in a defective population on average. The initial dynamical phase crucially determines which trajectory a model realizes. At the beginning of each simulation, the population collectively evolves positive investments before a cooperative branch emerges. Most agents focus on extracting basic resources, and only a small fraction of the population interacts to extract synergistic resources. Similar to $r = 3$, the cooperative branch evolves higher and higher (expected) investments, eventually resulting in a collapse of the cooperative branch and the death of all high-investment agents. We again observe a Red Queen dynamics. While the cooperative branch evolves ever-higher investments, the remaining population exhibits a tendency to evolve lower investments; thus, the loner subpopulation becomes slightly more defective in their infrequent interactions. Again as for $r = 3$, the traits of the surviving population after a collapse are crucial. If the investments are positive, the population develops ever-higher values, while, if they are negative and low enough, they evolve negative investments. We recognize that the latter, the evolution of defective populations, happens qualitatively similar to the $r = 3$ regime; Meanwhile, the former, the evolution of cooperation, exhibits qualitative differences. We explore both attractors in the following.

Attractor ①: stable cooperation with final monopolies. For $r = 10$, the population evolves continuously increasing investments collectively, while the agents' probability of extracting synergistic resources p_s stays roughly constant only slightly above $p_s \approx 0$ (see Figure A.12). As we can see from the μ_i evolution of the corresponding example realization (middle row), frequently high investment agents quickly evolve but immediately die out again. They survive only on short time scales. Therefore, more defective agents do not have time to evolve and successfully exploit cooperators. Instead, the majority of agents evolve high investments in the long run but seldomly use them to extract synergistic resources. They become cooperators that rarely interact and strongly rely on the basic resources for survival. Still, if they interact, they are hugely cooperative, such that we observe the on average evolving cooperation. This collective evolution of infrequent cooperation is stable as no simulations show dying populations, which happens eventually in the long run for $r = 3$. Instead, all simulations reach a stable final cooperative attractor around $\bar{\mu}_i \approx 20$. Inspection of the data reveals that the population consists of $N \approx 600$ with $\bar{p}_s \approx 0.02$. I.e., on average in this regime, $N \cdot \bar{p}_s \approx 12$ agents intend to extract $N \cdot \bar{\mu}_i \cdot r \approx 1.2 \times 10^4$ resources per time step. However, there are only $\mathcal{A}_s = 10^3$ synergistic resources available. Therefore, on average, only 1 agent gains all synergistic resources per time step, and 11 agents invested but do not receive any resources. The successful agent evolves a monopoly. Increasing the investment further does not yield an advantage anymore because there is no competition and all resources already get extracted; Hence, further increasing investments in this situation equals a waste of resources without an additional gain.

Attractor ②: defective attractor. For $r = 3$, we already explored the long-term stable defective attractor. Here, qualitatively nothing changes. Still, a minimal set of a few successful agents is required such that the defective strategy can indeed be successful.

Overall, we observe no population to entirely die in contrast to the findings in $r = 3$. The system is more stable, even though the distribution dilemma is more extreme for $r = 10$. However, the individual social dilemma is weaker. In general, cooperation emerges most of the time for the extremely high synergy factor and remains stable throughout

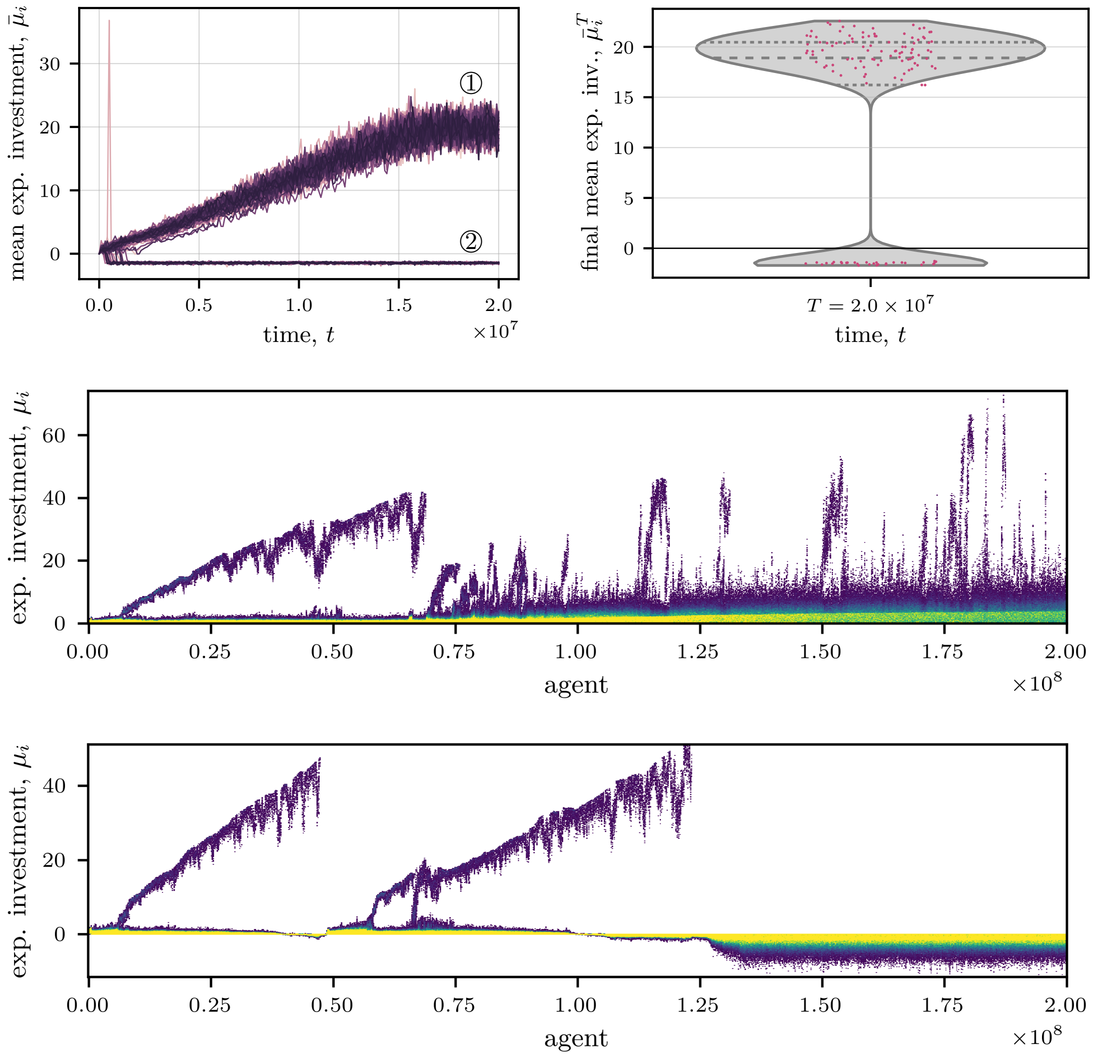


Figure 6.14: Long-term evolution of the extreme high synergy regime of $r = 10$ for 128 different seeds showing ① the cooperative attractor and ② the defective attractor. The time evolution of the mean expected investment $\bar{\mu}_i$ (top left) and its final distribution $\bar{\mu}_i^T$ after $T = 2 \times 10^7$ time steps (top right) with are shown together with trait evolution plots of μ_i that exemplify the evolution towards the cooperative attractor (middle) and the defective one (bottom). On the top left, each line represents one system realization. On the right, each dot represents $\bar{\mu}_i^T$ of a single system realization. The violin in the background shows the kernel density estimate. The middle and bottom plots contain 1000(500) bins in the x(y) direction each, with counts higher than 50 shown in yellow. Figure A.12 complements this figure by showing the trait evolution plot of p_s. See the caption in Figure 6.1 for the explanation of a trait evolution plot.

the simulation time. Cooperator branches that eventually result either in the death of the whole population or in the eventual evolution towards defection do only occur in the initial phase and have no great impact.

We have seen that in the regime of very high synergy, the system reaches either a cooperative attractor or a defective one in the long term. Ultimately, only the sequence of random numbers and the resulting microscopic configurations determine the populations' evolutionary trajectory, i.e., whether infrequent cooperation or defection evolves.

6.4.3 Summary and Discussion

For high synergies, cooperation always emerges from an initially neutral population but can eventually trigger a collapse into permanent defection or extinction. High synergy factors assure that plenty of goods are generated via synergies, in the in detail investigated cases three times and ten times the value of the initial investment. For such high synergies, the personal dilemma becomes minimal and can indeed vanish entirely dependent on the local network configuration (see also the social dilemma section in chapter 5). As a result, cooperation emerges each time in the observed simulations either within a specialized subset of the population or collectively within the entire population.

The emerging cooperators show ever-rising expected investments in a Red Queen dynamics pattern, i.e., they must constantly adapt in the evolutionary setting to survive and successfully compete against others. With individuals investing increasing amounts of resources, fewer agents obtain larger shares of the limited synergistic resources. However, cooperators need other cooperators to be successful and to survive exploitation through defectors. Cooperation is self-supportive. Thus, with fewer and fewer cooperators being able to survive, the effective self-support shrinks, making cooperators more vulnerable, which results in the eventual collapse of cooperation. If the entire population evolves cooperation as a bulk, this Red Queen dynamics can lead to the complete extinction of the population if no subpopulation evolves to specialize in basic resources before. However, if only a subpopulation evolves cooperation, the surviving population of basic resource extracting agents will reevolve cooperation. The cycle of emerging cooperation with Red Queen dynamics towards ever higher expected investments until the eventual collapse starts anew.

Importantly, agents do not actively choose to become ever more cooperative. Instead, heritable traits determine the behavior, and the evolutionary mechanism selects for high-investment agents through Red Queen dynamics. In a sense, their biotic and abiotic environment forces agents to invest more and more, thereby creating an increasingly hostile environment for themselves and each other to survive.

However, for a synergy factor $r = 3$, the Red Queen dynamics cycles are metastable and eventually collapse. Even though populations usually re-evolve a cooperative branch if the surviving population is cooperative, occasionally a defective population survives that does not re-evolve cooperation. Instead, the system remains on an attractor that exhibits the characteristic dynamics on a critical point: On medium time scales the population remains on the trajectory and slowly evolves either more cooperative or defective behavior as observed in phase ② in section 6.4.2. If investments slowly decrease at the beginning, eventually, the system trajectory gets quickly repelled and reaches the final defective attractor. Once in the final defective phase ③ in section 6.4.2, populations neither die out nor leave the attractor; If the population becomes defective, it stays defective.

The synergy factor, as well as the simulation history, determine whether the defective or cooperative attractor is reached. For a synergy factor $r = 3$, the absolute majority of realizations reach the defective attractor, while for $r = 10$, they predominantly reach the cooperative attractor. Only varying the sequence of random numbers influences the course of the simulation significantly. It determines which attractor is eventually reached, how long it takes, and which dynamical phases are realized before. After an eventual collapse of a cooperative branch, it is predominantly important which strategies the surviving agents have, which decides whether they reevolve cooperation or evolve defection. If the surviving population is cooperative, the emergence of a new cooperative branch is the rule; However, if the surviving population is defective, the emergence of a final defective branch is probable. We may see it as a manifestation of evolution's tinkering: What is available at a time shapes and determines what will be realized in the future, especially when new niches become available.

The observed Red Queen dynamics cycles eventually transition into a final attractive state in the long-term—either defective or cooperative—with infrequently extracting agents. Both are characterized by low probabilities p_s but vast positive or negative investments, respectively. I.e., agents rarely extract synergistic resources but invest huge amounts of resources if they do. In these attractive states, the effective network is minimal, and only one or a few agents are successful per time step, indeed receiving resources. All other interacting agents receive nothing but still have to pay for their investments or the created bads, respectively. Within the observed time frame, no population was able to leave such an attractive system state. Thus, even though agents evolve cooperation or defection in the long run, they evolve to interact rarely at the same time.

The simulations reveal yet another facet on the social dilemma, which agents experience. A qualitative new dilemma emerges with increasing cooperation in the limited environment—a distribution dilemma with cooperators as effectively selfish actors that are driven to invest more and more. Highly cooperative agents invest much to extract a lot of resources for themselves and their social environment. However, because resources are limited and successful agents get to extract first, cooperators leave less or even nothing for other agents. Larger individual investments result in larger shares for successful strategies. Therefore, the higher the cooperators's investments are, the fewer cooperators can survive from the available synergistic resource amount. Because cooperators need each other to survive, they undermine their survival when becoming too cooperative in the limited environment. Thus, a situation-dependent distribution dilemma emerges.

I will discuss the results in more generality in chapter 7.

7 Summary and Discussion

Within each result section, I summarized and discussed the respective detailed results (see section 6.2.2, section 6.3.4, and section 6.4.3) obtained from analyzing and simulating the newly introduced *ReCooDy* model (see chapter 4). In this chapter, I will focus on the general comprehensive findings and their holistic discussion.

Computer simulations are a powerful tool enabling the investigation of intricate complex adaptive and evolutionary models (*Macal* 2016). Often, these are otherwise ungraspable analytically (*Holovatch* et al. 2017). The ideal basis of complex and multifaceted computer models is a performant, reliable, and comprehensive modeling framework. In chapter 3, I introduced Utopia, a comprehensible modeling framework for complex and evolving systems, that we developed within our research group collectively, to boost synergies, introduce software development best practices in our research to improve software quality, and facilitate model development during all stages[1].

As for every computer simulation model, there is a *risk of erroneous code* despite extensive countermeasures such as testing, which is especially true for the complicated code required for a model as complicated and complex as *ReCooDy*. Still, software development best practices help minimize the error potential, not only for the modeling framework development but also for the individual models. I implemented *ReCooDy* in the tested and reviewed Utopia framework and further includes automated model tests checking for the correctness of microscopic units as well as macroscopic expectations.

One focus of this work was to investigate the emergence of cooperation from an initially neutral state. Once cooperation evolved as a strategy in the first place, we know which mechanisms promote its evolution (see section 2.1.1). Most evolutionary games prescribe an existing fixed set of strategies. Where these strategies come from in the first place usually remains an open question. Even the less presuming continuous game formulations at least prescribe the population structure. In *ReCooDy*, cooperative as well as defective strategies need to evolve from consecutive small changes. Further, the population structure needs to evolve and develop. It is not prescribed, and there are no restrictions that assure fixed network properties such as a fixed mean degree. Through self-organization, highly connected networks of frequently interacting cooperative or defective populations can emerge as well as sparsely connected ones of infrequently interacting neutral populations. Both of which we indeed observe in the simulations dependend on the dynamical regime (see section 6.2 and section 6.3).

The results indicate that, within the range of studied situations, *the emergence of cooperation from consecutive small steps is the rule* rather than the exception. Already minute synergies can yield cooperative populations as long as individuals do not crucially rely on the outcome of interactions and their effective interaction network is minimal. Both requirements coevolved enabling a slow but steady emergence of cooperation (see section 6.2). It corresponds to what we observe in our world, as cooperation is indeed

[1] The interested reader finds more information on the applications and experiences using Utopia in section 3.3 and a concise summary in section 3.4

ubiquitous, but contrasts expectations from simple EGT viewpoints, which often promote the necessity for additional mechanisms such as reciprocity, punishment, or rewards as solutions (see section 2.1). These boost cooperation once it exists. However, such mechanisms are relatively advanced and cannot sufficiently explain how cooperation emerges in the first place. Of course, also network reciprocity is a well-known mechanism promoting the evolution of cooperation (*Lieberman* et al. 2005; *Allen* et al. 2017); however, usually, models again presume the existence of cooperation as a strategy, not letting it emerge out of the blue. Further, network models usually prescribe a static or dynamically relinking network with a constant mean degree. Thus, they prescribe a population structure. In *ReCooDy*, the population structure evolves and dynamically self-organizes from agent traits, costly linking actions, and an initially unconnected population. Obviously, these evolution and development rules are still prescribed but on a higher level giving the population more flexibility to evolve, adapt and thereby construct a beneficial realization. As a result of the increased but not complete degree of freedom in shaping one's social environment combined with a cooperator's profit from creating cooperative offspring, cooperation can emerge even in hostile social environments that offer just minute synergies. Hereby, the driving force towards cooperation is accessing an otherwise inaccessible resource. In *ReCooDy*, in minute-synergy settings, a form of cooperation emerges out of the blue because agents are not crucially dependent on the success of the interactions but still sometimes profit from otherwise inaccessible resources.

For intermediate and high synergy, *agents specialize in strategies that coexist* and rely either on basic resources or synergistic ones. Cooperation evolves through agents that depend crucially on synergistic resources. They evolve high probabilities to extract synergistic resources and therefore devolve their ability to extract basic resources. The access of previously inaccessible resources is the driving force primarily in the initial phase. Still, after specialization, the agents' evolutionary success is predominantly determined by the competition against other specialized agents. It allows for more intricate system dynamics and richer phenomenology such as Red Queen dynamics, which we will come back to below. In intermediate and high synergy factor regimes, success against the specialized competitors becomes the system's driving force.

Usually, the simulations show that *defection requires the existence of cooperation to emerge but then emerges as a natural consequence. ReCooDy* not only deals with the emergence and evolution of cooperation but also focuses on the emergence and evolution of defection. In contrast to the typical definition of defection—paying no cost in the PGG but profiting from the created goods—I introduced the concept of true defection in this work within the gPGG: the act of selfishly taking resources while destroying goods and generating collateral damage (see mathematical formulation in section 4.2.2.2). Therefore, I introduce a worse kind of defector within the PGG setting, which increases the potential of cooperators to get exploited. However, defection needs to emerge in the first place because all simulations start as neutral, unconnected, and non-interacting populations. As for cooperation, instead of assuming an already existing set of (typically binary) strategies like cooperation and defection, *ReCooDy* contains the option for varying degrees of cooperation and defection to evolve from consecutively accumulated small trait mutations. But agents do not need to evolve cooperation or defection because they could, in principle, just live from basic resources alone. Defection does not immediately evolve from a neutral population because defectors harm themselves via their destructive nature, which only becomes a competitive strategy if they grab sufficient resources and distribute their destruction costs. If there is nobody to exploit, there is no incentive to evolve defection; therefore, defection only emerges in the first place if cooperation already exists, but then as a natural consequence.

We frequently observe Red Queen dynamics yielding increasingly defective agents in medium synergy environments and increasingly cooperative agents in high synergy ones (see section 6.3 and section 6.4, respectively). Because resources are limited, and more extreme strategies come with bigger individual resource shares, the number of agents that can survive with the corresponding strategy shrinks. However, cooperators need other cooperators to support and shield themselves against defectors; And defectors need other interacting agents to pay for the destructed goods else they effectively hurt themselves. Therefore, both Red Queen dynamics towards rising cooperation and rising defection collapse eventually, potentially resulting in the death of all predominantly interacting agents or even the entire population. *ReCooDy* does not implement any active measures that could prevent the eventual collapse of (sub)populations. The results indicate that active prevention mechanism against such Red Queen dynamics would be needed to stop a collapse of a (sub)population.

