Evolutionary Agent-Based Economics

Evolutionary Agent-Based Economics

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van
de Rector Magnificus, prof.dr. R.A. van Santen, voor een
commissie aangewezen door het College voor
Promoties in het openbaar te verdedigen
op donderdag 7 oktober 2004

door

Floortje Alkemade

geboren te Hengelo

Dit proefschrift is goedgekeurd door de promotoren:

prof.dr.ir. J.A. La Poutré
en
prof.dr. H.M. Amman

The work in this thesis has been carried out at the Centrum voor Wiskunde en Informatica (CWI) under the auspices of the research school IPA (Institute for Programming research and Algorithmics). The research is part of the project "Evolutionary Exploration Systems for Electronic Markets" funded by NWO, project 612-60-008.

CIP-DATA LIBRARY TECHNISCHE UNIVERSITEIT EINDHOVEN

Alkemade, Floortje

Evolutionary agent-based economics / by Floortje Alkemade. - Eindhoven : Technische Universiteit Eindhoven, 2004. - Proefschrift. -
ISBN 90-386-2107-8
NUR 788
Keywords: Agent-based economics / Evolutionary algorithms / Electronic commerce / Social networks

IPA dissertation series 2004-15

Acknowledgments

I am grateful to the many people who have helped and supported me in the past few years. First of all, I thank my supervisors Han La Poutré and Hans Amman for their valuable input and the many helpful discussions. I had a very enjoyable time at the Centre for Mathematics and Computer Science (CWI) in the SEN4 group and I am grateful to Ivan, Enrico, Michiel, David, Rob, Pieter Jan, Cees, Bill, Koye, Jano, Valentin and Sander for this. I am also grateful to Herbert Dawid and Joost Kok for their valuable input as members of my kernel committee.

The Santa Fe Institute 2002 Complex Systems Summer School was a very inspiring and fun event and working with Carolina Castaldi and Bill Rand provided both interesting research opportunities and a lot of fun. I also thank Carolina and Ben Vermin for being my 'paranimfen'. The course on Economics of Networks by Sanjeev Goyal at the Tinbergen institute proved to be a great inspiration and starting point for an important part of this thesis.

On a personal level, I would like to thank my parents for their invaluable love and support throughout the years. Finally, I thank Tomas for being there and making me laugh every day.

Contents

Chapter 1

Evolutionary Agent-Based Economics: Introduction

In the last few years, scientists working in the fields of economics and social science have abandoned the assumption of perfect rationality. Theories of (economic) behavior and decision making often assume that decision makers have all the capabilities and information needed to make an optimal decision, that is that they exhibit perfect rationality. Furthermore, in such theories it is often assumed that agents (decision makers) are homogeneous (i.e., have similar characteristics and preferences) and that interaction is anonymous and global. Theoretical outcomes predicted by these theories often differ from outcomes observed in the real world. However, less restrictive and more realistic assumptions regarding agent behavior often lead to analytically untractable models. Recent advances in computer science have however made it possible to simulate artificial societies and thus study economic models by running computer simulations. This new field of study is called Agent-based Computational Economics (ACE).

ACE combines elements and views from economics, social sciences and computer science to study such artificial societies. The decision makers in these societies are called agents. Also, ACE allows us to study large systems of such interacting economic agents from the bottom up. Furthermore, we can relax some of the assumptions of traditional economics. More specifically, we can consider agents that are so-called boundedly rational as opposed to perfectly rational and that are heterogeneous in their behavior, that is agents may differ in their decision making capabilities and strategies. Furthermore, this approach allows us to explicitly take the role of interaction into account. ACE holds the view that individual actions, and in turn aggregate outcomes, are in a large part determined by the interaction between agents (Tesfatsion 2001). This stands in contrast with the market view of the economy where buyers and sellers are anonymous and the structure of the interaction is typically considered less important. This thesis deals with the behavior of boundedly rational, heterogeneous agents as described above. More specifically, we look at boundedly rational agents that have to adapt their behavior and strategies both in response to changes in the environment and to changes in the behavior of other agents.

We use an evolutionary algorithm approach to model the learning and search behavior of economic agents. Evolutionary algorithms are based on the principle of "survival of the fittest" from nature. Evolutionary algorithms were first developed by Holland (1975) and he identified economics as an application field for evolutionary algorithms. However, the first evolutionary algorithm applications to economics were in the operations research domain where evolutionary algorithms were used to solve specific problems. Only recently have researchers begun to study and refine economic theories and the relation between micro-level behavior and macro-level outcomes through agent-based simulation. The evolutionary approach allows us to study the process as well as the possible equilibrium outcomes of the economic systems under investigation. An example of this approach that we study in this thesis are the new market structures that arise through electronic commerce and trading over the Internet. We use Evolutionary Agent-Based Computational Economics models (ACE models using evolutionary algorithms) to study, for example, the emergent behavior and the new market structures that arise, when we consider the trade and social networks between agents in Chapter 5 and Chapter 6.