The long-term evolution of *ReCooDy* often exhibits a *final state of defection* or, for huge synergy factors, a *final state of cooperation.* However, the quality of cooperation and defection is different from preceding dynamics regimes as the respective magnitude of agents' investment is vast. Still, they tend not to extract synergistic resources. Only a tiny fraction of the population interacts to extract resources, and even fewer actually receive a benefit. The effective social network is minimal, while the agent's commitment to the interaction is maximal. Which dynamical phase precedes the attractive final states depends on the synergy factor r. For $r = 1.4$, we observe the previously mentioned Red Queen dynamics towards ever rising defection with high probability of extracting synergistic resources p_s, while for $r = 3$ and $r = 10$, the transition is less drastic as p_s remained nearly non-existing. The results indicate to expect defection as final stable state predominantly. However, we should be aware that there is no possibility to escape from this final state because *ReCooDy* does not implement a mechanism that could lead out of the trapped system state. In a sense, we could also interpret the final state as a model deficit. In the real world, more than three million agent generations, which roughly corresponds to the 2×10^9 agents living during the long-term simulations, will not pass without evolution coming up with new inventions and innovations as evolutionary pressures change in varying environments over time(*Murugan* et al. 2021). For example, cooperative relations between hosts and symbionts typically coevolve such that symbionts compete in the environment provided by their host, while the latter evolve to control and shape their symbionts (*Foster* et al. 2017). Such control mechanisms evolve together with and as a reaction to the cooperative benefit created by the emergent cooperative symbiosis. Even in a simple laboratory experiment of one single initial bacteria population evolving over more than 6×10^4 generations ($>$ 25 years), *Lenski* (2017) summarizes: "We have quantified the dynamics of adaptation by natural selection, seen some of the populations diverge into stably coexisting ecotypes, described changes in the bacteria's mutation rate, [and] observed the new ability to exploit a previously untapped carbon source [...]". As most evolutionary models, *ReCooDy* is not comprehensive enough to adequately represent the very long-term evolution, which is especially true for humans and their unseen impact on nature and evolution itself.

The system's trajectory crucially depends on its history. Changing the sequence of random numbers alone changes the set of actually realized dynamic phases from the set of possible dynamical phases in the transition regime (section 6.3). It also determines which final attractive state is reached for high synergies and how long it takes (section 6.4). Agents accumulate changes in nine different traits, some of which are not strongly selected for or against within specific dynamic phases. However, after a collapse, the combined traits of the surviving population determine the new initial population that can reconquer the newly available niche by taking a potentially new evolutionary path after a population bottleneck. What we observed in a simple form in *ReCooDy*, is a key property of evolution, which we can already deduce from simple bacteria experiments (*Blount* et al. 2012; *Lenski* 2017) but is a general property of classic evolutionary systems (*Lande* 1988): What is present at a given time in history determines what will be possible in the future.

ReCooDy's dynamics in the transformation regime *shows characteristics of complex and chaotic systems.* Its macroscopic phenomenology exhibits a set of attractors and metastable dynamical phases that eventually transition to other attractive or repellent states. Within each dynamical phase, *ReCooDy* self-organizes into a metastable system state, which eventually transitions often through collapses caused by exceeded thresholds and cascaded agent deaths. Such self-organized critical behavior is a typical property of a complex system. In *ReCooDy*, the synergy factor determines the forcing. In contrast to a typical complex system, this forcing is not global but local, intensifying the distribution inequality of the constant inflowing resources. Besides *ReCooDy*'s complex nature, we see that the sequence of random numbers significantly alters the system's trajectory. It determines which metastable dynamical phases the system indeed reaches. Changing the sequence of random numbers can result in entirely different trajectories (as visually exemplified in Figure 6.5). After a transition, we cannot predict which metastable dynamical phase the system reaches next. Further, we observe that the system reaches completely different attractors in the long-term, which previously did not exist, due to the inherently complex model dynamics. Before the system reaches its final attractors, the long-term behavior of *ReCooDy* is not predictable— a characteristic of a deterministic chaotic system. Usually, models of chaotic systems focus on low-dimensional state spaces, equal or slightly above two-dimensional for discrete and three-dimensional for continuous state spaces. In *ReCooDy*, the dimensionality is much higher, which allows the system's self-organization. Future research should further investigate *ReCooDy*'s chaotic and complex nature, for example, by quantifying critical exponents and critical points. Still, from the macroscopic phenomenology, we can qualitatively classify *ReCooDy* as a deterministic chaotic self-organized system.

One of the main results presented in this work is the reflection on the *nature of the effectively emerging social dilemma* obtained by simple analytical considerations. In chapter 5, I presented the varying notions of the social dilemma, which I summarized, exemplified, and shortly discussed in its final section 5.6. Let me condense the key messages:

1. **Tragedy of the Commons:** By construction, agents have a microscopically defined competitive social dilemma captured in the gPGG, which is independent of the synergy factor. On its own, it models an extended version of the tragedy of the commons (see section 5.1).
2. **Personal dilemma:** However, because the network structure prescribes multiple interactions, *agents can indeed profit from increased investments* within a constant network above a critical synergy factor. The latter only depends on the agent's local network configuration, i.e., its specific social environment (see section 5.2).
3. **Competition dilemma:** Even if investing agents have a personal profit from increased investments, usually, their neighbors profit even more. Nevertheless, the relative profit shrinks with an increasing synergy factor and can even turn into a setting where agents yield higher total payoffs than their neighbors (see section 5.3).
4. **Offspring benefit:** Generally, a cooperator's offspring benefits the cooperator, while a defector's offspring harms the defector. This property not only depends on their respective personal strategy but also on their strategy relative to their neighbors. The latter can even turn offspring benefit into harm, for cooperators in a much more cooperative neighborhood, or harm into benefit for defectors in a much more defective neighborhood (see section 5.5).
5. **Distribution dilemma:** For excessive levels of cooperation (or defection), a distribution dilemma arises. It effectively turns cooperators into selfish actors that take vast shares and leave nothing for other cooperators. This distribution dilemma arises as a consequence of limited resources, specifically due to *ReCooDy*'s limited resource-inflow. It is not a property of the gPGG in itself but observable in the simulation results for intermediate and high synergy factors in section 6.3 and section 6.4.

The results indicate that reducing the model interaction to a simple tragedy of the commons (public goods) dilemma is not necessarily an adequate representation of the agents' experienced interaction setting. For a fixed synergy factor, we could, in principle, partition agents into different groups experiencing qualitatively different social dilemma settings dependent on their respective social environment; E.g., agents that experience a competition dilemma but no personal dilemma. Some agents would not even experience any dilemma for specific synergy factors and directly profit from being cooperative, being effectively selfish. Furthermore, an agent can theoretically traverse multiple dilemma qualities during its lifetime while consecutively building and removing links to others. Already simple analytical considerations, arguably even simplistic ones, enabled extracting these results. For a two-player game with active linking, *Pacheco* et al. (2006) used effective parameters to show that linking can effectively transform games from a prisoner's dilemma (PD) to a coordination game (CG) or from a snowdrift game (SD) to a harmony game (HG). However, the results presented in this work question whether there are meaningful, effective descriptions of social dilemmas that adequately describe all facets of the qualitative microscopic variance in strategy. Even if it is possible for simple models such as in *Pacheco* et al. (2006), it is not clear whether it is adequate in more complicated situations such as in *ReCooDy* or real-world systems. Moreover, we cannot reduce an agent's strategy to being selfish or selfless as both are emergent properties in *ReCooDy* dependent on local population structure, environmental conditions, and states[2].

The results indicate that adding more realism exhibits an *unintuitive and rich phenomenology*. It relies on the interplay of processes operating on similar time scales such that they can react mutually and create feedbacks that produce emergent structures and evolutionary pressures. For example, the limited resources only start affecting the population dynamics when the agents' total resource extraction exceeds the available resources flowing into the system. Typically, they already evolved a specialized strategy by then. The emerging distribution dilemma comes hand in hand with increased evolutionary pressure. Agents either evolutionary adapt to the new circumstances or cannot survive when the distribution inequality maximizes eventually, which is often triggered by changed linking behavior. Linking costs prevent extensive linking, which potentially reduces the individual dilemma, but can result in exhausting spendings on linking over the agent's lifetime. Without or with low linking costs, the system's trajectories could be expected to look differently, which could be a focus of future work. As we observed in the simulations, agents evolved a metastable strategy invading other agents that specialize in basic resource extracting. The former evolved huge strengths that guaranteed success against the latter, however, with guaranteed high net resource losses (see section 6.3.3.2). Such sophisticated strategies and model behaviors exist due to the lifetime-accumulated development and impact of agents' actions. On the mesoscopic level, we frequently encounter such surprising relations, most notably when we explored the intricacy of the emerging social dilemma, which indicates that we do not have an intuitive understanding of such systems.

ReCooDy is not a typical physics system with a prescribed structure; instead, dependent on the system state, a *structure emerges* due to the strong interactions between the state and the system rules. The hierarchically nested microscopic rules result in complicated structure and dynamics a few hierarchical levels above. For example, we could find rules describing the macroscopic behavior of individual cooperator-defector coexistence regimes as observed in the transition regime (see section 6.3). However, these rules would not be capable of describing the system behavior in the regime of increasing defection or the afterward following regime of light defection. The observed time-varying set of reachable, potentially emerging attractors raises a severe challenge. The rules defining the macroscopic behavior change over time. The system's quality is entirely different from a typical physical system, such as one described through Newtonian mechanics.

Some presented results are based on a heuristic approach by *hypothetical explanations* of the observed simulations in combination with a check for consistency of these hypotheses with analytically deduced expectations for minimal settings. Currently, to the best of my knowledge, such an approach is the only practically feasible one, as a comprehensive rigorous mathematical formalism does not exist for complex systems(*Holovatch* et al. 2017), especially with the complexity *ReCooDy* encompasses. While the phenomenology exposed by *ReCooDy* is already fairly complicated, real situations, particularly in highly

[2] Thinking further, we could even ask: "Does altruism exist?" (*Wilson* 2015)

developed societies as in humankind's, may be expected to be more complicated. What aspects of them, if any, can be usefully represented by models of the current class remains to be seen. The results presented in this dissertation are a small step towards a more holistic modeling approach in eco-evolutionary settings.

The presented results entail only a *minute fraction of exploration possibilities*, leaving many paths open for future investigations. A few example paths are the impact of linking costs, the influence of global versus local linking, and the system size that could be explored without the need to change any existing code. Future work could also focus on the application of *ReCooDy* to a real-world scenario by working out the specific details and systematically explore the respective system, all of which would, however, require a lot of time and effort mainly on the data-side. More far-reaching extensions comprise the introduction of multiple distinct resources providing separate niches, coupling *ReCooDy*'s population on a spatial grid, which would be operationally possible with Utopia, or the introduction of more advanced information processing capabilities for agents to take decisions such as through evolving artificial neural networks (*Stanley* et al. 2019). These proposed extensions embrace the idea of increasing model complexity to increase realism with the risk of potential conceptual and operational challenges.

To fundamentally explain complex coupled systems, *we need fundamentally complex models* in future research. The principles of this direction within the field of EGT are noticed and promoted in recent years (*McNamara* 2013; *Akçay* 2020), however still to a less comprehensive degree than presented in this dissertation in my perception. As I motivated in the introduction and theory, general statements extracted from simple model systems can become invalid if more realism or complexity is introduced. Fundamental questions in eco-evolutionary systems cannot be answered by simple models. The findings presented in this thesis show the potential of less restricted, more comprehensive modeling as a complementing approach to simple models. Modeling without analytical feasibility in mind, e.g., by omitting omnipresent weak-selection limitations or allowing several coevolving traits instead of one or two, can exhibit a rich phenomenology from which we can extract lessons to apply to real systems. In order to fundamentally understand the origins of complex and evolving systems, we presumably need to embrace more modeling complexity and coupled processes—a scientific path successfully taken in other fields of research such as climate systems research.

8 Conclusion and Outlook

> "Now, here, you see, it takes all the running you can do, to keep in the same place."
>
> Red Queen to Alice in *Carroll* (1871)

Our world, with its inhabitants and societies, is utterly complex and evolving at an unseen rate within a rapidly changing environment with crucial feedback – exceedingly forced and shaped by humanity's cultural and technological (r)evolution. We only start to recognize, let alone understand, its inherent complexity and interwovenness. Once we notice the importance of process coupling and evolution's capacity to adapt rapidly to new situations through tinkered inventions and innovations, we see the necessity for comprehensive modeling for eco-evolutionary social systems. In the context of social evolution and Evolutionary Game Theory, the necessity for richer modeling approaches already get promoted (*McNamara* 2013; *Akçay* 2020) but often still focussing on one or two processes at a time.

In this work, I presented *ReCooDy* that I see as a first step towards much more comprehensive modeling; an attempt to take a potentially game-changing perspective on the question of how cooperation and defection emerge and evolve. It incorporates resource-flows and couples Evolutionary Game Theory, population dynamics and agents' lifetime development, as well as the coevolution of nine traits. Simulations unveiled a rich phenomenology, intricate history-dependent dynamics with metastable states of strategy coexistence, sudden transitions of dynamical phases, and Red Queen dynamics towards exhausting and eventually deadly strategies within a limited environment.

We observed the emergence of cooperation for minute synergies out of the blue from consecutive small mutations – how evolution fundamentally operates – that also optimized conditions in the given setting by minimizing the interaction structure and frequency to make new resources slowly accessible and usable. For higher synergy and only after cooperation existed, defection emerged, frequently creating instability and at times resulting in collapses due to an evolutionary arms race followed by a population bottleneck from which new strategies could evolve. For very high synergy, cooperation could turn even into an inherently selfish strategy within the limited environment. When we do not presume the existence of cooperation and defection a priori but let them evolve within a continuous strategy space and the aim to access otherwise inaccessible resources, then the results indicate that the emergence of cooperation is the rule rather than the exception–as observed in nature and with vastly greater extent human societies–and that true defection can effectively evolve as a response.

The agent-centric simple analysis of the context-dependent social dilemma revealed a multi-facetted emergent setting that, together with the emerging distribution dilemma, exhibits a much more complicated setting as suggested by the microscopically implemented generalized public goods game. We could partition the population into context-dependent groups of qualitatively distinct emerging dilemma settings for a fixed synergy factor. Hence, in *ReCooDy*, selfishness and selflessness are emergent context-dependent interaction properties that cannot be captured adequately in a single effective number because they can exhibit different qualities. It leads to the broader question of whether a reductionist view on selfish or selfless behavior in evolutionary social systems supports the required complexity for systems outside of simple model settings.

In general, we need further research to support the derived explanations, claims, and results and check for their scope. As mentioned and discussed in the previous chapter, we should interpret the presented work cautiously because they come from initial steps into a new direction of comprehensive modeling.

Although more realistic modeling comes with severe conceptual and operational challenges, the results presented in this work indicate that its capabilities open new paths to address and eventually understand aspects of social dynamics that have turned into inseparable aspects of our physical environmental system. It may be a next step in the run-away dynamics between our understanding of social dynamics and the actual development of our society.

Acknowledgements

The authors acknowledge support by the state of Baden-Württemberg through bwHPC and the German Research Foundation (DFG) through grant INST 35/1134-1 FUGG.

I would like to express my gratitude and appreciation to Prof. Kurt Roth, my advisor, for his extraordinary mentoring, contagious optimism, and enthusiasm. This research would not have been possible without his guidance and extraordinary ability to motivate. Thank you for inspiring, for giving ideas, and for the discussions that expanded my horizon.

I want to thank Dr. Hannes Bauser, the co-advisor of my dissertation, for his time, valuable reflections, and guidance during this research. Further, I would like to thank Prof. Dr. Fred Hamprecht, co-advisor of my dissertation.

I want to thank my second referee Prof. Björn Malte Schäfer, who immediately agreed to be my second referee.

Big thanks to the chaotic, complex, and evolving working group for creating an inspirational, motivating, and amicable atmosphere, foremost to my fellow Ph.D. students and friends Yunus Sevinchan, Lukas Riedel, and Harald Mack. I thank Angelika Gassamer for her general assistance and for taking care of my plants during times of lockdown.

I am grateful to all the contributors of Utopia that enabled the collective endeavor of our research group, which indeed boosted our group work as well as group life. Thanks (in alphabetical order) to the moderators of the project Thomas Gaskin, Harald Mack, Lukas Riedel, Yunus Sevinchan, and Julian Weninger. Further thanks to all the contributors: Narek Baghumian, Lorenzo Biasi, Unai Fischer Abaigar, Leonhard Holtmann, Christian Kobrow, Fabian Krautgasser, Daniel Lake, Hendrik Leusmann, Peter Manshausen, Robert Rauschen, Soeren Rubner, Laila Pilar Schmidt, Simeon Schreib, Lukas Siebert, Jeremias Traub, Josephine Westermann, and Paul Wiesemeyer. Utopia would, however, be unthinkable without the conceptual guidance, the inspiring discussions, and in general, the mentoring of our professor Kurt Roth.

Thanks to Mike Taylor for proofreading and Prof. Marina Solnyshkina for handing me his contact.

I thank all my friends for their great support and all the inspirational—often humorous—times throughout my life. Special thanks goes to Johannes and Sonja Haux, Daniel Lake, David Stöckel, Anna Lina Gummersbach, Michael Schneiß, Benjamin Aslan, Makan Rafiee, Joannes Anand, Yanic García-Muñoz, Philipp Breul, and last but not least David Brückner.

I am extremely grateful for all the support from Polina, without whom I probably would not have been able to finish this thesis. Thank you for your overall assistance, especially during the final months, your kind nature, and all the motivation that you give me.

Finally, my deep and sincere gratitude to my family for their continuous and unparalleled love, help, and support. Thank you, Mama, Papa, Clara, Peter, and Ioannis.

Appendices

A Supplementary Figures

In the following, I present additional figures that supplement the results presented in chapter 6.