As described above, Evolutionary Agent-Based Computational Economics differs from traditional economics in two important aspects. Agents are boundedly rational instead of perfectly rational and interaction is not global and anonymous. These two factors form the basis of this thesis and are discussed in more detail below. Section 1.1 discusses heterogeneous, boundedly rational agents while Section 1.2 is concerned with interaction structure. Section 1.3 gives an overview of the relevant concepts of evolutionary algorithms. Finally, Section 1.4 provides the outline for the remainder of this thesis and Section 1.5 gives an overview of previously published parts of this thesis.

1.1 Heterogeneous, boundedly rational agents

This thesis is concerned with heterogeneous, boundedly rational agents. In this section we will describe the field of agent-based computational economics that studies systems of such interacting agents. Agent-based computational economics combines elements from computer science and economics. First, we will give a brief overview of the concept of agent as it is used in those disciplines and relate this to our study. Then we will give a more detailed description of agent-based economics. Finally we will address the issue of bounded rationality in more detail.

1.1.1 Agents in economics

In economics we encounter ‘agents’ in two different contexts. First, we discuss the notion of the so-called *representative agent*. A representative agent is an agent that has the characteristics that best represent the average economic actor in the model. Representative agent models then study the behavior of a whole population of such agents. Models that make use of a representative agent, are thus essentially homogeneous. ACE is both a useful and necessary addition to representative agent models since it allows us to design and study heterogeneous agent models.

Second, economic agency theory—see Eisenhardt (1989) for an overview—describes and investigates the *agency relationship*, in which one party (the principal) delegates work to another (the agent), who performs that work. The so-called principal-agent problem arises when the agent has an informational advantage over the principal and has different interests and a different attitude towards risk than the principal. Agency theory investigates how to get the agent to act in the best interests of the principal. Eisenhardt (1989) states that: “In general, agency theory is thus concerned with relationships between a principal and an agent who are engaged in cooperative behavior, but have differing goals and differing attitudes toward risk". Agency theory attempts to ensure this cooperative behavior by formulating a contract between the principal and the agent. Principal-agent theory can be applied to employer-employee, lawyer-client, buyer-supplier and other agency relationships.

1.1.2 Agents in computer science

The following definition for software *agent* as can be found in Weiss (1999) and Wooldridge and Jennings (1995) is often used in computer science:

> *An agent is a computer system that is situated in some environment, and that is capable of autonomous action in this environment in order to meet its design goals.*

Software agent research was first motivated by the wish to design pieces of software that can perform some task for their owner as would a human agent, i.e., a travel agent or stockbroker (Maes 1994). In the case of software agents, the principal from economic agency theory (see Section 1.1.1) is usually referred to as the *owner* or *user* of the agent. Autonomy is central to the notion of agency in computer science and refers to the property that agents are able to act without the intervention of humans or other systems: they have control both over their own internal state and their behavior. The degree of autonomy the agent has depends on whether it is able (and allowed) to adequately represent the user and how competent it is in autonomously performing the delegated tasks. The level of intelligence the agent exhibits is determined by its reasoning and learning capabilities and its ability to adapt to his environment (Turban and Aronson 1998, Gilbert and Janca 1997). As technology progresses, the boundaries between ordinary programs and software agents and between "dumb" and intelligent agents will continually shift. Therefore, there exists no commonly accepted definition of intelligent software agent. Several authors have however suggested characteristics that intelligent agents should exhibit, see for instance (Bradshaw 1997).
Multi-agent systems (Faratin *et al.* 2003, Chiarella *et al.* 2003, Fatima *et al.* 2003, Hu and Wellman 2003, Walsh and Wellman 2003, Weiss 1999) are

systems where a group of software agents work together to solve a number of tasks.

1.1.3 Agent-Based Computational Economics

In agent-based computational economics, we define an *agent* as an individual actor that has some (limited) reasoning and decision-making capabilities. This is the definition we will use in the remainder of this thesis. Agent-based computational economics (ACE) is the research field that is concerned with systems of such interacting agents. Tesfatsion (2001) defines ACE as:

> *The computational study of economies modeled as evolving systems of autonomous interacting agents.*

ACE investigates how the interactions between individuals can give rise to certain economic or social phenomena that are so far unexplained by traditional economics. An ACE simulation starts with a population of agents, some behavioral rules for these agents, a definition of the possible interactions between the agents and some external influences. During the simulation agents try to achieve their goals by adapting to other agents and the changing environment. Thus the relationship between micro-level behavior, that is the behavior of the individual agents, and macro-level outcomes, that is the global patterns that emerge through all those individual interactions, can be studied. ACE simulation can give us insights in topics such as equilibrium selection (i.e., which rest-points are most likely for the system of agents when theory predicts multiple possibilities), new sources of dynamics, analysis of 'regime changes' (sudden changes in aggregate behavior), retreating from perfect rationality as well as new optimization and estimation models (Simon 1984, Sargent 1993, Albin 1998, Kalai and Lehrer 1993, NachBar 1997).

The notion of *agent* in ACE is thus broader than in economic agency theory. While principal-agent problems are interesting and can be studied using ACE, agents in ACE can represent many other types of economic actors. More specifically, ACE can also be used to study agent systems where agents display competitive instead of cooperative behavior. Furthermore, ACE allows us to study systems where many agents interact while principal-agent theory is mostly concerned with the relationship between two cooperating agents. The notion of *agent* in ACE also differs from that of software agent described above. Software agents do not necessarily model economic behavior as in ACE. Furthermore, intelligent software agents are often specialists that excel at a particular task but do not necessarily display learning or adaptive behavior as in ACE. While multi-agent systems can also be used to model economic systems, the majority of multi-agent systems research focuses on solving a particular problem or developing intelligent agents that do not necessarily model the behavior of economic agents. Furthermore, the majority of multi-agents systems research is concerned with cooperative agent systems while ACE also focuses on competitive behavior.