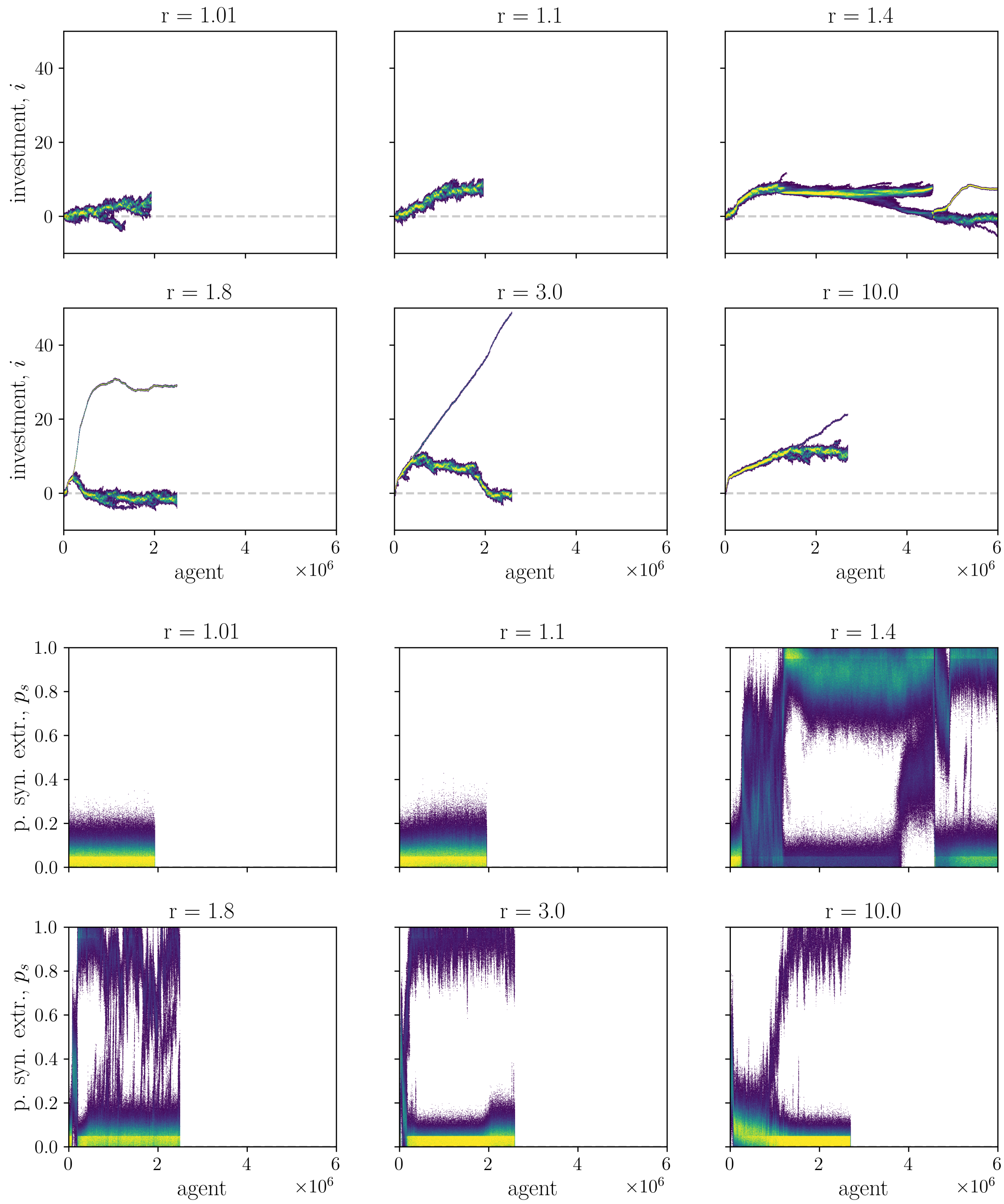


Figure A.1: Trait evolution plots of the investment i (top two rows) and the probability of extracting synergistic resources p_s in the short-term. This figure complements Figure 6.1 as it shows the evolution of the individual traits constituing the expected investment: $\mu_i = ip_s$. See the corresponding figure caption for the plotting details and an explanation of a trait evolution plot.

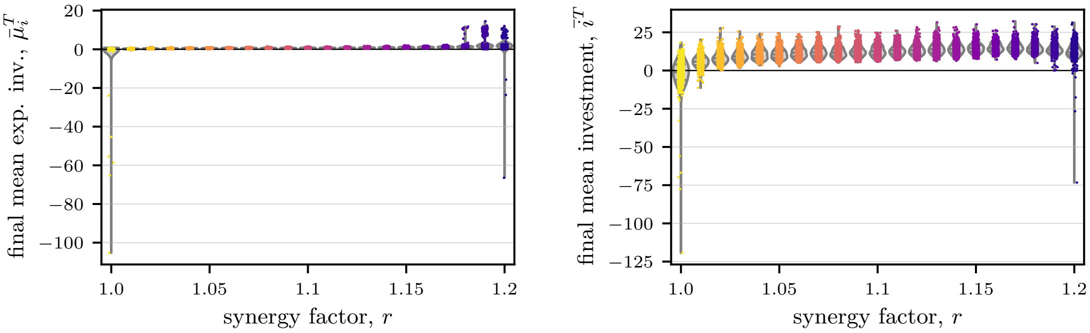


Figure A.2: Final mean expected investment distribution $\bar{\mu}_i^T$ (left) and final mean investment distribution $\bar{i}^T$ after $T = 5 \cdot 10^5$ time steps ($\approx 10^3$ generations) in their full value range for varying synergy factor r showing the emergence of cooperation regime. It complements Figure 6.2. See there for figure details.

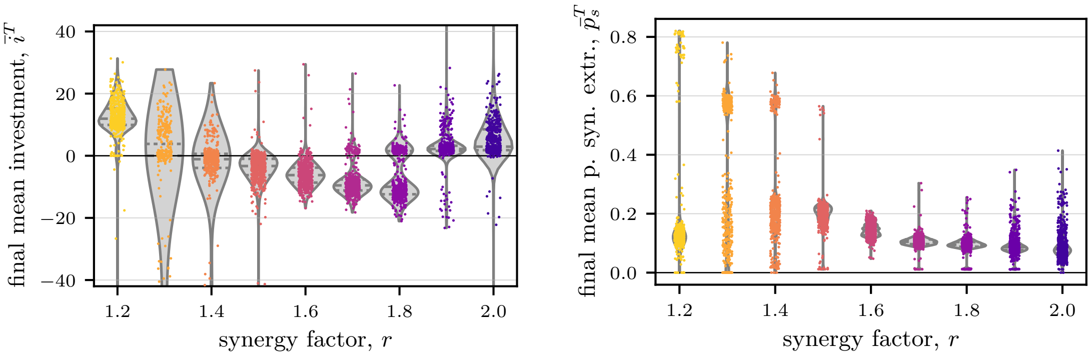


Figure A.3: The transition regime. The top row shows the time evolution of the number of agents N for different synergy factors r. The mean and standard deviation (shaded area) are calculated over system realizations. The bottom row shows the final mean investments $\bar{i}^T$ (bottom left) in a zoomed value range and the final mean probability of extracting synergistic resources $\bar{p}_s^T$ (bottom right). All simulations ran for $T = 5 \cdot 10^5$ steps. Each dot represents the final population average of one system realization, 512 per r. Color encodes rising synergy factors from low (yellow) to high values (blue). The violin plots in the background show the kernel density estimates of the distribution with width scaled according to equal areas. Lines within the violins show the median and quartiles of the corresponding distribution. This figure uses the same data as presented in Figure 6.4 but plots additional relevant quantities.

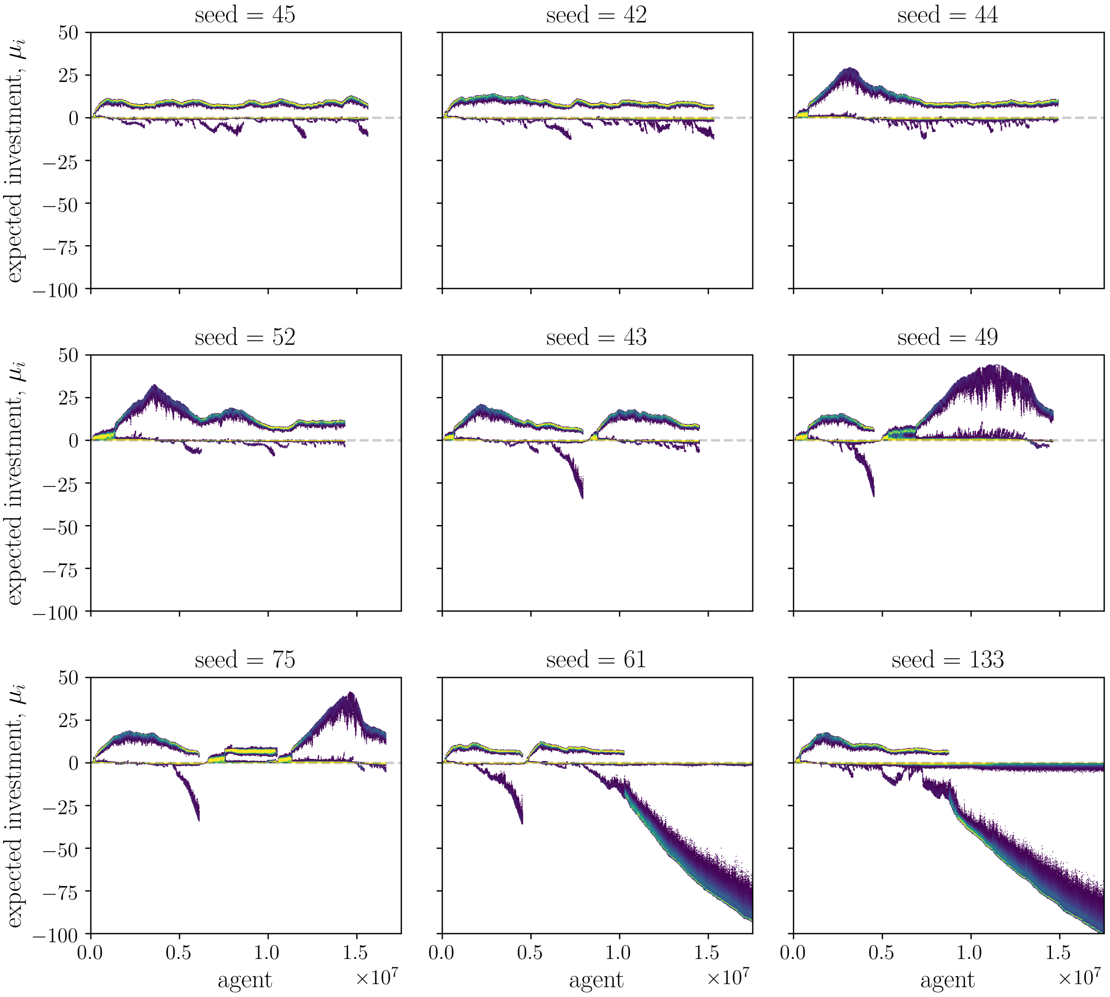


Figure A.4: Overview of simulation results for $r = 1, 4$ and 9 random number generator seeds. Trait evolution plots of the corresponding expected investment μ_i are shown in their full value range in contrast to a zoomed interval in Figure 6.5. See the caption in Figure 6.1 for the explanation of a trait evolution plot.

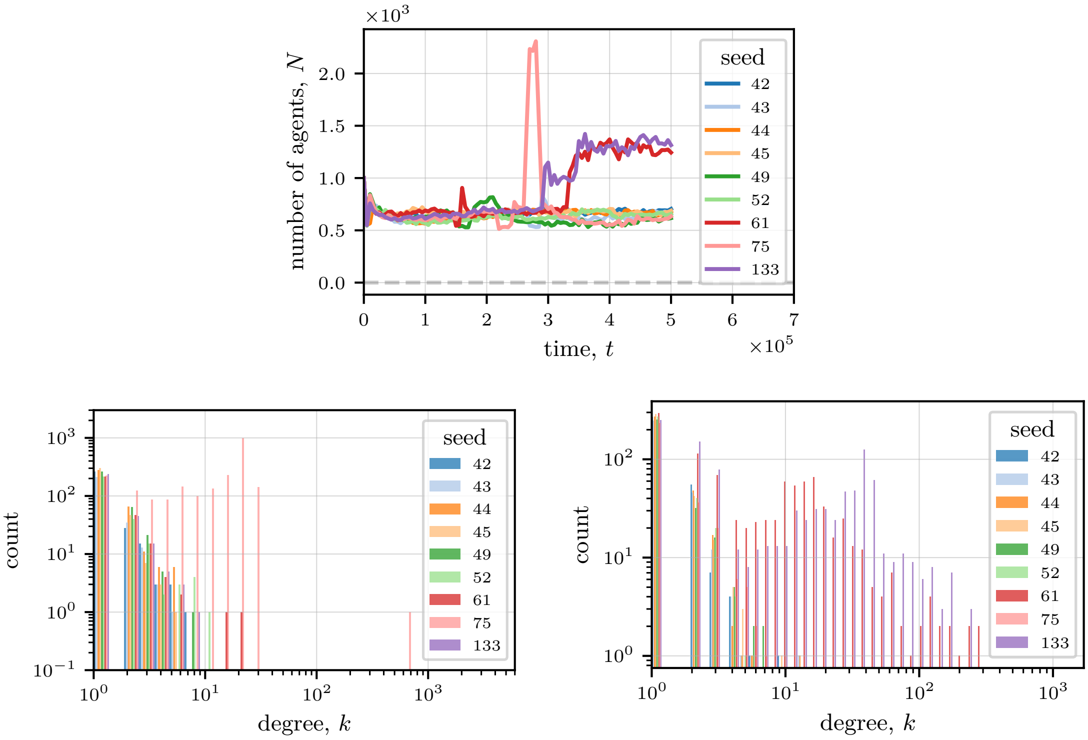


Figure A.5: Overview of simulation results for $r = 1, 4$ and 9 random number generator seeds showing the number of agents (top), the degree distribution at $t = 2.8 \times 10^5$ (bottom left), and the final degree distribution at $t = 5 \times 10^5$. In combination with the expected investment evolution (see Figure 6.5) we observe the following: A highly dynamic cooperative phase (seed 44 at $t = 2.8 \times 10^5$) yields more than thrice the number of agents (peak in top plot) and a single high-degree hub agent (bottom left). Phases of rising defection show approximately twice the number of agents and a higher degree than typical dynamical phases with a cooperative branch.

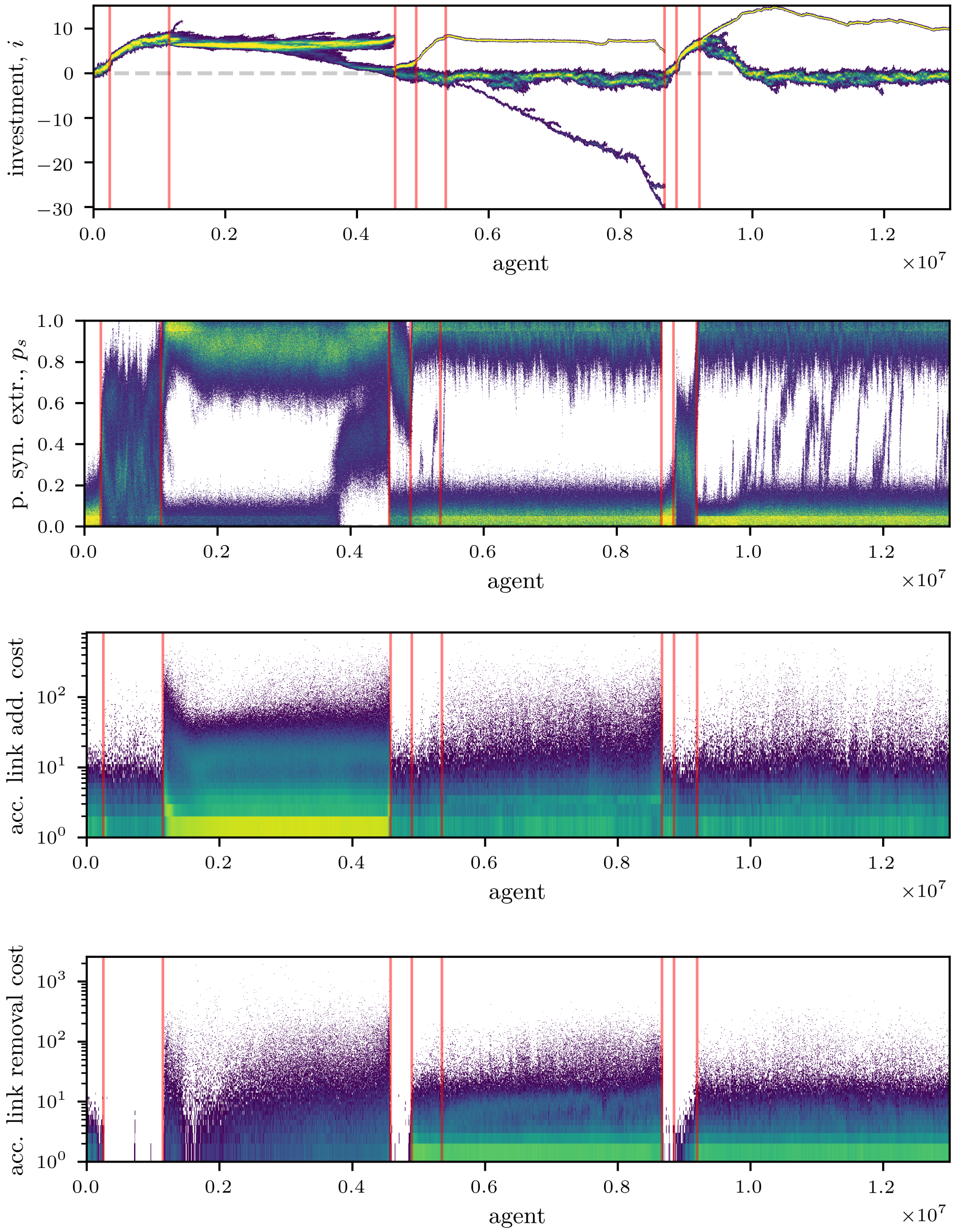


Figure A.6: Examples of transitions complementing Figure 6.6. Trait evolution plots of the investment trait i, the probability to extract synergistic resources p_s, and the accumulated link addition and removal costs are shown. The simulation ran for $T = 5 \times 10^5$ steps. See the caption in Figure 6.1 for the explanation of a trait evolution plot.

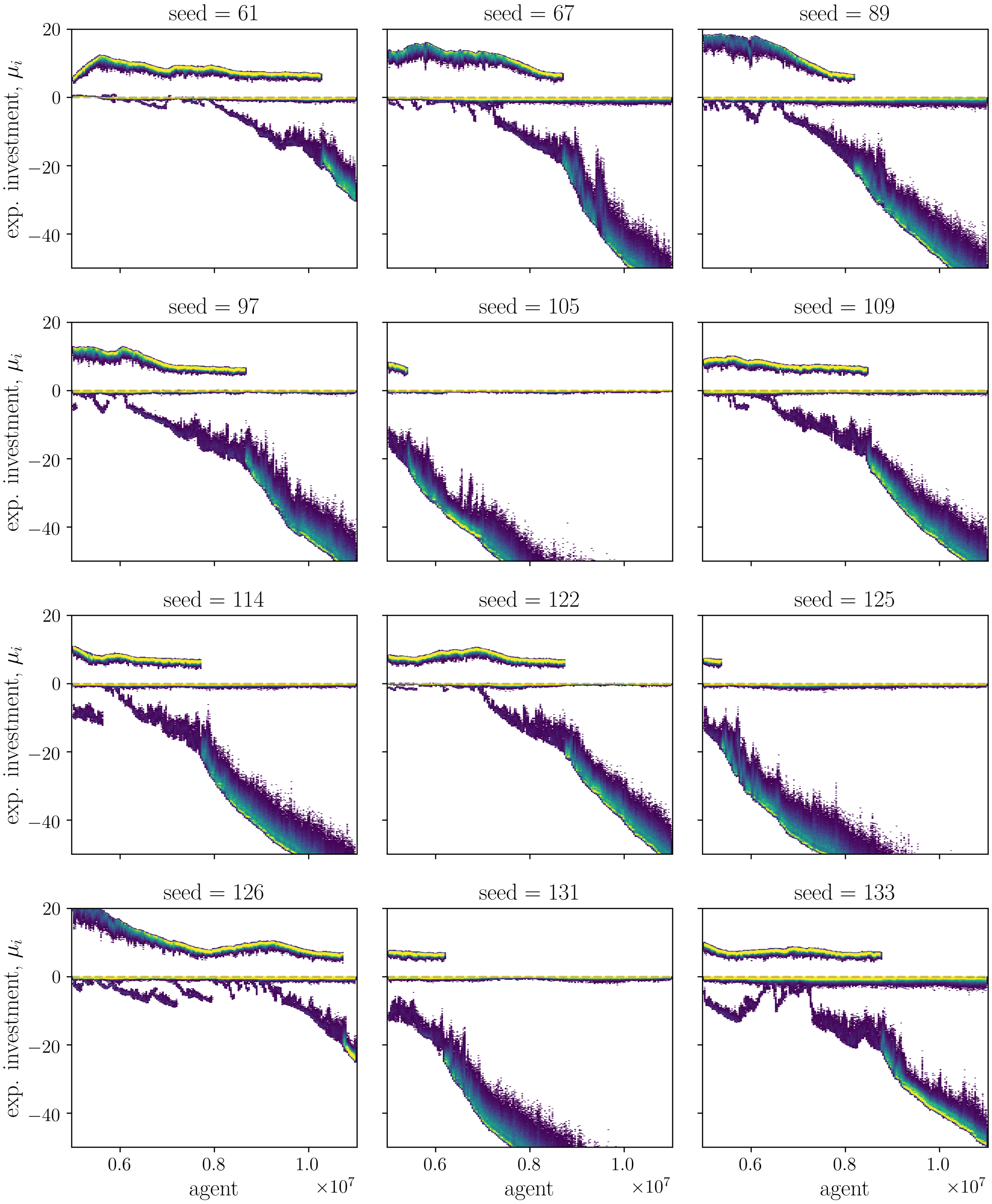


Figure A.7: Examples of triggers into the rising defection phase for $r = 1.4$. Trait evolution plots of the expected investment μ_i are shown for a selection of 12 seeds that all exhibit a transitions into growing defection. Each plot has 500 bins in each dimension with counts higher than 200 shown in yellow. See the caption in Figure 6.1 for the explanation of a trait evolution plot.

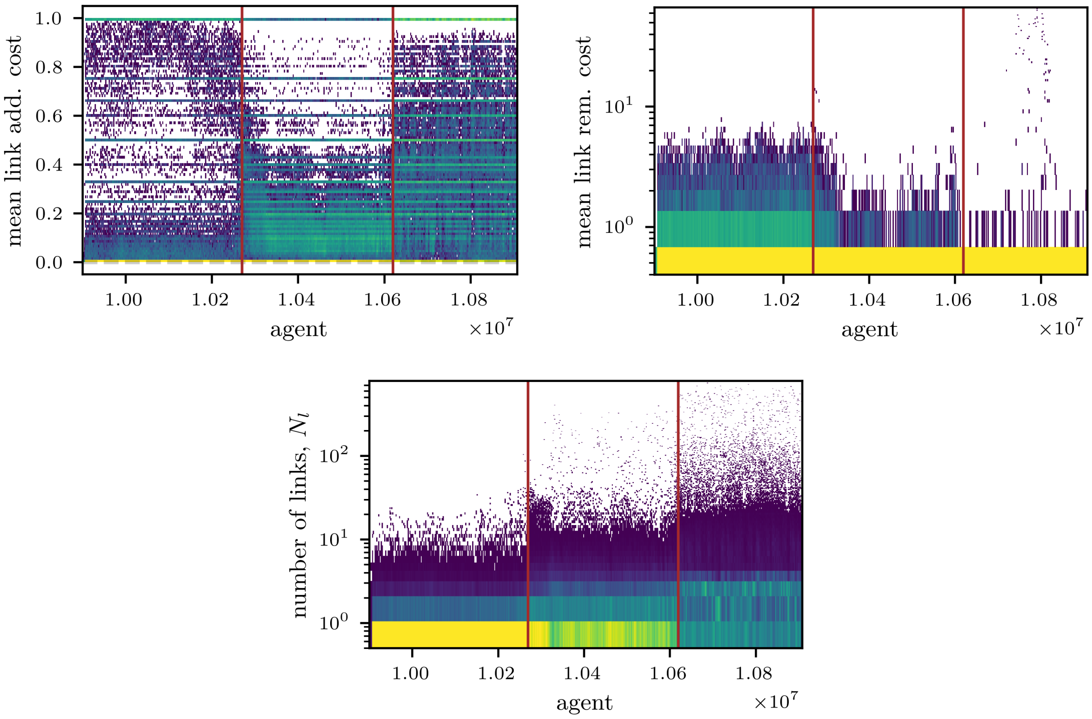


Figure A.8: Collapse of cooperation and transition towards growing defection happens in two main steps (red and orange line) that significantly alter agent traits and the population structure. This figure complements Figure 6.8. It shows the mean link addition and removal costs (top) averaged over an agent's lifetime and the total number of links an agent accumulates over its lifetime in trait evolution plots. The simulation ran for $T = 5 \times 10^5$ steps. See the caption in Figure 6.1 for the explanation of a trait evolution plot.

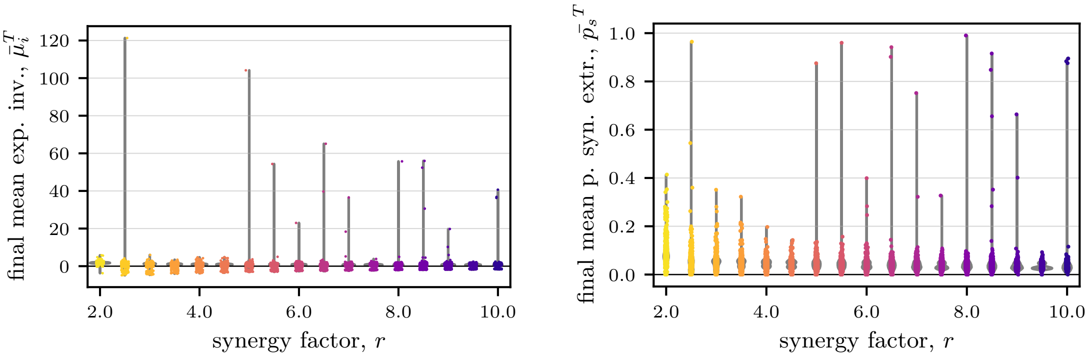


Figure A.9: Full value range of the high synergy regime complementing Figure 6.10. The mean expected investment time evolution of the population averaged over 512 simulation runs each (left) and their final distribution at time $T = 10^5$ after approximately 500 generations (right) depending on the synergy factor r. For clarity, we only show a selection of r on the left. The shaded areas show the standard deviation over varying system realization. Each dot represents the final population average of a single system realization (512 per r). Color encodes rising synergy factors from low ($r = 2.0$; yellow) to high values ($r = 10.0$; blue). The violin plots in the background show the kernel density estimate of the distribution. The width of each violin is scaled according to equal areas.

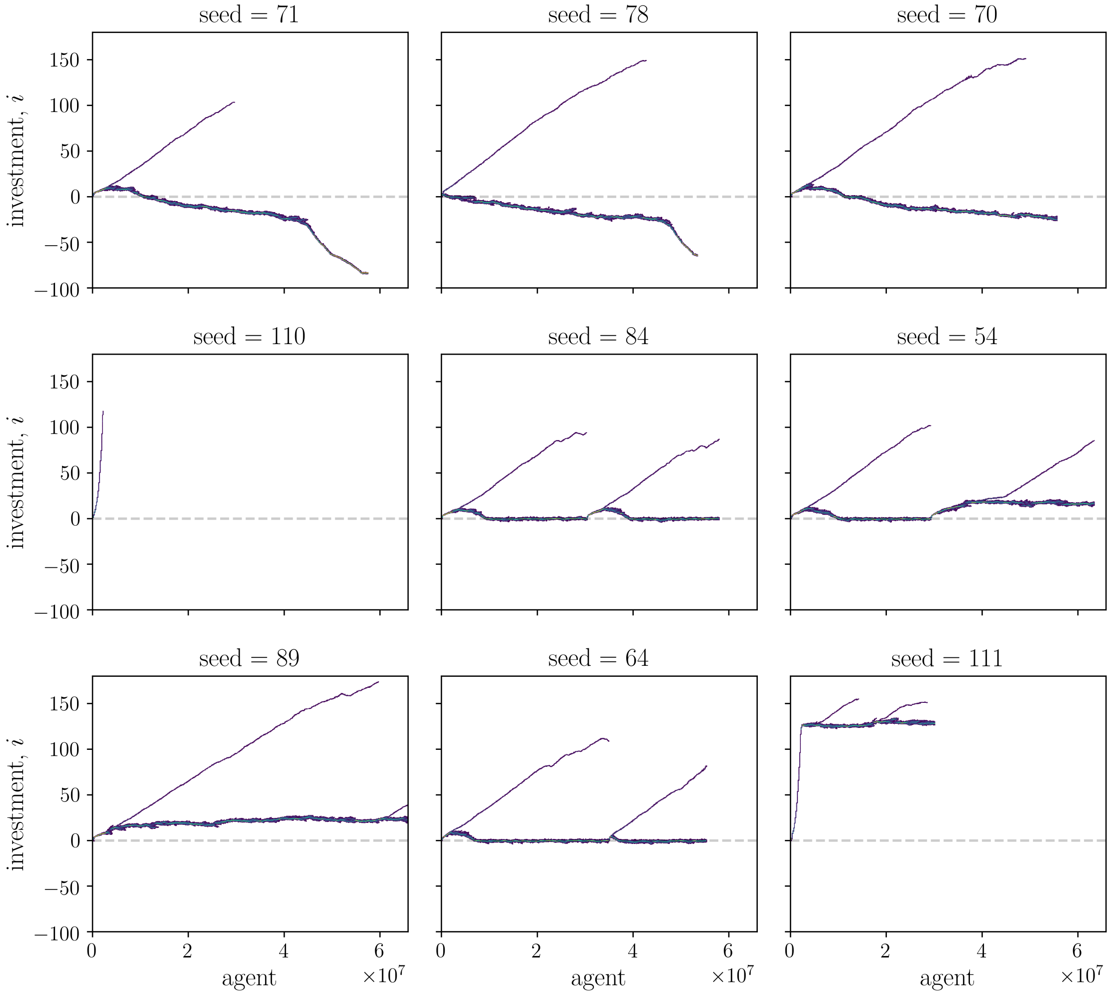


Figure A.10: Overview of simulation results for $r = 3, 0$ and 9 random number generator seeds showing trait evolution plots of the expected investment μ_i. A growingly defective branch emerges if the surviving population is defective after a branch collapse and a cooperative branch emerges if the surviving population is cooperative. The ordering corresponds from left to right and top to bottom to the lowest to highest final mean expected investment values. I.e., we can identify simulations in the corresponding violin in Figure 6.10 from bottom to top. The simulation ran for $T = 5 \times 10^5$ simulation steps. Seeds 71, 78, and 70 result in $\bar{\mu}_i^T < 0$; Seed 110 results in $\bar{\mu}_i^T < 0$ because the population dies out quickly; Seeds 84, 54, 89, 64, and 111 result in $\bar{\mu}_i^T > 0$ with the latter representing a positive outlier in the violins; We expect more variance for higher $\bar{\mu}_i$ value because a constant spread of p_s multiplied with high i will exhibit a broader itnerval of resulting values as for los i. Each plot has 900 bins in each dimension with counts higher than 2000 shown in yellow. Bin counts are capped showing values higher than 2×10^3 in yellow. See the caption in Figure 6.1 for the explanation of a trait evolution plot.

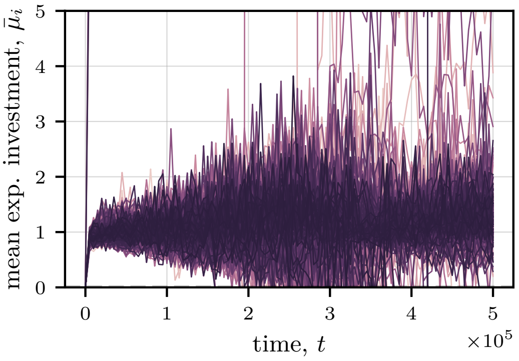


Figure A.11: This figure complements Figure 6.12, which shows the long-term evolution of the high synergy regime of $r = 3.0$ for 512 different seeds. It shows the moderately cooperative range, i.e., the positive value extension of the bottom left plot in Figure 6.12. Each line represents one universe run.

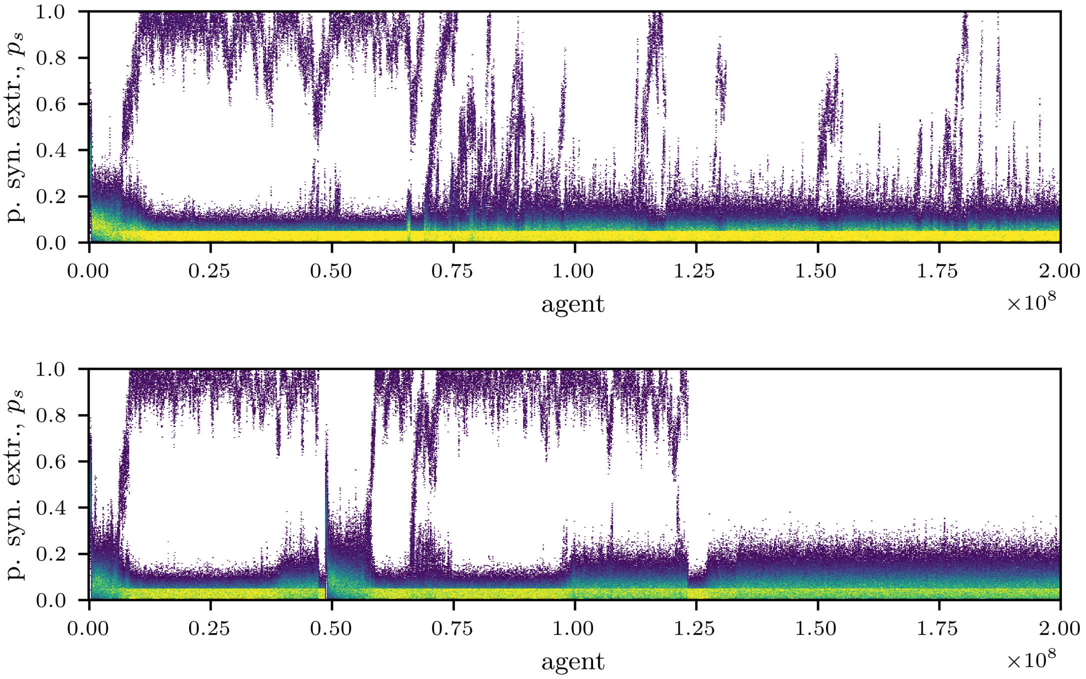


Figure A.12: The figure complements Figure 6.14, which shows the long-term evolution for very synergy factors of $r = 10$ by showing the trait evolution plots of the probability of extracting synergistic resources p_s for two example runs. The top exemplifies a system realization reaching the cooperative attractor and the bottom the defective attractor. Each plot has 1000(500) bins in x(y) direction, with counts higher than 400 shown in yellow. See the caption in Figure 6.1 for the explanation of a trait evolution plot.

B List of Figures & Tables

B.1 List of Figures

4.1 The *ReCooDy* process cycle. At the beginning of a simulation run, we initialize *ReCooDy* with a neutral, unconnected population and set up all relevant model parameters. Then, the actual process cycle starts. One cycle defines one model iteration and one time step. As an overview, I concisely summarize the model processes in the following, which I will, however, explain in detail within the corresponding sections of this chapter. First, agents either extract basic resources independently with probability p_b or, with probability $p_s = 1 - p_b$, synergistic resources through generalized public goods game (gPGG) interactions that exhibit social dilemma characteristics. Then, agents pay a constant cost of living c_l and a strength cost c_s. The evolving strength trait determines their success during basic resource extraction. They use their internal resources to link, i.e., they add or remove interaction connections to others to optimized their social environment. If their internally stored resources fall below a death threshold t_δ they die of exhaustion; They get removed from the population with all their links. Death also occurs randomly with probability p_δ per agent and time step. If their resources exceed a birth threshold t_β they create an offspring with probability p_β per agent and time step, pay a birth cost c_β and transfer resources j_β to their offspring. Afterward, the process cycle starts again with resource extraction. 38

4.2 Schematic effective interaction network of agents $a \in \mathcal{S}_t$ extracting synergistic resources at time t. The graph exemplifies the local effective social environment of one central agent (big blue). The central agent participates in $n_a^t = 4$ sub-interactions: one centered around itself (green $\boldsymbol{\alpha}$ area), and the other three centered around its respective neighbors (light green $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$, and $\boldsymbol{\delta}$ areas). Each agent a invests an equal share ι_a/n_a^t in all its sub-interaction partitioning the potential gain as well as their risk. Colors indicate the existence of cooperative ($\iota_a > 0$, blue) and defective ($\iota_a < 0$, red) strategies. The vertex size schematically encodes payoff differences resulting from the interactions. 44

4.3 Strategy classification for the standard public goods game (PGG) (left) and the generalized public goods game (gPGG) (right). In the PGG, agents are either cooperators (C) or defectors (D). Cooperators usually pay a fixed cost c to create public goods, from which all benefit. Some models include varying costs from a continuous space, but costs are always restricted to positive values. Defectors do not pay a cost but still profit from the generated goods. Thus, the standard PGG only implements public goods creation. The gPGG generalizes the standard PGG to incorporate public goods creation as well as destruction. Agents are either loners (L), cooperators (C), or defectors (D). Loners do not interact, i.e., they have $\mu_i \approx 0$ with $p_s \approx 0$. Cooperators create public goods through positive investments, and defectors destroy public goods through negative investments. A positive investment means paying a cost and a negative one actively grabbing resources. Thus, the gPGG extends the concept of defection by incorporating true defection—active destruction of goods for a personal benefit. All interacting agents benefit from the created public goods but have to pay for the destructed ones. We will further use a contextual classification: cooperators separate into benefactors (B) with high μ_i and profiteers (P) with low μ_i through a gap in μ_i, and defectors separate into exploiters (E) with low $|\mu_i|$ and malefactors (M) with high $|\mu_i|$ also through a gap. *Blue* represents cooperative strategies and *red* defective ones. More opaque colors represent a comparatively smaller magnitude of cooperative or defective behavior, respectively. 50