1.1.4 Bounded Rationality

Perfectly rational agents have all the decision-making capabilities and the information needed to make an optimal decision. Perfect rationality is assumed in many economic and game theoretic models. Motivated by the fact that decision makers often have to deal with limited time, limited resources and incomplete information, Simon (1984) introduced the notion of bounded rationality. Bounded rationality describes the fact that decision makers often do not have the computational capabilities and the information needed to make a perfectly rational decision. Bounded rationality thus addresses the fact that economic agents usually do not exhibit perfect rationality as assumed in most economic models, but rather that their rationality is bounded. Bounded rationality with respect to computational capabilities means that agents do not have the resources (for example time, money, memory or computing facilities) to consider all possible courses of action and remember the effects of past behavior. Bounded rationality with respect to available information covers a wide range of restrictions on the behavior of the agents. First, agents may not have perfect information about their environment but rather acquire information through interaction with the environment. Furthermore, the environment may change over time due to agent actions or external influences. Second, agents may not know all there is to know about the other agents in the environment, and third, agents may not be able to correctly observe the decisions made by other agents.

Replacing the assumption of perfectly rational agents by that of boundedly rational agents allows us to build more realistic models of agent decision making processes. Several authors have addressed the issue of bounded rationality in economic models (Sargent 1993, Albin 1998, Kalai and Lehrer 1993, NachBar 1997). It is interesting to note that sometimes boundedly rational agents are able to arrive at economic outcomes that are preferred (in terms of e.g., payoff to the agents) over the theoretical outcome (Arthur 1993). This suggests that the concept of bounded rationality can play a role in explaining the discrepancies between theoretical outcomes and the observed reality.

Different ways of dealing with bounded rationality can be observed in real economic agents and have also been investigated using agent-based economics. An example of such boundedly rational agents in ACE that are inspired by real-world observations are agents that use heuristic rules or simple rules of thumb to make their decisions. Other mechanisms such as imitation, the use of tags, trust, loyalty and reputation play a large role in real markets and have been incorporated in ACE models. All these mechanisms help to compensate for the lack of (correct) information in boundedly rational decision processes and help agents to make decisions based on limited information. In this thesis several of these mechanism are studied. Chapter 2, for example, discusses several types of bounded rationality and their effects on economic outcomes and Chapter 3 addresses the use of tags as a means to guide the interaction between agents.

1.2 The role of interaction in ACE

In agent-based economics, interaction plays an important role. In traditional economics, interaction is anonymous and global, that is, agents cannot choose and do not have preferences about whom they have interaction with and furthermore, they are equally likely to interact with all other agents. In the real world interaction is often not anonymous, agents have repeated encounters with other agents and mechanisms like for example reputation and trust may lead to a preference for one agent over the other. Furthermore, agents may display different (types of) behavior dependent on whom they interact with, this is called conditional interaction. Spatial and time restrictions often cause interaction not to be global. But even in the current times of instant trade over the internet, the trade partners of an agent are restricted by the number of partners he knows about, that is the interaction structure of the agents depends on their social network.

The importance of considering the effects of interaction between agents is illustrated in the famous work by Schelling (1978) on segregation. Schelling considered a grid with two types of agents. The agents only interact with their eight immediate neighbors. An agent decides to move to a preferred spot on the grid if less than 37.5 percent of its neighbors are of the same type as itself. The simulation was initialized with a fully mixed grid (a checkerboard pattern) and then started by a small perturbation of this pattern. Schelling found that even such a slight preference for one's neighbor to be of the same type as himself would lead to complete segregation on the grid within a few iterations, even though initially the grid was mixed. This work is an example of a strong emerging phenomenon based upon social interaction and Schelling's work is considered to be one of the first examples of ACE.

In this thesis we examine how outcomes change if the assumptions of global and anonymous interaction are relaxed. Chapter 3 treats the situation where interaction is not anonymous. Chapter 5 and Chapter 6 discuss the case where interaction is not global but restricted by the social network of the economic agents. Below we will give a short introduction to social networks as they are used in Chapter 5 and Chapter 6.

1.2.1 Social Networks

A social network is a set of people or groups of people with some pattern of contacts or interactions between them (Scott 2000, Wasserman and Faust 1994). A social network can be represented by a graph where the nodes are agents and the edges are the connections or links between the agents. These connections signify that there is a relationship between two agents and that they can for example trade or exchange information with each other. Both social networks with undirected and directed edges have been studied in the literature (Bala and Goyal 2000, Kranton and Minehart 2001).