6.1 Trait evolution plots of the expected investment μ_i for varying synergy factors r that exemplify different dynamical regimes. A trait evolution plot is a two-dimensional histogram of a quantity shown on the y-axis and agents on the x-axis, in their order of birth, which allows visualizing the evolutionary quantity change with respect to fixed-sized agent generations. Thus, it shows the evolution of a quantity in trait space, which corresponds to a non-linear time evolution. A trait evolution plot directly shows the total number of agents that lived throughout a simulation in the maximum value on the agent-axis that can be used for comparison between simulations. It shows the color-encoded local agent and trait density, respectively, from low densities (purple) to high (yellow) ones. Bins with counts equal to zero are always shown as white bins. Each time evolution plot has 900 bins in both dimensions with bin counts higher than 300 shown as yellow bins. Each simulation ran for $T = 10^5$ steps. Figure A.1 complements this figure as it shows the evolution of i and p_s individually. 79

6.2 Time evolution of the mean expected investment $\bar{\mu}_i$ (left) and its final distribution $\bar{\mu}_i^T$ after $T = 5 \cdot 10^5$ time steps ($\approx 10^3$ generations) for varying synergy factor r (right) showing the emergence of cooperation regime. On the left, lines represent the mean and their shaded areas the standard deviation of $\bar{\mu}_i$ calculated over simulation runs. On the right, each of the 512 dots represents the final population-averaged expected investment $\bar{\mu}_i^T$ of a single system realization. Color encodes the synergy factor r from low values (yellow) to high ones (blue). The violin plots in the background show the distributions' kernel density estimates with width scaled to equal areas. The right figure shows a zoomed $\bar{\mu}_i^T$ value range. Figure A.2 shows the entire value range for completeness. 82

6.3 Simulation results for a minute synergy factor of $r = 1.01$ showing the slow but steady emergence of infrequent cooperation in sparse networks. It shows trait evolution plots of the investment trait i (top left) and the probability of extracting synergistic resources p_s (top right) with 900 bins in each dimension and bin counts higher than 100 for i and 20 for p_s shown in yellow. The middle left shows the phase space of the expected investment μ_i and the expected strength μ_s for the last 10^6 agents with dots representing single agents and color encoding time from early (yellow) until the simulation end (blue). The middle right shows a final largest connected component example for a population. The bottom left visualizes the final degree distribution, and the bottom right the final connected component size distribution over 128 simulation runs, each with color encoding different simulation runs. Blue lines show least-squares fit results of a power-law function. All simulations ran for $T = 10^6$ time steps for the single runs and $T = 2 \times 10^6$ for the distributions. See the caption in Figure 6.1 for the explanation of a trait evolution plot. 84

6.4 The transition regime. The top row shows the time evolution of the mean expected investment $\bar{\mu}_i$ (top left) and the number of agents N (top right) for different synergy factors r. The mean and standard deviation (shaded area) are calculated over system realizations. The bottom row shows the final mean expected investment $\bar{\mu}_i^T$ after $T = 5 \cdot 10^5$ time steps in the whole value range (bottom left) and zoomed (bottom right). Each dot represents the final population average of one system realization, 512 per r. Color encodes rising synergy factors from low (yellow) to high values (blue). The violin plots in the background show the kernel density estimates of the distributions with width scaled to equal areas. Lines within the violins show the median and quartiles of the corresponding distribution. Figure A.3 complements this figure by showing the time evolution of the number of agent and the trait evolution plots of $\bar{i}^T$ and $\bar{p}_s^T$. 87

6.5 Trait evolution plots of the expected investments μ_i that exemplify the rich dynamics of the transition regime. Simulation results for $r = 1.4$ and nine selected seeds representing typical dynamical phases are shown. Each plot contains 500 bins in each dimension with counts higher than 500 shown in yellow. Each simulation ran for $T = 10^5$ steps. See Figure A.4 for the entire value range and Figure A.5 for the corresponding time evolution of the number of agents and the final degree distribution. See the caption in Figure 6.1 for the explanation of a trait evolution plot. 89

6.6 Trait evolution plots of the most important quantities and the time evolution of the number of agents (top) that show examples of dynamical phases in the transition regime in detail. Circled letters represent dynamical phases referred to in the text. The simulation ran for $T = 5 \times 10^5$ steps. Figure A.6 complements this figure by showing further trait evolution plots. See the caption in Figure 6.1 for the explanation of a trait evolution plot. 94

6.7 Long-term evolution for a synergy factor $r = 1.4$. The top shows the trait evolution plot of the expected investment μ_i for a single example system realization run for $T = 10^7$ time steps. It has 1000(500) bins in the x(y) direction with counts higher than 10^3 shown in yellow. The middle and bottom rows show simulation results of 128 system realizations through varying seed. It shows the time evolution of the mean expected investment $\bar{\mu}_i$ in the entire value range (middle left) and zoomed (middle right). Each line represents one universe run. The bottom shows the final mean expected investment $\bar{\mu}_i^T$ distribution. The orange line marks $t = 5 \times 10^5$, the simulation time used previously (Figure 6.5). The red lines mark the transitions in the example runs. There are three regimes: ① Cooperative populations that ② eventually become hugely defective before ③ collapsing into a regime of moderate defection. All initially cooperative populations die out on the visible time-scale, but only part of the extremely defective populations collapse into moderate defective populations (bottom). These transitions are highly improbable events with a huge impact. No moderately defective population evolves cooperation anew. See the caption in Figure 6.1 for the explanation of a trait evolution plot. 101

6.8 Trait evolution plots of the most important quantities that show the transition into growing defection. It happens in two main steps (red lines) that both significantly alter agent traits and the population structure. The simulation ran for $T = 5 \times 10^5$ steps. Figure A.8 complements this figure by showing the mean link addition and removal costs and the agents' accumulated number of links. See the caption in Figure 6.1 for the explanation of a trait evolution plot. 103

6.9 Trait evolution plots of the most important quantities showing the final collapse of growing defection into moderate defection. It has 500 bins in each dimension, except for the bottom two rows with 100 bins in y-direction. The transition happens in two stages (red lines) that significantly alter the agents' traits. The entire simulation ran for $T = 10^7$ steps. It is the same as in Figure 6.7 (top). See the caption in Figure 6.1 for the explanation of a trait evolution plot. 107

6.10 Time evolution plot of the mean expected investment $\bar{\mu}_i$ (left) and its final distribution $\bar{\mu}_i^T$ at time $T = 10^5$ (right) for different synergy factors r showing the high synergy regime. On the left, lines represent the mean and their shaded areas the standard deviation of $\bar{\mu}_i$ calculated over simulation runs. On the right, each dot represents the final population average of a single system realization (512 per r). Color encodes rising synergy factors from low (yellow) to high values (blue). The violin plots in the background show the kernel density estimates of the distributions. The width of each violin is scaled according to equal areas. Lines within the violins show the median and quartiles of the corresponding distribution. The right plot shows a zoomed value range for $\bar{\mu}_i^T$. Figure A.9 shows the full value range with extreme outliers. Dead populations are shown as dots at $\bar{\mu}_i^T = 0$. . . 113

6.11 Trait evolution plots of the expected investment μ_i for $r = 3$ and varying random number generator seed that exemplify dynamical phases in the high synergy regime. Each plot has 900 bins in both dimensions with counts higher than 200 shown in yellow. The seed ordering follows lowest to highest final mean expected investment values from top to bottom and left to right. Each simulation ran for $T = 5 \times 10^5$ simulation steps. The populations with seeds 71, 78, and 70 show $\bar{\mu}_i^T < 0$, 110 shows $\bar{\mu}_i^T = 0$, and 84, 54, 89, 64, and 111 show $\bar{\mu}_i^T > 0$. Seed 111 is a positive outlier in the respective violin in Figure 6.10; Figure A.10 complements this figure by showing the trait evolution plot of the corresponding investment trait. See the caption in Figure 6.1 for the explanation of a trait evolution plot. . . 116

6.12 Long-term evolution of the high synergy regime. The time evolution of the mean expected investment $\bar{\mu}_i$ for synergy factor $r = 3$, simulation time $T = 2 \times 10^7$, and 512 different seeds in the entire value range (top left), zoomed into $\bar{\mu}_i$ (top right), and further zoomed into $\bar{\mu}_i$ and t (bottom left) together with its final distributions $\bar{\mu}_i^T$ (bottom right) are shown. Each line represents one system realization. The vertical orange line represents the zoomed time $t = 5 \times 10^5$. Single dots represent $\bar{\mu}_i^T$ of a single system realization with dead populations visualized as $\bar{\mu}_i^T = 0$. The violin in the background shows the kernel density estimate of the distribution. The plots show the long-term evolution of the high synergy regime moving from ① cooperative populations through ② a metastable population state of defection either back to a cooperative population or ③ to the final defective attractor state. 118

6.13 Example realizations showing the long-term evolution for $r = 3$ with a defective final attractor (top two rows) and continuously cooperative populations (bottom two rows) as shown in Figure 6.12. Trait evolution plots of the respective expected investments μ_i and the investment traits i are shown. Each plot has 1000(500) bins in the x(y) direction with counts higher than 200(2000) shown in yellow for $\mu_i(i)$. See the caption in Figure 6.1 for the explanation of a trait evolution plot. 119

6.14 Long-term evolution of the extreme high synergy regime of $r = 10$ for 128 different seeds showing ① the cooperative attractor and ② the defective attractor. The time evolution of the mean expected investment $\bar{\mu}_i$ (top left) and its final distribution $\bar{\mu}_i^T$ after $T = 2 \times 10^7$ time steps (top right) with are shown together with trait evolution plots of μ_i that exemplify the evolution towards the cooperative attractor (middle) and the defective one (bottom). On the top left, each line represents one system realization. On the right, each dot represents $\bar{\mu}_i^T$ of a single system realization. The violin in the background shows the kernel density estimate. The middle and bottom plots contain 1000(500) bins in the x(y) direction each, with counts higher than 50 shown in yellow. Figure A.12 complements this figure by showing the trait evolution plot of p_s. See the caption in Figure 6.1 for the explanation of a trait evolution plot. 122

A.1 Trait evolution plots of the investment i (top two rows) and the probability of extracting synergistic resources p_s in the short-term. This figure complements Figure 6.1 as it shows the evolution of the individual traits constituing the expected investment: $\mu_i = ip_s$. See the corresponding figure caption for the plotting details and an explanation of a trait evolution plot. 140

A.2 Final mean expected investment distribution $\bar{\mu}_i^T$ (left) and final mean investment distribution $\bar{i}^T$ after $T = 5 \cdot 10^5$ time steps ($\approx 10^3$ generations) in their full value range for varying synergy factor r showing the emergence of cooperation regime. It complements Figure 6.2. See there for figure details. 141

A.3 The transition regime. The top row shows the time evolution of the number of agents N for different synergy factors r. The mean and standard deviation (shaded area) are calculated over system realizations. The bottom row shows the final mean investments $\bar{i}^T$ (bottom left) in a zoomed value range and the final mean probability of extracting synergistic resources $\bar{p}_s^T$ (bottom right). All simulations ran for $T = 5 \cdot 10^5$ steps. Each dot represents the final population average of one system realization, 512 per r. Color encodes rising synergy factors from low (yellow) to high values (blue). The violin plots in the background show the kernel density estimates of the distribution with width scaled according to equal areas. Lines within the violins show the median and quartiles of the corresponding distribution. This figure uses the same data as presented in Figure 6.4 but plots additional relevant quantities. 141

A.4 Overview of simulation results for $r = 1, 4$ and 9 random number generator seeds. Trait evolution plots of the corresponding expected investment μ_i are shown in their full value range in contrast to a zoomed interval in Figure 6.5. See the caption in Figure 6.1 for the explanation of a trait evolution plot. 142

A.5 Overview of simulation results for $r = 1, 4$ and 9 random number generator seeds showing the number of agents (top), the degree distribution at $t = 2.8 \times 10^5$ (bottom left), and the final degree distribution at $t = 5 \times 10^5$. In combination with the expected investment evolution (see Figure 6.5) we observe the following: A highly dynamic cooperative phase (seed 44 at $t = 2.8 \times 10^5$) yields more than thrice the number of agents (peak in top plot) and a single high-degree hub agent (bottom left). Phases of rising defection show approximately twice the number of agents and a higher degree than typical dynamical phases with a cooperative branch. 143

A.6 Examples of transitions complementing Figure 6.6. Trait evolution plots of the investment trait i, the probability to extract synergistic resources p_s, and the accumulated link addition and removal costs are shown. The simulation ran for $T = 5 \times 10^5$ steps. See the caption in Figure 6.1 for the explanation of a trait evolution plot. 144

A.7 Examples of triggers into the rising defection phase for $r = 1.4$. Trait evolution plots of the expected investment μ_i are shown for a selection of 12 seeds that all exhibit a transitions into growing defection. Each plot has 500 bins in each dimension with counts higher than 200 shown in yellow. See the caption in Figure 6.1 for the explanation of a trait evolution plot. 145

A.8 Collapse of cooperation and transition towards growing defection happens in two main steps (red and orange line) that significantly alter agent traits and the population structure. This figure complements Figure 6.8. It shows the mean link addition and removal costs (top) averaged over an agent's lifetime and the total number of links an agent accumulates over its lifetime in trait evolution plots. The simulation ran for $T = 5 \times 10^5$ steps. See the caption in Figure 6.1 for the explanation of a trait evolution plot. 146

A.9 Full value range of the high synergy regime complementing Figure 6.10. The mean expected investment time evolution of the population averaged over 512 simulation runs each (left) and their final distribution at time $T = 10^5$ after approximately 500 generations (right) depending on the synergy factor r. For clarity, we only show a selection of r on the left. The shaded areas show the standard deviation over varying system realization. Each dot represents the final population average of a single system realization (512 per r). Color encodes rising synergy factors from low ($r = 2.0$; yellow) to high values ($r = 10.0$; blue). The violin plots in the background show the kernel density estimate of the distribution. The width of each violin is scaled according to equal areas. 147

A.10 Overview of simulation results for $r = 3,0$ and 9 random number generator seeds showing trait evolution plots of the expected investment μ_i. A growingly defective branch emerges if the surviving population is defective after a branch collapse and a cooperative branch emerges if the surviving population is cooperative. The ordering corresponds from left to right and top to bottom to the lowest to highest final mean expected investment values. I.e., we can identify simulations in the corresponding violin in Figure 6.10 from bottom to top. The simulation ran for $T = 5 \times 10^5$ simulation steps. Seeds 71, 78, and 70 result in $\bar{\mu}_i^T < 0$; Seed 110 results in $\bar{\mu}_i^T < 0$ because the population dies out quickly; Seeds 84, 54, 89, 64, and 111 result in $\bar{\mu}_i^T > 0$ with the latter representing a positive outlier in the violins; We expect more variance for higher $\bar{\mu}_i$ value because a constant spread of p_s multiplied with high i will exhibit a broader itnerval of resulting values as for los i. Each plot has 900 bins in each dimension with counts higher than 2000 shown in yellow. Bin counts are capped showing values higher than 2×10^3 in yellow. See the caption in Figure 6.1 for the explanation of a trait evolution plot. 148

A.11 This figure complements Figure 6.12, which shows the long-term evolution of the high synergy regime of $r = 3.0$ for 512 different seeds. It shows the moderately cooperative range, i.e., the positive value extension of the bottom left plot in Figure 6.12. Each line represents one universe run. . . 149

A.12 The figure complements Figure 6.14, which shows the long-term evolution for very synergy factors of $r = 10$ by showing the trait evolution plots of the probability of extracting synergistic resources p_s for two example runs. The top exemplifies a system realization reaching the cooperative attractor and the bottom the defective attractor. Each plot has 1000(500) bins in x(y) direction, with counts higher than 400 shown in yellow. See the caption in Figure 6.1 for the explanation of a trait evolution plot. 149

B.2 List of Tables

4.1 Evolving parameters, their respective mutation sampling distributions, and value spaces. Distributions with curly brackets indicate discrete uniform distributions while square brackets denote continuous uniform ones. . . . 58

4.2 Initialized Parameters. For a detailed description of all parameters, look up the symbol in the corresponding Appendix list. 60

Own Publications

I contributed as coauthor in the following publications. Their content constitutes the basis of chapter (chapter 3), in which I present the *Utopia* modeling framework.

- *Riedel*, L. *and Herdeanu*, B. *and Mack*, H. *and Sevinchan*, Y., & *Weninger*, J. (2020). Utopia: A comprehensive and collaborative modeling framework for complex and evolving systems. *JOSS*, *5*(53), 2165. https://doi.org/10.21105/joss.02165

- *Sevinchan*, Y. *and Herdeanu*, B. *and Mack*, H. *and Riedel*, L., & *Roth*, K. (2020a). Boosting group-level synergies by using a shared modeling framework (V. V. *Krzhizhanovskaya and* G. *Závodszky and* M. H. *Lees and* J. J. *Dongarra and* P. M. A. *Sloot and* S. *Brissos*, & J. *Teixeira*, Eds.). In V. V. *Krzhizhanovskaya and* G. *Závodszky and* M. H. *Lees and* J. J. *Dongarra and* P. M. A. *Sloot and* S. *Brissos*, & J. *Teixeira* (Eds.), *Computational Science – ICCS 2020*, Cham, Switzerland, Springer International Publishing. https://doi.org/10.1007/978-3-030-50436-6_32

- *Sevinchan*, Y. *and Herdeanu*, B., & *Traub*, J. (2020b). dantro: A Python package for handling, transforming, and visualizing hierarchically structured data. *JOSS*, *5*(52), 2316. https://doi.org/10.21105/joss.02316

Bibliography

Abbot, P., *Abe*, J., *Alcock*, J., *Alizon*, S., *Alpedrinha*, J. A., *Andersson*, M., *Andre*, J. B., *Van Baalen*, M., *Balloux*, F., *Balshine*, S., *Barton*, N., *Beukeboom*, L. W., *Biernaskie*, J. M., *Bilde*, T., *Borgia*, G., *Breed*, M., *Brown*, S., *Bshary*, R., *Buckling*, A., ... *Zink*, A. (2011). Inclusive fitness theory and eusociality. *Nature*, *471*(7339), 1057–1062. https://doi.org/10.1038/nature09831

Adami, C., *Schossau*, J., & *Hintze*, A. (2016). Evolutionary game theory using agent-based methods. *Phys. Life Rev.*, *19*, 1–26. https://doi.org/10.1016/j.plrev.2016.08.015

Adams, A., *Zenil*, H., *Davies*, P. C., & *Walker*, S. I. (2017). Formal Definitions of Unbounded Evolution and Innovation Reveal Universal Mechanisms for Open-Ended Evolution in Dynamical Systems. *Sci. Rep.*, *7*(1), 1–15.

Akçay, E. (2018). Collapse and rescue of cooperation in evolving dynamic networks. *Nat. Commun.*, *9*(1), 1–9. https://doi.org/10.1038/s41467-018-05130-7

Akçay, E. (2020). Deconstructing evolutionary game theory: Coevolution of social behaviors with their evolutionary setting. *Am. Nat.*, *195*(2), 315–330. https://doi.org/10.1086/706811

Aktipis, A. (2016). Principles of cooperation across systems: From human sharing to multicellularity and cancer. *Evol. Appl.*, *9*(1), 17–36. https://doi.org/10.1111/eva.12303

Albert, R., & *Barabási*, A.-L. (2002). Statistical mechanics of complex networks. *Rev. Mod. Phys.*, *74*(1), 47–97. https://doi.org/10.1103/RevModPhys.74.47

Allen, B., *Lippner*, G., *Chen*, Y.-T., *Fotouhi*, B., *Momeni*, N., *Yau*, S.-T., & *Nowak*, M. A. (2017). Evolutionary dynamics on any population structure. *Nature*, *544*(7649), 227–230. https://doi.org/10.1038/nature21723

Allen, B., & *Nowak*, M. A. (2016). There is no inclusive fitness at the level of the individual. *Curr. Opin. Behav. Sci.*, *12*, 122–128. https://doi.org/10.1016/j.cobeha.2016.10.002

Assenza, S., *Gómez-Gardeñes*, J., & *Latora*, V. (2008). Enhancement of cooperation in highly clustered scale-free networks. *Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.*, *78*(1), 22–25. https://doi.org/10.1103/PhysRevE.78.017101

Axelrod, R., *Axelrod*, D. E., & *Pienta*, K. J. (2006). Evolution of cooperation among tumor cells. *PNAS*, *103*(36), 13474–9. https://doi.org/10.1073/pnas.0606053103

Axelrod, R., & *Hamilton*, W. D. (1981). The evolution of cooperation. *Evolution*, *211*(4489), 1390–1396. https://doi.org/10.1086/383541

Bak, P., *Chen*, K., & *Tang*, C. (1990). A forest-fire model and some thoughts on turbulence. *Phys. Lett. A*, *147*(5-6), 297–300. https://doi.org/10.1016/0375-9601(90)90451-S

Bak, P., *Tang*, C., & *Wiesenfeld*, K. (1987). Self-organized criticality: An explanation of 1/f noise. *Phys. Rev. Lett.*, *59*(4), 381–384.

Baldwin, I. T., & *Schultz*, J. C. (1983). Rapid changes in tree leaf chemistry induced by damage: evidence for communication between plants. *Science*, *221*(4607), 277–279. https://doi.org/10.1126/science.221.4607.277

Barabási, A.-L., & *Albert*, R. (1999). Emergence of scaling in random networks. *Science*, *286*(October), 509–513. https://doi.org/10.1126/science.286.5439.509

Barabási, A.-L., & *Pósfai*, M. (2016). *Network Science*. Cambridge, Cambridge University Press.

Barfuss, W., *Donges*, J. F., *Vasconcelos*, V. V., *Kurths*, J., & *Levin*, S. A. (2020). Caring for the future can turn tragedy into comedy for long-term collective action under risk of collapse. *PNAS*, *117*(23), 12915–12922. https://doi.org/10.1073/pnas.1916545117

Battiston, F., *Cencetti*, G., *Iacopini*, I., *Latora*, V., *Lucas*, M., *Patania*, A., *Young*, J. G., & *Petri*, G. (2020). Networks beyond pairwise interactions: Structure and dynamics. *Phys. Rep.*, *874*(June), 1–92. https://doi.org/10.1016/j.physrep.2020.05.004

Ben-Kiki, O., *Evans*, C., & döt. *Net*, I. (2009). YAML ain't markup language. https://yaml.org/spec/1.2/spec.html

Birch, J. (2017). Kin selection, group selection, and the varieties of population structure. *Br. J. Philos. Sci.*, *0*(July), 1–30. https://doi.org/10.1093/bjps/axx028

Birch, J., & *Okasha*, S. (2015). Kin selection and its critics. *Bioscience*, *65*(1), 22–32. https://doi.org/10.1093/biosci/biu196

Blount, Z. D., *Barrick*, J. E., *Davidson*, C. J., & *Lenski*, R. E. (2012). Genomic analysis of a key innovation in an experimental Escherichia coli population. *Nature*, *489*(7417), 513–518. https://doi.org/10.1038/nature11514

Boccaletti, S., *Bianconi*, G., *Criado*, R., del *Genio*, C. I., *Gómez-Gardeñes*, J., *Romance*, M., *Sendiña-Nadal*, I., *Wang*, Z., & *Zanin*, M. (2014). The structure and dynamics of multilayer networks. *Phys. Rep.*, *544*(1), 1–122. https://doi.org/10.1016/j.physrep.2014.07.001

Boccaletti, S., *Latora*, V., *Moreno*, Y., *Chavez*, M., & *Hwang*, D. U. (2006). Complex networks: Structure and dynamics. *Phys. Rep.*, *424*(4-5), 175–308. https://doi.org/10.1016/j.physrep.2005.10.009

Boguñá, M., *Bonamassa*, I., *De Domenico*, M., *Havlin*, S., *Krioukov*, D., & *Serrano*, M. Á. (2021). Network geometry. *Nat. Rev. Phys.*, *3*(2), 114–135. https://doi.org/10.1038/s42254-020-00264-4

Bourke, A. F. G. (2011). *Principles of social evolution*. Oxford University Press, https://doi.org/10.1093/acprof:oso/9780199231157.001.0001

Boyd, R., & *Richerson*, P. J. (1982). Cultural transmission and the evolution of cooperative behavior. *Hum. Ecol.*, *10*(3), 325–351. https://doi.org/10.1007/BF01531189

Boyd, R., & *Richerson*, P. J. (2005). *The origin and evolution of cultures*. Oxford University Press. https://doi.org/10.1525/aa.2006.108.3.547

Bozic, I., & *Nowak*, M. A. (2013). Unwanted evolution. *Science*, *342*(6161), 938–939. https://doi.org/10.1126/science.1247887

Brown, J. S. (2016). Why Darwin would have loved evolutionary game theory. *Proc. R. Soc. B Biol. Sci.*, *283*, 20160847. https://doi.org/10.1098/rspb.2016.0847

Cardinot, M., *O'Riordan*, C., *Griffith*, J., & *Perc*, M. (2019). Evoplex: A platform for agent-based modeling on networks. *SoftwareX*, *9*, 199–204. https://doi.org/10.1016/j.softx.2019.02.009

Carroll, L. (1871). *Through the Looking-Glass, and What Alice Found There*. Macmillan.

Carter, G. (2014). The reciprocity controversy. *Anim. Behav. Cogn.*, *1*(3), 368. https://doi.org/10.12966/abc.08.11.2014

Carter, G. G., & *Wilkinson*, G. S. (2015). Social benefits of non-kin food sharing by female vampire bats. *Proc. R. Soc. B Biol. Sci.*, *282*(1819). https://doi.org/10.1098/rspb.2015.2524

Cassar, A. (2007). Coordination and cooperation in local, random and small world networks: Experimental evidence. *Games Econ. Behav.*, *58*(2), 209–230. https://doi.org/10.1016/j.geb.2006.03.008

Cavaliere, M., *Sedwards*, S., *Tarnita*, C. E., *Nowak*, M. A., & *Csikász-Nagy*, A. (2012). Prosperity is associated with instability in dynamical networks. *J. Theor. Biol.*, *299*, 126–138. https://doi.org/10.1016/j.jtbi.2011.09.005

Cavicchioli, R., *Ripple*, W. J., *Timmis*, K. N., *Azam*, F., *Bakken*, L. R., *Baylis*, M., *Behrenfeld*, M. J., *Boetius*, A., *Boyd*, P. W., *Classen*, A. T., *Crowther*, T. W., *Danovaro*, R., *Foreman*, C. M., *Huisman*, J., *Hutchins*, D. A., *Jansson*, J. K., *Karl*, D. M., *Koskella*, B., *Mark Welch*, D. B., ... *Webster*, N. S. (2019). Scientists' warning to humanity: Microorganisms and climate change. *Nat. Rev. Microbiol.*, *17*(9), 569–586. https://doi.org/10.1038/s41579-019-0222-5

Chopard, B., *Dupuis*, A., *Masselot*, A., & *Luthi*, P. (2002). Cellular automata and lattice boltzmann techniques: An approach to model and simulate complex systems. *Adv. Complex Syst.*, *05*(02n03), 103–246. https://doi.org/10.1142/s0219525902000602

Clutton-Brock, T. H., & *Parker*, G. A. (1995). Punishment in animal societies. *Nature*, *373*(19), 209–201. https://doi.org/10.1038/373209a0

Collette, A. (2013). *Python and HDF5*. O'Reilly Media.

Cordero, O. X., *Wildschutte*, H., *Kirkup*, B., *Proehl*, S., *Ngo*, L., *Hussain*, F., *Le Roux*, F., *Mincer*, T., *Polz*, M. F., & *Roux*, F. L. (2012). Ecological populations of bacteria act as social cohesive units of antibiotic production and resistance. *Science*, *337*(September), 1228–1231. https://doi.org/10.1126/science.1219385

Dantro Developer Team. (2021). Dantro documentation. Retrieved August 19, 2021, from https://dantro.readthedocs.io/

Darwin, C. (1859). *On the origin of species by means of natural selection, or the preservation of favoured races in the struggle for life* (1st ed.). London, John Murray.

Dask Development Team. (2016). Dask: Library for dynamic task scheduling.

Dawkins, R. (1976). *The selfish gene*. Oxford University Press.

Deffuant, G., *Neau*, D., *Amblard*, F., & *Weisbuch*, G. (2000). Mixing belief among interacting agents. *Adv. Complex Syst.*, (3), 87–98. https://doi.org/10.1142/s0219525900000078

Diggle, S. P., *Griffin*, A. S., *Campbell*, G. S., & *West*, S. A. (2007). Cooperation and conflict in quorum-sensing bacterial populations. *Nature*, *450*(7168), 411–414. https://doi.org/10.1038/nature06279

Dodd, M. S., *Papineau*, D., *Grenne*, T., *Slack*, J. F., *Rittner*, M., *Pirajno*, F., *O'Neil*, J., & *Little*, C. T. (2017). Evidence for early life in Earth's oldest hydrothermal vent precipitates. *Nature*, *543*(7643), 60–64. https://doi.org/10.1038/nature21377

Doebeli, M., & *Hauert*, C. (2005). Models of cooperation based on the Prisoner's Dilemma and the Snowdrift game. *Ecol. Lett.*, *8*(7), 748–766. https://doi.org/10.1111/j.1461-0248.2005.00773.x

Doebeli, M., *Hauert*, C., & *Killingback*, T. (2004). The evolutionary origin of cooperators and defectors. *Science*, *2*(10), 1–5.

Doebeli, M., *Ispolatov*, Y., & *Simon*, B. (2017). Towards a mechanistic foundation of evolutionary theory. *Elife*, *6*, 1–17. https://doi.org/10.7554/eLife.23804.001

Doebeli, M., & *Knowlton*, N. (1998). The evolution of interspecific mutualisms. *PNAS*, *95*(15), 8676–8680. https://doi.org/10.1073/pnas.95.15.8676

Dorogovtsev, S. N., *Goltsev*, A. V., & *Mendes*, J. F. (2008). Critical phenomena in complex networks. *Rev. Mod. Phys.*, *80*(4), 1275–1335. https://doi.org/10.1103/RevModPhys.80.1275

Drossel, B., & *Schwabl*, F. (1992). Self-organized criticality in a forest-fire model. *Phys. A Stat. Mech. its Appl.*, *191*(1-4), 47–50. https://doi.org/10.1016/0378-4371(92)90504-J

Eguíluz, V. M., *Zimmermann*, M. G., *Cela-Conde*, C. J., & *Miguel*, M. S. (2005). Cooperation and the emergence of role differentiation in the dynamics of social networks. *Am. J. Sociol.*, *110*(4), 977–1008.

Erdös, P., & *Rényi*, A. (1959). On random graphs I. *Publ. Math.*, *6*, 290–297.

Farmer, E. E., & *Ryan*, C. A. (1990). Interplant communication: Airborne methyl jasmonate induces synthesis of proteinase inhibitors in plant leaves. *PNAS*, *87*(19), 7713–7716. https://doi.org/10.1073/pnas.87.19.7713

Fehr, E., & *Fischbacher*, U. (2003). The nature of human altruism. *Nature*, *425*(6960), 785–791. https://doi.org/10.1038/nature02043

Fehr, E., & *Gächter*, S. (2002). Altruistic punishment in humans. *Nature*, *415*(6868), 137–140. https://doi.org/10.1038/415137a

Foster, K. R., *Schluter*, J., *Coyte*, K. Z., & *Rakoff-Nahoum*, S. (2017). The evolution of the host microbiome as an ecosystem on a leash. *Nature*, *548*(7665), 43–51. https://doi.org/10.1038/nature23292

Fowler, J. H. (2005). Altruistic punishment and the origin of cooperation. *PNAS*, *102*(19), 7047–9. https://doi.org/10.1073/pnas.0500938102

Fowler, J. H., & *Christakis*, N. A. (2009). Cooperative behavior cascades in human social networks. *PNAS*. https://doi.org/10.1073/pnas.0913149107

Frank, S. A. (2007). All of life is social. *Curr. Biol.*, *17*(16), 648–650. https://doi.org/10.1016/j.cub.2007.06.005

Friedman, D., & *Barry*, S. (2016). *Evolutionary games in natural, social, and virtual worlds*. Oxford, Oxford University Press.

García, J., & *Traulsen*, A. (2019). Evolution of coordinated punishment to enforce cooperation from an unbiased strategy space. *J. R. Soc. Interface*, *16*(156). https://doi.org/10.1098/rsif.2019.0127

Gardner, A. (2015). More on the genetical theory of multilevel selection. *J. Evol. Biol.*, *28*(9), 1747–1751. https://doi.org/10.1111/jeb.12684

Gardner, A. (2020). Price s equation made clear. *Philos. Trans. R. Soc. B Biol. Sci.*, *375*(1797). https://doi.org/10.1098/rstb.2019.0361

Gintis, H. (2009). *Game Theory Evolving* (2nd ed.). Princeton university press. https://doi.org/10.1515/9781400830077

Gintis, H., & *Helbing*, D. (2015). Homo Socialis: An analytical core for sociological theory. *Rev. Behav. Econ.*, *2*(1-2), 1–59. https://doi.org/10.1561/105.00000016

Gokhale, C. S., & *Hauert*, C. (2016). Eco-evolutionary dynamics of social dilemmas. *Theor. Popul. Biol.*, *111*, 28–42. https://doi.org/10.1016/j.tpb.2016.05.005

Goldenfeld, N., & *Kadanoff*, L. P. (1999). Simple lessons from complexity. *Science*, *284*(5411), 87–89. https://doi.org/10.1126/science.284.5411.87

Goldenfeld, N., & *Woese*, C. (2011). Life is physics: Evolution as a collective phenomenon far from equilibrium. *Annu. Rev. Condens. Matter Phys.*, *2*(1), 375–399. https://doi.org/10.1146/annurev-conmatphys-062910-140509

Gómez-Gardeñes, J., *Campillo*, M., *Floría*, L. M., & *Moreno*, Y. (2007). Dynamical organization of cooperation in complex topologies. *Phys. Rev. Lett.*, *98*(10), 1–4. https://doi.org/10.1103/PhysRevLett.98.108103

Gracia-Lázaro, C., *Ferrer*, A., *Ruiz*, G., *Tarancón*, A., *Cuesta*, J. A., *Sánchez*, A., & *Moreno*, Y. (2012). Heterogeneous networks do not promote cooperation when humans play a prisoner's dilemma. *PNAS*, *109*(32), 12922–12926. https://doi.org/10.1073/pnas.1206681109

Gross, T., & *Blasius*, B. (2008). Adaptive coevolutionary networks: A review. *J. R. Soc. Interface*, *5*(20), 259–71. https://doi.org/10.1098/rsif.2007.1229

Grujíc, J., *Fosco*, C., *Araujo*, L., *Cuesta*, J. A., & *Sánchez*, A. (2010). Social experiments in the mesoscale: Humans playing a spatial prisoner's dilemma. *PLoS One*, *5*(11). https://doi.org/10.1371/journal.pone.0013749

Hamilton, W. D. (1964). The genetical evolution of social behaviour. I. *J. Theor. Biol.*, *7*(1), 1–16. https://doi.org/10.1016/0022-5193(64)90038-4

Hammoud, Z., & *Kramer*, F. (2020). Multilayer networks: aspects, implementations, and application in biomedicine. *Big Data Analytics*, *5*(1), 1–18. https://doi.org/10.1186/s41044-020-00046-0

Hardin, G. (1968). The tragedy of the commons. *Science*, *162*(June), 1243–1248. https://doi.org/10.1126/science.162.3859.1243

Hauert, C., *De Monte*, S., *Hofbauer*, J., & *Sigmund*, K. (2002). Volunteering as Red Queen mechanism for cooperation in public goods games. *Science*, *296*(5570), 1129–1132. https://doi.org/10.1126/science.1070582

Hauert, C., & *Doebeli*, M. (2004). Spatial structure often inhibits the evolution of coopertion in public goods games. *Science*, *296*, 1129–1132. https://doi.org/10.1038/nature02422.1.

Hauert, C., *Holmes*, M., & *Doebeli*, M. (2006). Evolutionary games and population dynamics: Maintenance of cooperation in public goods games. *Proc. R. Soc. B Biol. Sci.*, *273*(1605), 3131–3132. https://doi.org/10.1098/rspb.2006.3717

Hauert, C., *Saade*, C., & *McAvoy*, A. (2019). Asymmetric evolutionary games with environmental feedback. *J. Theor. Biol.*, *462*, 347–360. https://doi.org/10.1016/j.jtbi.2018.11.019

Hauert, C., & *Szabó*, G. (2003). Prisoner's dilemma and public goods games in different geometries: Compulsory versus voluntary interactions. *Complexity*, *8*(4 Spec. Iss.), 31–38. https://doi.org/10.1002/cplx.10092

Hauert, C., *Traulsen*, A., *Brandt*, H., *Nowak*, M. A., & *Sigmund*, K. (2007). Via freedom to coercion: The emergence of costly punishment. *Science*, *316*(5833), 1905–1907. https://doi.org/10.1126/science.1141588

Hegselmann, R., & *Krause*, U. (2002). Opinion dynamics and bounded confidence: Models, analysis and simulation. *Jasss*, *5*(3).