Below, we discuss some relevant terms and concepts from the literature on networks. Dense networks are networks that contain many links, the network where each agent has a direct link to all other agents is called the complete, or fully-connected network. The empty network is the network where there are no links at all between the agents, while a network with only a few links is called a sparse network. A path through a graph is a traversal of consecutive vertices along a sequence of edges. When there exists a path in the network between any two agents in the network, the network is said to be connected and information from one agent can spread over the links to all other agents in the network. This is not possible in a disconnected network. Disconnected networks can consist of multiple connected subnetworks (components) or of one connected subnetwork (component) and a few isolated agents. Figure 1.1 gives examples of the different types of networks. Network A in Figure 1.1 thus consists of twelve components, network B consists of one component and finally, network C has two components.

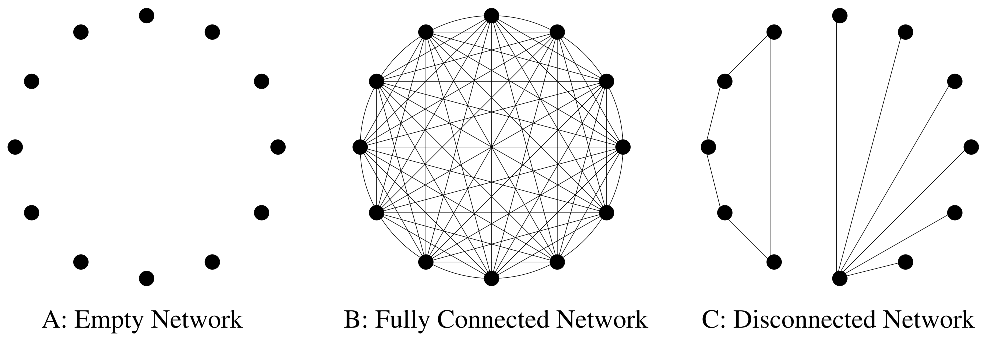

Figure 1.1: Different network topologies: A: Empty, B: Fully-connected network and C: disconnected network.

The degree of a node in the network describes the number of neighbors of a given node (agent) in the network. If all nodes have an equal number of links, then the network is called a regular network, and this number is referred to as the degree of the network. Two properties of networks are especially important when we consider the spread of information over a network. First, the average path length of a network is a measure for how may steps it takes on average to go from one random node to a random second node. Note that this path length may be infinitely long for disconnected networks. In general, (connected) regular graphs have long path lengths while (connected) random graphs are characterized by short path lengths.

Second, the clustering coefficient of a network describes how 'clustered' a network is. If a network exhibits a high level of clustering, there is an increased probability that the neighbors of a particular node are also direct neighbors of each other (compared to random networks). Clustering, or network transitivity as it is sometimes called, thus describes the fact that in many networks it is found that if node A is connected to node B and node B to node C, then there is a high probability that node A will also be connected to node C. One measure that can be used to make the level of clustering in a network more explicit is the clustering coefficient C. C is the probability that two nodes that are network neighbors of the same other node will themselves be neighbors and can be defined as follows:

$$C = \left[\frac{\text{3 x number of triangles in the network}}{\text{number of connected triples of nodes}}\right]$$

Figure 1.2 depicts three types of networks which are of special interest: The k-regular network is a network where all agents have exactly k neighbors. In this thesis we restrict ourselves to regular networks that are modeled as a (one dimensional) ring lattice where each agent is connected to its k nearest neighbors by undirected edges. Apart from regular networks we also consider the random networks, small world networks and networks formed by preferential attachment. In a random network the links between agents are constructed randomly. Small world networks and networks formed by preferential attachment are inspired by observations made on real world networks, as follows.

As originally defined in Watts and Strogatz (1998), small world networks are obtained from regular lattices (in this case one dimensional 'ring' networks) by rewiring a few randomly chosen edges. In another version (Newman and Watts 1999), they are graphs whose vertices are connected together in a regular lattice, with the addiction of a small number of connections bridging randomly chosen vertices. In this thesis we restrict ourselves to Watts-Strogatz small world networks, i.e, small world networks that can be disconnected. Small world networks are thought to be a good model

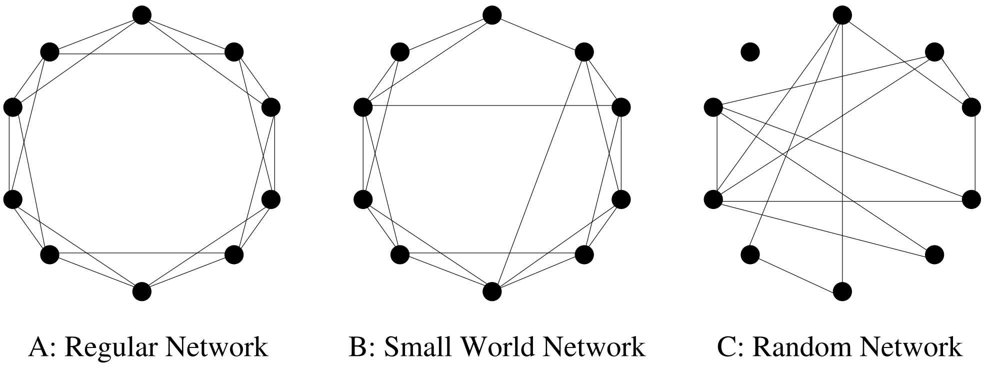

Figure 1.2: Different network topologies: A regular network, a small world network and a random network.

for many real world networks (Watts and Strogatz 1998). The small world effect, is the observation that most pairs of nodes in most networks seem to be connected by a short path through the network. The small world effect has obvious implications for the dynamics of processes taking place on such networks. For example, the small world effect implies that the spread of information across a network will be fast on most real-world networks. In recent years the term small world effect has taken on a more precise meaning: networks are said to show the small world effect if the value of the average path length scales logarithmically or slower with network size for fixed mean degree. Small world networks are thought to be a good approximation of many real world networks and they are special because they exhibit both short path lengths and high clustering. Small-world graphs show a higher level of clustering than random graphs, while preserving short average path lengths. Their pattern is not as ordered as in a regular lattice, but they still exhibit high clustering.