Helbing, D. (2013). Economics 2.0: The natural step towards a self-regulating, participatory market society. *SSRN Electron. J.*, *10*(1), 3–41. https://doi.org/10.2139/ssrn.2267697

Helbing, D. (2015). Homo Socialis: The road ahead. *Rev. Behav. Econ.*, *2*(1-2), 239–253. https://doi.org/10.1561/105.00000032

Hofbauer, J., & *Sigmund*, K. (1998). *Evolutionary Games and Population Dynamics.* Cambridge University Press. https://doi.org/10.1017/cbo9781139173179

Holland, J. H. (2006). Studying complex adaptive systems. *J. Syst. Sci. Complex.*, *19*(1), 1–8. https://doi.org/10.1007/s11424-006-0001-z

Hölldobler, B., & *Wilson*, E. O. (1990). *The ants.* Harvard University Press.

Hölldobler, B., & *Wilson*, E. O. (2010). *The leafcutter ants: Civilization by instinct.* WW Norton & Company.

Holme, P. (2019). Rare and everywhere: Perspectives on scale-free networks. *Nat. Commun.*, *10*(1), 8–10. https://doi.org/10.1038/s41467-019-09038-8

Holme, P., & *Ghoshal*, G. (2006). Dynamics of networking agents competing for high centrality and low degree. *Phys. Rev. Lett.*, *96*(9), 1–4. https://doi.org/10.1103/PhysRevLett.96.098701

Holovatch, Y., *Kenna*, R., & *Thurner*, S. (2017). Complex systems: Physics beyond physics. *Eur. J. Phys.*, *38*(2). https://doi.org/10.1088/1361-6404/aa5a87

Hoyer, S., & *Hamman*, J. J. (2017). xarray: N-D labeled Arrays and Datasets in Python. *J. Open Res. Softw.*, *5*, 1–6. https://doi.org/10.5334/jors.148

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Comput. Sci. Eng.*, *9*(3), 90–95. https://doi.org/:10.1109/MCSE.2007.55

Ibsen-Jensen, R., *Chatterjee*, K., & *Nowak*, M. A. (2015). Computational complexity of ecological and evolutionary spatial dynamics. *PNAS*, 201511366. https://doi.org/10.1073/pnas.1511366112

Ilany, A., & *Akçay*, E. (2016). Social inheritance can explain the structure of animal social networks. *Nat. Commun.*, *7*(May). https://doi.org/10.1038/ncomms12084

IPCC. (2021). *Climate Change 2021: The Physical Science Basis. Contribution of Working Group I to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change. Cambridge University Press. In Press.* (V. *Masson-Delmotte*, P. *Zhai*, A. *Pirani*, S. L. *Connors*, C. *Péan*, S. *Berger*, N. *Caud*, Y. *Chen*, L. *Goldfarb*, M. I. *Gomis*, M. *Huang*, K. *Leitzell*, E. *Lonnoy*, J. B. R. *Matthews*, T. K. *Maycock*, T. *Waterfield*, O. *Yelekçi*, R. *Yu*, & B. *Zhou*, Eds.).

Jackson, M. O. (2008). *Social and economic networks.* Princeton university press. https://doi.org/doi:10.2307/j.ctvcm4gh1

Jackson, M. O., & *Rogers*, B. W. (2007). Meeting strangers and friends of friends: How random are social networks? *Am. Econ. Rev.*, *97*(3), 890–915. https://doi.org/10.1257/aer.97.3.890

Jackson, M. O., *Rogers*, B. W., & *Zenou*, Y. (2015). The economic consequences of social network structure. *SSRN Electron. J.*, (February), 1–65. https://doi.org/10.2139/ssrn.2467812

Jackson, M. O., & *Zenou*, Y. (2015). *Games on Networks* (Vol. 4). Elsevier Inc. https://doi.org/10.1016/B978-0-444-53766-9.00003-3

Jacob, F. (1977). Evolution and Tinkering. *Science*, *196*(4295), 1161–1166.

Kanewala, U., & *Bieman*, J. M. (2014). Testing scientific software: A systematic literature review. *Inf. Softw. Technol.*, *56*(10), 1219–1232. https://doi.org/10.1016/j.infsof.2014.05.006

Kauffman, S. A. (1993). *The origins of order: Self-organization and selection in evolution.* Oxford University Press, USA.

Killingback, T., *Doebeli*, M., & *Knowlton*, N. (1999). Variable investment, the Continuous Prisoner's Dilemma, and the origin of cooperation. *Proc. R. Soc. London B*, *266*(1430), 1723–1728. https://doi.org/10.1098/rspb.1999.0838

Kirchkamp, O., & *Nagel*, R. (2007). Naive learning and cooperation in network experiments. *Games Econ. Behav.*, *58*(2), 269–292. https://doi.org/10.1016/j.geb.2006.04.002

Kivelä, M., *Arenas*, A., *Barthelemy*, M., *Gleeson*, J. P., *Moreno*, Y., & *Porter*, M. A. (2014). Multilayer networks. *J. Complex Networks*, *2*(3), 203–271. https://doi.org/10.1093/comnet/cnu016

Lande, R. (1988). Genetics and demography in biological conservation. *Science*, *241*(4872), 1455–1460. https://doi.org/10.1126/science.3420403

Lenski, R. E. (2017). Experimental evolution and the dynamics of adaptation and genome evolution in microbial populations. *ISME J.*, *11*(10), 2181–2194. https://doi.org/10.1038/ismej.2017.69

Leonard, A. S., & *Francis*, J. S. (2017). Plant–animal communication: past, present and future. *Evol. Ecol.*, *31*(2), 143–151. https://doi.org/10.1007/s10682-017-9884-5

Levin, S. A. (2003). Complex adaptive systems: Exploring the known, the unknown and the unknowable. *Bull. Am. Math. Soc.*, *40*(1), 3–19. https://doi.org/10.1090/S0273-0979-02-00965-5

Lewin-Epstein, O., *Aharonov*, R., & *Hadany*, L. (2017). Microbes can help explain the evolution of host altruism. *Nat. Commun.*, *8*, 1–7. https://doi.org/10.1038/ncomms14040

Lewontin, R. C. (1961). Evolution and the theory of games. *J. Theor. Biol.*, *1*, 382–403. https://doi.org/10.1016/0022-5193(61)90038-8

Lieberman, E., *Hauert*, C., & *Nowak*, M. A. (2005). Evolutionary dynamics on graphs. *Nature*, *1*(January), 1–5. https://doi.org/10.1038/nature03211.1.

Lloyd, W. F. (1833). *Two lectures on the checks to population.* JH Parker.

Lorenz, E. N. (1963). Deterministic nonperiodic flow. *J. Atmos. Sci.*, *20*, 130–141. https://doi.org/10.1175/1520-0469(1963)020<0130:DNF>2.0.CO;2

Macal, C. M. (2016). Everything you need to know about agent-based modelling and simulation. *J. Simul.*, *10*(2), 144–156. https://doi.org/10.1057/jos.2016.7

Masad, D., & *Kazil*, J. (2015). Mesa : An agent-based modeling framework. *Proc. 14th Python Sci. Conf.*, (Scipy), 51–58.

Maynard Smith, J. (1972). Game theory and the evolution of fighting. *Evol.*, 8–28.

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

Maynard Smith, J., & *Price*, G. R. (1973). The logic of animal conflict. *Nature*, *246*(5427), 15–18. https://doi.org/10.1038/246015a0

Maynard Smith, J., & *Szathmáry*, E. (1995). *The Major Transitions in Evolution.* https://doi.org/10.1017/CBO9781107415324.004

Maynard Smith, J., & *Szathmáry*, E. (1999). *The Origins of Life.* Oxford University Press.

McFall-Ngai, M. (2014). Divining the essence of symbiosis: Insights from the squid-vibrio model. *PLoS Biol.*, *12*(2), 1–6. https://doi.org/10.1371/journal.pbio.1001783

McGill, B. J., & *Brown*, J. S. (2007). Evolutionary game theory and adaptive dynamics of continuous traits. *Annu. Rev. Ecol. Evol. Syst.*, *38*, 403–435. https://doi.org/10.1146/annurev.ecolsys.36.091704.175517

McNamara, J. M. (2013). Towards a richer evolutionary game theory. *J. R. Soc. Interface.*

McNamara, J. M., *Barta*, Z., *Fromhage*, L., & *Houston*, A. I. (2008). The coevolution of choosiness and cooperation. *Nature*, *451*(7175), 189–192. https://doi.org/10.1038/nature06455

Mcnickle, G. G., & *Dybzinski*, R. (2013). Game theory and plant ecology. *Ecol. Lett.*, *16*(4), 545–555. https://doi.org/10.1111/ele.12071

Michor, F., *Iwasa*, Y., & *Nowak*, M. A. (2004). Dynamics of cancer progression. *Nat. Rev. Cancer*, *4*(3), 197–205. https://doi.org/10.1038/nrc1295

Michor, F., & *Nowak*, M. A. (2002). Evolution: The good, the bad and the lonely. *Nature*, *419*(6908), 677–679. https://doi.org/10.1038/419677a

Murugan, A., *Husain*, K., *Rust*, M., *Bass*, J. T., *Hepler*, C., *Pietsch*, J. M., *Swain*, P. S., *Toettcher*, J. E., *Jena*, S., *Chakraborty*, A. K., *Sprenger*, K. G., *Mora*, T., *Walczak*, A., *Rivoire*, O., *Wang*, S., *Wood*, K., *Kussell*, E., *Skanata*, A., *Ranganathan*, R., ... *Goldenfeld*, N. (2021). Roadmap on biology in time varying environments. *Phys. Biol.* https://doi.org/10.1088/1478-3975/abde8d

Newton, J. (2018). Evolutionary game theory: A renaissance. *Games*, *9*(2). https://doi.org/10.3390/g9020031

Nowak, M. A., & *Sigmund*, K. (1998). Evolution of indirect reciprocity by image scoring. *Nature*, *393*(6685), 573–577. https://doi.org/10.1038/31225

Nowak, M. A. (2006a). *Evolutionary Dynamics*. Harvard University Press. https://doi.org/10.2307/j.ctvjghw98

Nowak, M. A. (2006b). Five rules for the evolution of cooperation. *Science*, *314*(5805), 1560–1563. https://doi.org/10.1126/science.1133755

Nowak, M. A., & *May*, R. M. (1992). Evolutionary games and spatial chaos. https://doi.org/10.1038/359826a0

Nowak, M. A., *McAvoy*, A., *Allen*, B., & *Wilson*, E. O. (2017). The general form of Hamilton's rule makes no predictions and cannot be tested empirically. *PNAS*, *114*(22), 5665–5670. https://doi.org/10.1073/pnas.1701805114

Nowak, M. A., & *Sigmund*, K. (1993). A strategy of win-stay, lose-shift that outperforms tit-for-tat in the Prisoner's Dilemma game. *Nature*, *364*, 56–58.

Nowak, M. A., *Tarnita*, C. E., & *Wilson*, E. O. (2010). The evolution of eusociality. *Nature*, *466*(7310), 1057–1062. https://doi.org/10.1038/nature09205

Pacheco, J. M., *Traulsen*, A., & *Nowak*, M. A. (2006). Coevolution of strategy and structure in complex networks with dynamical linking. *Phys. Rev. Lett.*, *97*(25), 258103. https://doi.org/10.1103/PhysRevLett.97.258103

Perc, M., *Jordan*, J. J., *Rand*, D. G., *Wang*, Z., *Boccaletti*, S., & *Szolnoki*, A. (2017). Statistical physics of human cooperation. *Phys. Rep.*, *687*, 1–51. https://doi.org/10.1016/j.physrep.2017.05.004

Perc, M., & *Szolnoki*, A. (2010). Coevolutionary games – A mini review. *BioSystems*, *99*(2), 109–125. https://doi.org/10.1016/j.biosystems.2009.10.003

Pinker, S. (1994). *The language instinct: How the mind creates language*. Penguin Books.

Post, D. M., & *Palkovacs*, E. P. (2009). Eco-evolutionary feedbacks in community and ecosystem ecology: Interactions between the ecological theatre and the evolutionary play. *Philos. Trans. R. Soc. B Biol. Sci.*, *364*(1523), 1629–1640. https://doi.org/10.1098/rstb.2009.0012

Rand, D. G., *Nowak*, M. A., *Fowler*, J. H., & *Christakis*, N. A. (2014). Static network structure can stabilize human cooperation. *PNAS*, *111*(48), 17093–17098. https://doi.org/10.1073/pnas.1400406111

Rapoport, A., *Chammah*, A. M., & *Orwant*, C. J. (1965). *Prisoner's dilemma: A study in conflict and cooperation* (Vol. 165). University of Michigan press.

Richerson, P., *Baldini*, R., *Bell*, A. V., *Demps*, K., *Frost*, K., *Hillis*, V., *Mathew*, S., *Newton*, E. K., *Naar*, N., *Newson*, L., *Ross*, C., *Smaldino*, P. E., *Waring*, T. M., & *Zefferman*, M. (2016). Cultural group selection plays an essential role in explaining human cooperation: A sketch of the evidence. *Behav. Brain Sci.*, *39*. https://doi.org/10.1017/S0140525X1400106X

Riedel, L., *Herdeanu*, B., *Mack*, H., *Sevinchan*, Y., & *Weninger*, J. (2020). Utopia: A comprehensive and collaborative modeling framework for complex and evolving systems. *JOSS*, *5*(53), 2165. https://doi.org/10.21105/joss.02165

Rocklin, M. (2015). Dask: Parallel computation with blocked algorithms and task scheduling (K. *Huff* & J. *Bergstra*, Eds.). In K. *Huff* & J. *Bergstra* (Eds.), *Proc. 14th python sci. conf.* https://doi.org/10.25080/Majora-7b98e3ed-013

Rong, Z., *Yang*, H. X., & *Wang*, W. X. (2010). Feedback reciprocity mechanism promotes the cooperation of highly clustered scale-free networks. *Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.*, *82*(4), 1–4. https://doi.org/10.1103/PhysRevE.82.047101

Roth, K. (2008). Scaling of water flow through porous media and soils. *Eur. J. Soil Sci.*, *59*(1), 125–130. https://doi.org/10.1111/j.1365-2389.2007.00986.x

Samuelson, P. A. (1954). The pure theory of public expenditure. *Rev. Econ. Stat.*, *72*(4), 387–389. https://doi.org/10.2307/1925895

Sánchez, A. (2018). Physics of human cooperation: Experimental evidence and theoretical models. *J. Stat. Mech. Theory Exp.*, *2018*(2). https://doi.org/10.1088/1742-5468/aaa388

Sanderson, C., & *Curtin*, R. (2016). Armadillo: a template-based C++ library for linear algebra. *JOSS*, *1*(2), 26. https://doi.org/10.21105/joss.00026

Sanderson, C., & *Curtin*, R. (2018). A user-friendly hybrid sparse matrix class in C++ (J. H. *Davenport*, M. *Kauers*, G. *Labahn*, & J. *Urban*, Eds.). In J. H. *Davenport*, M. *Kauers*, G. *Labahn*, & J. *Urban* (Eds.), *Math. softw. – icms 2018*, Cham, Springer International Publishing. https://doi.org/10.1007/978-3-319-96418-8_50

Santos, F. C., & *Pacheco*, J. M. (2005). Scale-free networks provide a unifying framework for the emergence of cooperation. *Phys. Rev. Lett.*, *95*(9), 1–4. https://doi.org/10.1103/PhysRevLett.95.098104

Santos, F. C., *Santos*, M. D., & *Pacheco*, J. M. (2008). Social diversity promotes the emergence of cooperation in public goods games. *Nature*, *454*(7201), 213–6. https://doi.org/10.1038/nature06940

Satyanarayan, A., *Moritz*, D., *Wongsuphasawat*, K., & *Heer*, J. (2017). Vega-Lite: A grammar of interactive graphics. *IEEE Trans. Vis. Comput. Graph.*, *23*(1), 341–350. https://doi.org/10.1109/TVCG.2016.2599030

Scheuring, I. (2005). The iterated continuous prisoner's dilemma game cannot explain the evolution of interspecific mutualism in unstructured populations. *J. Theor. Biol.*, *232*(1), 99–104. https://doi.org/10.1016/j.jtbi.2004.07.025

Schoener, T. W. (2011). The newest synthesis: Understanding ecological dynamics. *Science*, *331*(January), 426–429. https://doi.org/10.1126/science.1193954

Semmann, D., *Krambeck*, H. J., & *Milinski*, M. (2003). Volunteering leads to rock-paper-scissors dynamics in a public goods game. *Nature*, *425*(6956), 390–393. https://doi.org/10.1038/nature01986

Sender, R., *Fuchs*, S., & *Milo*, R. (2016). Are we really vastly outnumbered? Revisiting the ratio of bacterial to host cells in humans. *Cell*, *164*(3), 337–340. https://doi.org/10.1016/j.cell.2016.01.013

Sevinchan, Y. (2020). Paramspace. Zenodo. https://doi.org/10.5281/zenodo.3826471

Sevinchan, Y. (2021). *Evolution Mechanics and Perspectives on Food Web Ecology* (Doctoral dissertation). Heidelberg University.

Sevinchan, Y., *Herdeanu*, B., *Mack*, H., *Riedel*, L., & *Roth*, K. (2020a). Boosting group-level synergies by using a shared modeling framework (V. V. *Krzhizhanovskaya*, G. *Závodszky*, M. H. *Lees*, J. J. *Dongarra*, P. M. A. *Sloot*, S. *Brissos*, & J. *Teixeira*, Eds.). In V. V. *Krzhizhanovskaya*, G. *Závodszky*, M. H. *Lees*, J. J. *Dongarra*, P. M. A. *Sloot*, S. *Brissos*, & J. *Teixeira* (Eds.), *Computational Science – ICCS 2020*, Cham, Switzerland, Springer International Publishing. https://doi.org/10.1007/978-3-030-50436-6_32

Sevinchan, Y., *Herdeanu*, B., & *Traub*, J. (2020b). dantro: A Python package for handling, transforming, and visualizing hierarchically structured data. *JOSS*, *5*(52), 2316. https://doi.org/10.21105/joss.02316

Siek, J., *Lee*, L., & *Lumsdaine*, A.-W. (2002). *The Boost Graph Library: User Guide and Reference Manual.* Addison-Wesley.

Simon, H. A. (1962). The architecture of complexity. *Proc. Am. Philos. Soc.*, *106*(6), 467–482. https://doi.org/10.1080/14786443908649806

Solé, R., & *Valverde*, S. (2020). Evolving complexity: How tinkering shapes cells, software and ecological networks. *Philos. Trans. R. Soc. B Biol. Sci.*, *375*(1796). https://doi.org/10.1098/rstb.2019.0325

Solé, R. V., & *Valverde*, S. (2008). Spontaneous emergence of modularity in cellular networks. *J. R. Soc. Interface*, *5*(18), 129–133. https://doi.org/10.1098/rsif.2007.1108

Solomonoff, R., & *Rapoport*, A. (1951). Connectivity of randomness. *Bull. Math. Biophys.*, *13*. https://doi.org/10.1007/BF02478357

Spencer, H. (1866). *The Principles of biology* (Vol. 1). D. Appleton & Company.