In addition to the research describing properties of real world networks researchers have recently begun to investigate how these properties have evolved. In these models, the networks typically grow by the gradual addition of vertices and edges in some manner intended to reflect growth processes that lead to characteristic structural features of the network. A common property of many large networks is that the node connectivities follow a scale-free power-law distribution. This feature was found to be a consequence of two generic mechanisms: (i) networks expand continuously by the addition of new nodes and (ii) new nodes attach preferentially to sites that are already well connected. This phenomenon is called **preferential attachment** as first described by Barabási and Albert (1999).

A related feature of real world social networks is assortative mixing or homophily. In most kinds of social networks there are at least a few different types of nodes or agents, and the probability of connection between agents often depends on types. That is, agents are more likely to interact with other agents that are similar to them in some respect, this is called **assortative mixing**. Assortative mixing thus describes processes where agents prefer to link up with similar agents instead of with well-connected agents as with preferential attachment.

The topology of the network can have great influence on the interaction between agents as well as on the outcome of the interaction. In Chapter 5 and Chapter 6 we study social networks and the economic outcomes that arise through interaction over such networks. More specifically, in Chapter 5 we study a network between consumers, producers and intermediaries where links represent opportunities to buy a product. Chapter 6 is concerned with the influence of the topology of the social network on the spread of information over the network.

1.3 Evolutionary algorithms

An evolutionary algorithm is a biologically inspired technique that uses the concept of survival of the fittest to evolve solutions to a particular problem and were first described by Holland (1975). For an overview of evolutionary algorithms see Mitchell (1996). The class of evolutionary algorithms consists of different techniques such as genetic programming, evolution strategies and genetic algorithms. In this thesis we restrict our attention to the genetic algorithms (GA). A typical GA can be described as follows (Mitchell 1996), see also Figure 1.3.

First, a population of randomly initialized strategies is generated. This is the population of chromosomes or genotypes. The behavior that is encoded in these chromosomes, that is, the behavior exhibited by an agent using that particular strategy can be seen as the phenotype. In this thesis we restrict

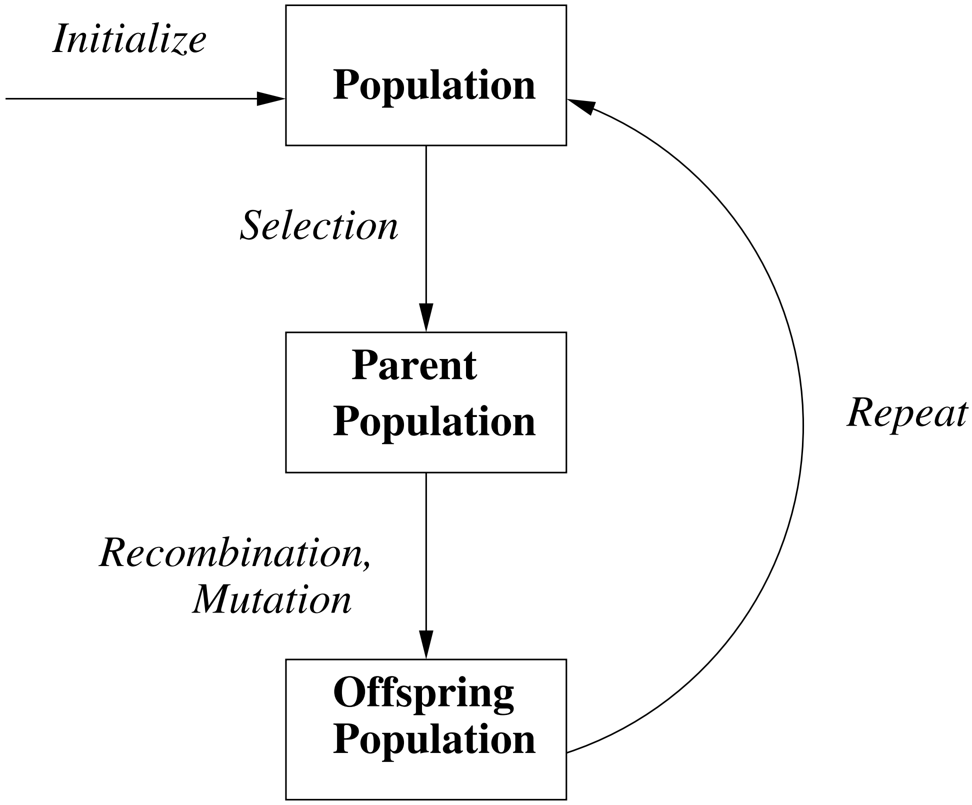

Figure 1.3: Outline of a typical Genetic Algorithm.

ourselves to binary encoded chromosomes (that is, the chromosome consists of zeros and ones). This population of chromosomes is subsequently changed and improved in a number of *generations* by means of *selection*, *recombination* (crossover), and *mutation*. Selection chooses the better strategies (with higher accumulated payoffs) that will serve as parents for the next generation of strategies. This corresponds to the concept of "survival of the fittest" in nature. Offspring is then formed by pairwise recombination of the parents. Finally, the offspring strategies are slightly changed, with a small probability and the new population replaces the old one. Recombination is usually regarded as information exchange between strategies, while mutation can be seen as error or innovation. Figure 1.4 gives an example of the mutation operator. Usually the chance of mutation is specified by the **mutation rate** per bit, that is each bit on the chromosome has a small chance of being mutated. Although the settings for the mutation rate depend on the particular problem that is studied, a general guideline (Bäck *et al.* 1997*b*) is that the mutation rate per bit can be taken as 1/(length of the chromosome).

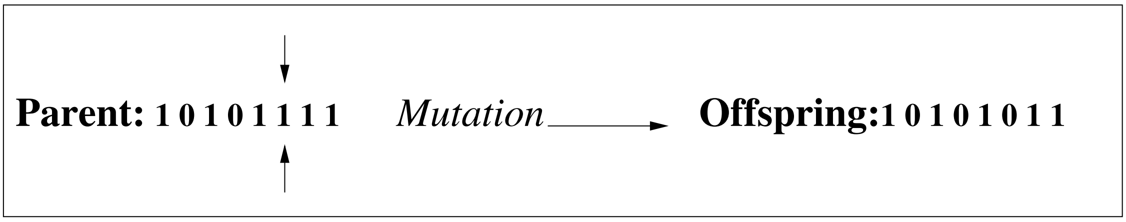

Figure 1.4: Example of mutation.

The recombination operator is also called crossover operator. The three most common crossover operators are one-point crossover, two-point crossover and uniform crossover. With one-point crossover a randomly selected point of the chromosome is chosen as the *crossover point*. This *crossover point* divides the parent chromosomes into two parts. The two offspring chromosomes are then created by recombining the individual parts of the parent chromosomes, as demonstrated in Figure 1.5. Two-point crossover works essentially the same, except that two crossover points are selected. Finally, with uniform crossover, for each bit-position it is randomly decided whether the first offspring inherits the bit value of the first or the second parent. The second offspring then inherits the bit value of the other parent.

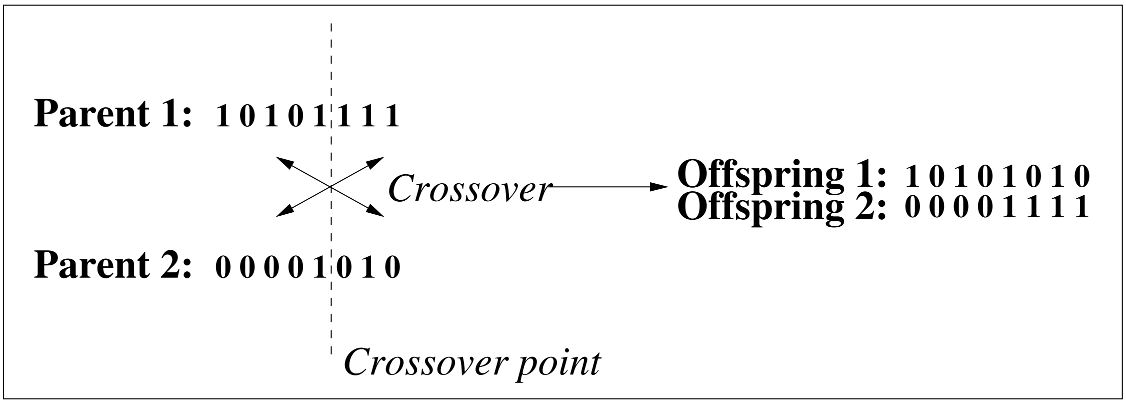

Figure 1.5: Example of one-point crossover.

1.3.1 Evolutionary algorithms for modeling Agent Behavior

Originally, evolutionary algorithms were designed for and used in optimization problems. Until recently, most applications of evolutionary computation to economics were in optimization and operations research, see Biethahn and (Eds.) (1995) for examples. Fudenberg and Levine (1998*a*) suggested that techniques from the field of artificial intelligence should be used to design more accurate models of boundedly rational learning.