Stanley, K. O., *Clune*, J., *Lehman*, J., & *Miikkulainen*, R. (2019). Designing neural networks through neuroevolution. *Nat. Mach. Intell.*, *1*(1), 24–35. https://doi.org/10.1038/s42256-018-0006-z

Steffen, W., *Richardson*, K., *Rockström*, J., *Cornell*, S. E., *Fetzer*, I., *Bennett*, E. M., *Biggs*, R., *Carpenter*, S. R., de *Vries*, W., de *Wit*, C. A., *Folke*, C., *Gerten*, D., *Heinke*, J., *Mace*, G. M., *Persson*, L. M., *Ramanathan*, V., *Reyers*, B., & *Sörlin*, S. (2015). Planetary boundaries: Guiding human development on a changing planet. *Science*, *347*(6223). https://doi.org/10.1126/science.1259855

Steffen, W., *Rockström*, J., *Richardson*, K., *Lenton*, T. M., *Folke*, C., *Liverman*, D., *Summerhayes*, C. P., *Barnosky*, A. D., *Cornell*, S. E., *Crucifix*, M., *Donges*, J. F., *Fetzer*, I., *Lade*, S. J., *Scheffer*, M., *Winkelmann*, R., & *Schellnhuber*, H. J. (2018). Trajectories of the Earth system in the anthropocene. *PNAS*, *115*(33), 8252–8259. https://doi.org/10.1073/pnas.1810141115

Stewart, A. J., & *Plotkin*, J. B. (2014). Collapse of cooperation in evolving games. *PNAS*, *111*(49), 17558–17563. https://doi.org/10.1073/pnas.1408618111

Storer, T. (2017). Bridging the chasm: A survey of software engineering practice in scientific programming. *ACM Comput. Surv.*, *50*(4), 1–32. https://doi.org/10.1145/3084225

Strogatz, S. H. (2014). *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering* (2nd ed.). Westview Press.

Stumpf, M. P., & *Porter*, M. A. (2012). Critical truths about power laws. *Science*, *335*(6069), 665–666. https://doi.org/10.1126/science.1216142

Szabó, G., & *Fáth*, G. (2007). Evolutionary games on graphs. *Phys. Rep.*, *446*(4-6), 97–216. https://doi.org/10.1016/j.physrep.2007.04.004

Szathmáry, E., & *Maynard Smith*, J. (1997). From replicators to reproducers: The first major transitions leading to life. *J. Theor. Biol.*, *187*(4), 555–71. https://doi.org/10.1006/jtbi.1996.0389

Szathmáry, E. (2015). Toward major evolutionary transitions theory 2.0. *PNAS*, *112*(33), 10104–10111. https://doi.org/10.1073/pnas.1421398112

Szathmáry, E., & *Maynard Smith*, J. (1995). The major evolutionary transitions. *Nature*, *374*, 227.

Szolnoki, A., & *Chen*, X. (2017). Environmental feedback drives cooperation in spatial social dilemmas. *Epl*, *120*(5). https://doi.org/10.1209/0295-5075/120/58001

Szolnoki, A., & *Perc*, M. (2009a). Emergence of multilevel selection in the prisoner's dilemma game on coevolving random networks. *New J. Phys.*, *11*. https://doi.org/10.1088/1367-2630/11/9/093033

Szolnoki, A., & *Perc*, M. (2009b). Resolving social dilemmas on evolving random networks. *EPL*, *86*(3). https://doi.org/10.1209/0295-5075/86/30007

Szolnoki, A., *Perc*, M., & *Szabó*, G. (2009). Topology-independent impact of noise on cooperation in spatial public goods games. *Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.*, *80*(5), 1–5. https://doi.org/10.1103/PhysRevE.80.056109

Taylor, T. (2016). WebAL Comes of Age. *Artif. Life*, *22*, 408–423. https://doi.org/10.1162/ARTL_a_00210

Taylor, T., & *Dorin*, A. (2020). *Rise of the self-replicators: Early visions of machines, AI and robots that can reproduce and evolve* (1st ed.). Springer International Publishing. https://doi.org/10.1007/978-3-030-48234-3

The European Space Agency (ESA). (2021). *How many stars are there in the Universe*. Retrieved August 21, 2021, from https://www.esa.int/Science_Exploration/Space_Science/Herschel/How_many_stars_are_there_in_the_Universe

The HDF Group. (1997). *Hierarchical data format version 5*. http://www.hdfgroup.org/HDF5

Tilman, A. R., *Plotkin*, J. B., & *Akçay*, E. (2020). Evolutionary games with environmental feedbacks. *Nat. Commun.*, *11*(1). https://doi.org/10.1038/s41467-020-14531-6

Trifonov, E. N. (2011). Vocabulary of definitions of life suggests a definition. *J. Biomol. Struct. Dyn.*, *29*(2), 259–266. https://doi.org/10.1080/073911011010524992

Trivers, R. L. (1971). The evolution of reciprocal altruism. *Q. Rev. Biol.*, *46*(1), 35–57. https://doi.org/10.1086/406755

Turcotte, M. M., *Reznick*, D. N., & *Hare*, J. D. (2011). The impact of rapid evolution on population dynamics in the wild: Experimental test of eco-evolutionary dynamics. *Ecol. Lett.*, *14*(11), 1084–1092. https://doi.org/10.1111/j.1461-0248.2011.01676.x

Utopia Developer Team. (2021). Utopia project webpage, The Utopia Developer Team.

Vahdati, A. (2019). Agents.jl: Agent-based modeling framework in Julia. *JOSS*, *4*(42), 1611. https://doi.org/10.21105/joss.01611 van der *Walt*, S., *Colbert*, S. C., & *Varoquaux*, G. (2011). The NumPy array: A structure for efficient numerical computation. *Comput. Sci. Eng.*, *13*(2), 22–30. https://doi.org/10.1109/MCSE.2011.37

van *Cleve*, J., & *Akçay*, E. (2014). Pathways to social evolution: Reciprocity, relatedness, and synergy. *Evolution*, *68*(8), 2245–2258. https://doi.org/10.1111/evo.12438

VanderPlas, J., *Granger*, B., *Heer*, J., *Moritz*, D., *Wongsuphasawat*, K., *Satyanarayan*, A., *Lees*, E., *Timofeev*, I., *Welsh*, B., & *Sievert*, S. (2018). Altair: Interactive statistical visualizations for Python. *JOSS*, *3*(32), 1057. https://doi.org/10.21105/joss.01057

van *Valen*, L. (1973). A new evolutionary law. *Evol. Theory*, *1*, 1–30.

van *Veelen*, M. (2020). The group selection-inclusive fitness equivalence claim: Not true and not relevant. *Evol. Hum. Sci.*, *2*, 2018–2020. https://doi.org/10.1017/ehs.2020.9

van *Veelen*, M., *Allen*, B., *Hoffman*, M., *Simon*, B., & *Veller*, C. (2017). Hamilton's rule. *J. Theor. Biol.*, *414*(November 2015), 176–230. https://doi.org/10.1016/j.jtbi.2016.08.019

Verhulst, P.-F. (1838). Notice sur la loi que la population suit dans son accroissement. *Corresp. Math. Phys.*, *10*, 113–126.

Visick, K. L., *Stabb*, E. V., & *Ruby*, E. G. (2021). A lasting symbiosis: How Vibrio fischeri finds a squid partner and persists within its natural host. *Nat. Rev. Microbiol.* https://doi.org/10.1038/s41579-021-00557-0

von *Neumann*, J. (1928). Zur Theorie der Gesellschaftsspiele. *Math. Ann.*, *100*(1), 295–320. https://doi.org/10.1007/BF01448847

Wahl, L. M., & *Nowak*, M. A. (1999). The continuous Prisoner's dilemma: II. Linear reactive strategies with noise. *J. Theor. Biol.*, *200*(3), 323–38. https://doi.org/10.1006/jtbi.1999.0997

Wang, X., & *Fu*, F. (2020). Eco-evolutionary dynamics with environmental feedback: Cooperation in a changing world. *Epl*, *132*(1). https://doi.org/10.1209/0295-5075/132/10001

Wang, Z., *Wang*, L., *Szolnoki*, A., & *Perc*, M. (2015). Evolutionary games on multilayer networks: A colloquium. *Eur. Phys. J. B*, *88*(5). https://doi.org/10.1140/epjb/e2015-60270-7

Waskom, M. (2021). Seaborn: Statistical data visualization. *JOSS*, *6*(60), 3021. https://doi.org/10.21105/joss.03021

Watson, R. A., *Mills*, R., *Buckley*, C. L., *Kouvaris*, K., *Jackson*, A., *Powers*, S. T., *Cox*, C., *Tudge*, S., *Davies*, A., *Kounios*, L., & *Power*, D. (2016). Evolutionary connectionism: Algorithmic principles underlying the evolution of biological organisation in evo-devo, evo-eco and evolutionary transitions. *Evol. Biol.*, *43*(4), 553–581. https://doi.org/10.1007/s11692-015-9358-z

Watson, R. A., & *Szathmáry*, E. (2016). How Can Evolution Learn? *Trends Ecol. Evol.*, *31*(2), 147–157. https://doi.org/10.1016/j.tree.2015.11.009

Watts, D. J., & *Strogatz*, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, *393*(6684), 440–442. https://doi.org/Doi10.1038/30918

Weitz, J. S., *Eksin*, C., *Paarporn*, K., *Brown*, S. P., & *Ratcliff*, W. C. (2016). An oscillating tragedy of the commons in replicator dynamics with game-environment feedback. *PNAS*, *113*(47), E7518–E7525. https://doi.org/10.1073/pnas.1604096113

Wenke, K., *Kai*, M., & *Piechulla*, B. (2010). Belowground volatiles facilitate interactions between plant roots and soil organisms. *Planta*, *231*(3), 499–506. https://doi.org/10.1007/s00425-009-1076-2

Whiteley, M., *Diggle*, S. P., & *Greenberg*, E. P. (2017). Progress in and promise of bacterial quorum sensing research. *Nature*, *551*(7680), 313–320. https://doi.org/10.1038/nature24624

Wilensky, U. (1999). NetLogo. *Evanston, IL*, Center for Connected Learning; Computer-Based Modeling, Northwestern University,

Wilkinson, G. S. (1990). Food sharing in vampire bats. *Sci. Am.*, *262*(2), 76–83.

Wilson, D. S. (2015). *Does altruism exist?* Yale University Press.

Wilson, D. S., & *Wilson*, E. O. (2007). Rethinking the theoretical foundation of socialbiology. *Q. Rev. Biol.*, *82*(4), 327–348.

Wolfram, S. (1983). Statistical mechanics of cellular automata. *Rev. Mod. Phys.*, *55*(3), 601–644. https://doi.org/10.1103/RevModPhys.55.601

Wu, B., *García*, J., *Hauert*, C., & *Traulsen*, A. (2013). Extrapolating weak selection in evolutionary games. *PLoS One*, *9*(12).

Yoneya, K., & *Takabayashi*, J. (2014). Plant-plant communication mediated by airborne signals: Ecological and plant physiological perspectives. *Plant Biotechnol.*, *31*(5), 409–416. https://doi.org/10.5511/plantbiotechnology.14.0827a

Yules, G. U. (1924). A mathematical theory of evolution, based on the conclusions of Dr. J. C. Willis, F.R.S. *R. Soc.*, *213*(3).

Symbols

Agent traits and states require a subscript a denoting the specific agent. For example, the age of agent a is A_a. For elegance and visual noise reduction, I will omit this subscript a in the following list, i.e., will denote the age with A. However, throughout the dissertation, I will use the exact notation where it is beneficial for understanding.

t	The time defining one model iteration step. 37
N	The total number of agents. 60
N_t	The total number of agents at time t. 37
R	An agent's internally stored resources; agent state. 40
A	An agent's age; agent state. 60
A^{final}	An agent's final age; agent state. 41
p_s	An agent's probability of extracting synergistic resources at any time t within its lifetime; evolving agent trait. 38
p_b	An agent's probability of extracting basic resources; $p_b = 1 - p_s$; evolving agent trait. 38
$\mathcal{B}_t$	The set of agents sampled to extract basic resources at time t. 41
$\mathcal{S}_t$	The set of agents sampled to extract synergistic resources at time t. 41
$\mathfrak{R}_b$	The basic resource, which agents can extract individually. 40
$\mathcal{A}_b$	The amount of basic resource $\mathfrak{R}_b$ available each time step; global system trait. 41
s	The strength of an agent; evolving agent trait. 42
μ_s	An agent's expected strength actually used for basic resource extraction; $\mu_s = s \cdot p_b$; deduced agent trait. 42
j_b	The resource inflow from basic resource $\mathfrak{R}_b$ to an agent in one time step; global system trait. 41
$\mathfrak{R}_s$	The synergistic resource, which agents can extract only through social interaction. 40
P	An agent's total payoff from all its interactions within it's social environment; agent state. 48

j_{s_a} The resource outflow from the synergistic resource $\mathfrak{R}_s$ from a sub-interaction centered around an agent in a time step. 48

$\mathcal{A}_s$ The amount of $\mathfrak{R}_s$ available each time step; global system trait. 51

i An agent's investment trait: How many resources will an agent invest if it has enough resources internally stored in R in one time step; evolving agent trait. 48

ι An agent's actual investment in one time step determined through equation 4.13; derived agent trait. 47

μ_i An agent's expected investment over its lifetime if it has enough resources internally stored: $\mu_i = i \cdot p_s$; derived agent trait. 48

r The synergy factor transforming investments into shared goods; global system trait. 47

c_{s_a} An agent's cost to invest in goods creation. 44

r^+ The positive synergy factor transforming positive investments into created goods; global system trait. 44

G_v^+ The goods created in the synergistic interaction restricted to positive investments centered around the agent at vertex v. 44

P^+ An agent's total payoff from all synergistic interactions restricted to positive investments; agent state. 44

g_a An agent's grabbed resources from the synergistic resource. 45

r^- The negative synergy factor transforming grabbed resources into bads (destructed goods); global system trait. 45

B_v The bads created (goods destructed) in the destructively synergistic interaction centered around the agent at vertex v. 45

P^- An agent's total payoff from all destructice interactions; agent state. 46

$\mathcal{K}_a^t$ The set of neighboring agents linked to an agent excluding a at time t. 43

k The number of neighbors of an agent corresponding to the degree of an agent; $k_a = \#\mathcal{K}_a$. 53

$\mathcal{N}_a^t$ The set of agents linked to an agent including a at time t. 43

c_l An agent's cost of living subtracted each time step. 38

c_s An agent's cost of strength subtracted each time step. 38

t_a An agent's threshold for adding links; evolving agent trait. 53

κ_l A source agent's cost for adding a local link; global trait. 53

κ_g A source agent's cost for adding a global link; global trait. 53

κ_r A source agent's cost for removing a link; global trait. 54

γ_l A target agent's cost for adding a local link; global trait. 53

γ_g A target agent's cost for adding a global link; global trait. 60

γ_r A target agent's cost for removing a link; global trait. 54

$t_a^{\max}$ The maximal threshold for adding links; global trait. 53

t_r An agent's threshold for removing links; evolving agent trait. 54

$t_r^{\max}$ The maximal threshold for removing links; global trait. 54

p_l An agent's probability of adding a local links at a time t; evolving agent trait. 53

p_g An agent's probability of adding a global links at a time t; $p_g = 1 - p_l$; evolving agent trait. 53

ν_l An agent's link mode defining which agent to target for local addition within the next-neighborhood; Integer encoding: $0 \leftrightarrow$ *None*, $1 \leftrightarrow R$, $2 \leftrightarrow p_s$, $3 \leftrightarrow i$, $4 \leftrightarrow$ *random*, $5 \leftrightarrow s$, $6 \leftrightarrow G$, and $7 \leftrightarrow P$; evolving agent trait. 53

ν_g The link mode defining which agent to target for global linking within the population; Integer encoding: $0 \leftrightarrow$ *None*, $1 \leftrightarrow R$, $2 \leftrightarrow p_s$, $3 \leftrightarrow i$, $4 \leftrightarrow$ *random*, $5 \leftrightarrow s$, $6 \leftrightarrow G$, and $7 \leftrightarrow P$; evolving agent trait. 53

ν_r The link mode defining which agent to target for removing links; Integer encoding: $0 \leftrightarrow$ *None*, $1 \leftrightarrow R$, $2 \leftrightarrow p_s$, $3 \leftrightarrow i$, $4 \leftrightarrow$ *random*, $5 \leftrightarrow s$, $6 \leftrightarrow G$, and $7 \leftrightarrow P$; evolving agent trait. 54

p_δ An agent's death probability per time step; global system trait. 38

t_δ An agent's death threshold: If $R < t_\delta$, the agent dies from exhaustion; global system trait. 38

t_β An agent's birth threshold: If $R \geq t_\beta$, the agent gives birth to an offspring with probability p_β; global system trait. 38

p_β An agent's probability of giving birth to an offspring per time step if the birth threshold t_β is exceeded; global system trait. 38

c_β The birth cost of creating an offspring subtracted from the parent; global system trait. 38

j_β The transferred resources from parent to offspring. 38

p_{ρ_l} The probability to mutate the local link addition mode. 58

p_{ρ_g} The probability to mutate the global link addition mode. 58

p_{ρ_r} The probability to mutate the link removal mode. 58

Index & Acronyms

ABM agent based model 25, 27, 28
agent 37

Barabási-Albert network 22
basic resources 37
benefactor 49, 50

CA cellular automaton 25, 27, 28
carrying capacity 39
CEP competitive exclusion principle 41
CG coordination game 16, 20
CLI command line interface 28
climate change 63
competition dilemma 73
connected component 21
cooperator 49

DAG directed acyclic graph 31
dantro 30
data transformation framework 30
data tree 30
defector 49
degree 21
degree distribution 21
development 39
distance 21
distribution dilemma 124

effective interaction network 42
effective social environment 43
EGT Evolutionary Game Theory 12
Erdös-Rényi network 22
ESS Evolutionary Stable Strategy 15
evolution 37
evolutionary pressure 39
exploiter 49, 50

generation 56
global warming 63
gPGG generalized public goods game 38, 43
greenhouse gas 63

HG harmony game 16, 20

in-degree 21
inheritance 57
internal resource reservoir 37

loner 49
long-tail 22

malefactor 49, 50
multiverse 29

neighborhood 43
network 20
network model 25, 27, 28
next-neighborhood 43

out-degree 21

path 21
PD prisoner's dilemma 13, 16
personal dilemma 69
PGG public goods game 13, 16, 17
plotting framework 30
population structure 43
profiteer 49, 50

quorum sensing 64

reciprocity 18
Red Queen dynamics 20
resource-flow 37
RPS rock-paper-scissors game 16
rule function 37

scale-free 20
SD snowdrift game 16
seed 83
small-world 20
social environment 37
social inheritance 22
state 37

strategy 39
strict tragedy of the commons 68
sub-interaction 43, 44
synergistic resources 37

tragedy of the commons 16
trait 37
trait evolution plot 79
true defection 13

universe 29
Utopia 23
utopya 29

variation 57

Watts-Strogatz network 22
weak selection 17