Since the pioneering work by Axelrod (1984) and Miller (1986), evolutionary algorithms are increasingly used in economics, see for example (Arifovic 1994, Arifovic 1996, Arifovic 1998, Arifovic and Eaton 1995, Chen and Yeh 1997, Bullard 1999) . More specifically, in this type of research, evolutionary algorithms are used to model the behavior and interaction of heterogeneous boundedly rational agents. The aim of such *evolutionary simulations* is not to develop isolated solutions to a single problem instance (as in optimization) but rather to gain insight in the dynamics and characteristics of a particular model. Evolutionary simulation can thus be applied to economic problems that are analytically untractable. Within these fields, however, evolutionary algorithms (EAs) can be used for different purposes. One application of evolutionary algorithms considers the evolution of strategies. EAs are then used to evolve good strategies that agents in an (economic) game setting can use, see for example the work of Axelrod (1984) and Beaufills *et al.* (1996) on the Iterated Prisoner's Dilemma game. Other researchers use EAs as a model for human learning and use data from experiments with human subjects to calibrate the EA. Examples are the work of Marks (1998), Dawid (1999) and Arifovic (1994). Another topic that is often considered in ACE research is how differences in the learning mechanism may influence the outcome of the system. The work of Marks (1992) and Vriend (2000) on oligopoly markets provide examples of this strand of literature.

In these evolutionary simulations, the genetic population is regarded as a pool of strategies that can be used by the learning agents. Agents then interact with each other (through a market or game) or with the environment and they obtain a payoff. This payoff determines the fitness of the strategy that was used by the agent and strategies are updated by the genetic algorithm. Economic evolutionary simulations thus differ from evolutionary optimization in the sense that there is no global absolute fitness function, but fitness is determined through interaction. Most evolutionary simulations are based either on individual learning (e.g., Classifier Systems) where agents learn based on their own previous experience or on population learning, where agents learn based on the experiences of others. Which type of learning is more suitable to model a specific problem depends on the economic model that is used. In a population learning model agents can adapt parts of the strategies of other agents through recombination which means that (parts of) strategies are visible to other agents. Whether this is realistic depends on the economic problem that is studied since economic agents (firms, bargaining individuals) often make great efforts to hide their strategies from their competitors.

Furthermore, the use of evolutionary algorithms also presents some problems for the modeler in terms of interpretation and model settings. Technical parameter settings are often chosen from an economic perspective neglecting known issues and guidelines from computer science (Bäck *et al.* 1997*b*, Jong 1975, Grefenstette 1986, Bäck *et al.* 1997*a*). Several authors have addressed the issue of interpretation of evolutionary algorithms in economics (Edmonds 2001, Chattoe 1998, Vriend 2000, Klos 1999). In addition, attempts were made to unify the theories of natural selection and economics (Knudsen 2002, Hodgson 2002, Weibull 2002, Alchian 1950, Nelson and Winter 1982, Blume and Easley 1993). Also, several papers appeared that unify simple genetic algorithms and economic theory (Arifovic forthcoming, Riechmann 2001*a*, Dawid 1999). In Chapter 2 we discuss these modeling and interpretation issues in more detail.

1.4 Outline of the thesis

As described above, Evolutionary Agent-Based Computational Economics differs from traditional economics in two important aspects. Agents are boundedly rational instead of perfectly rational and interaction is not global and not anonymous. These two factors form the basis of this thesis. In subsequent chapters we aim to include these aspects in our models. First we investigate the behavior of boundedly rational agents and the effects of (different types of) interaction in existing economic models, and then we address new models concerning internet markets and information goods. Chapter 2 is a methodological chapter addressing the design and implementation of evolutionary agent-based economic simulations using an economic textbook example, the Cournot game. In the subsequent chapters the insights provided in Chapter 2 function as a basis for designing robust and valid simulations. In Chapter 3, we study the behavior of different types of boundedly rational agents. Again we study the Cournot model since this allows us to relate our results to theoretical outcomes. In Chapter 4 we consider the effects of selective interaction on agent-behaviour and economic outcomes. Chapter 5 investigates the role of intermediaries in an electronic trade network; we focus on the question whether intermediaries can survive in electronic markets where consumers can also contact producers directly. Related to this subject, Chapter 6 is concerned with the spread of information on a social network. More specifically, we study whether firms can learn information about the social network and use this information to design better advertising strategies. Finally, in Chapter 7 we give some concluding remarks. Below we give a more detailed overview of the individual chapters.

Chapter 2 is concerned with the proper design and implementation of evolutionary economic simulations. Agent-based Computational Economics combines elements from economics and computer science. In this chapter, we focus on the relation between the evolutionary technique that is used and the economic problem that is modeled. Current economic simulations often derive parameter settings for the genetic algorithm directly from the values of the economic model parameters. In this chapter we show that this practice may hinder the performance of the GA and thereby hinder agent learning. More specifically, we show that economic model parameters and evolutionary algorithm parameters should be treated separately by comparing two widely used approaches to population learning with respect to their convergence properties and robustness.

Chapter 3 is also concerned with the Cournot model. In this chapter we present an individual learning model which allows us to compare different types of agents with different levels of rationality that are interacting in the market. A Cournot duopoly market modeled as a co-evolving system of autonomous interacting agents is investigated. We present results for different types of boundedly rational agents. Agent types differ both in the complexity of their strategies and the information they have available to make their decision. Some types of agents use very simple strategies to make a production decision, while other types use a quite sophisticated decision rule. All agents types are tested in a round robin tournament. We consider the evolutionary stability of the evolving populations, especially with respect to the different equilibria of the Cournot game. Furthermore, we investigate the performance of the different agent types under changing market conditions.

In **Chapter 4** we return to a population learning model but the focus of this chapter is on the structure of the interaction and the influence of the recombination operator on the outcomes of the simulation. The evolution of cooperation in a system of agents playing the Iterated Prisoner's Dilemma (IPD) is investigated. We present results for the standard two-person IPD as well as the more general N-person IPD (NIPD) game. In our computational model, agents have visible tags and choose whether to interact or not based upon these tags. We consider the evolutionary stability of the evolving populations. We extend previous work by introducing sexual reproduction (recombination) of strategies and by analyzing its influence on the evolving populations. We observed the occasional formation of very stable cooperative societies, as opposed to previous results without sexual reproduction. These cooperative societies are able to resist invasions of "mimics" (defecting agents with the tag of a cooperating agent).

Chapter 5 builds on the techniques presented in the previous chapters to address the question whether intermediaries will still exist and be able to make a profit if consumers can make direct connections with producers, as is often the case in electronic commerce. We have performed agent-based simulations to study the performance of intermediaries under different market conditions. We modeled an electronic trade network where an information good is traded over the network. Each trade period cost-minimizing consumers have to decide which links to form to sellers (i.e., producers and intermediaries); the good can only be purchased if a link between the buyer and the seller exists. Links thus represent trading possibilities or a flow of relevant information (i.e., price quotes between potential buyers and sellers in our case). The consumers in the model have to make a strategic decision about which (costly) links to form. We have used an evolutionary algorithm to model the search and learning behavior of the buyers. Our main finding is that if market dynamics are sufficiently complex, intermediaries that have better knowledge about the market than the average consumer are initially able to increase their market share and make a profit.

Chapter 6 investigates the spread of information on a social network. The network consists of agents that are exposed to the introduction of a new product. Consumers decide whether or not to buy the product based on their own preferences and the decisions of their neighbors in the social network. We have used and extended concepts from the literature on epidemics and herd behavior to study this problem. The central question of this chapter is whether firms can learn about the network structure and consumer characteristics when only limited information is available, and use this information to evolve a successful directed advertising strategy. In order to do so, we have extended existing models to allow for heterogeneous agents and positive as well as negative externalities. The firm can learn a directed advertising strategy that takes into account both the topology of the social consumer network and the characteristics of the consumer. Such directed advertising strategies outperform random advertising.

1.5 Publications

Chapter 2 appeared as:
F.Alkemade, H.M Amman and J.A. La Poutré. The Separation of Economic versus EA parameters in EA-Learning. *CWI Technical report*

Chapter 3 previously appeared as:
F. Alkemade and J.A. La Poutré (2003). Heterogeneous, Boundedly Rational Agents in the Cournot Duopoly. In *Heterogenous Agents, Interactions and Economic Performance* (R. Cowan and N. Jonard, Ed.). Lecture Notes

in Economics and Mathematical Systems **521**. pp. 3-17. Springer Verlag.

A preliminary version of Chapter 3 was published as:
F.Alkemade and J.A. La Poutré (2002). Boundedly Rational Agents Achieving Collusive Outcomes in the Cournot Game. In: *Proceedings of the Sixth Joint Conference on Information Sciences* (P.P. Wang, Ed.). Association for Intelligent Machinery. pp. 1143-1146.

Chapter 4 appeared as:
F. Alkemade, D.D.B. van Bragt and J.A. La Poutré (2004). Stabilization of Tag-Mediated Interaction by Sexual Reproduction in an Evolutionary Agent System. *Journal of Information Sciences*, in Press.

A preliminary version of Chapter 4 was published as:
F. Alkemade, D.D.B. van Bragt and J.A. La Poutré (2000). Stabilization of Tag-Mediated Interaction by Sexual Reproduction in an Evolutionary Agent System. In: *Proceedings of the Fifth Joint Conference on Information Sciences* (P.P. Wang, Ed.). Association for Intelligent Machinery. pp. 945-949.

Chapter 5 was published as:
F. Alkemade, J.A. La Poutré and H.M. Amman (2003). An Agent-Based Evolutionary Trade Network Simulation. In: *Innovations in Financial and Economic Networks* (A. Nagurney Ed.). Edward Elgar Publishers. pp. 237-252.

A short version of Chapter 5 appeared as:
F. Alkemade, H.M. Amman and J.A. La Poutré (2003). Intermediaries in an Electronic Trade Network. In: *Proceedings of the 4th ACM Conference on Electronic Commerce (EC-03).* ACM Press. pp 200-201.

Chapter 6 appeared as:
F. Alkemade and C. Castaldi (2004). Diffusion of Information on a Social Network. *CWI Technical Report*

Chapter 2

Robust Evolutionary Algorithm Design for Economic Modeling

2.1 Introduction

Agent-based Computational Economics (ACE) concerns the computational study of economies modeled as evolving systems of autonomous interacting agents (Tesfatsion 2001). The ACE approach combines elements from evolutionary economics, cognitive science and computer science. Evolution-inspired algorithms, such as genetic algorithms are increasingly used in ACE (Arifovic 1994, Dawid 1999, Arthur 1993, Chen and Yeh 2001, Tsang and Li 2000). Evolutionary algorithms are biologically inspired techniques that use the concept of "survival of the fittest" to evolve agent behaviour that becomes better and better adapted to the environment (i.e., a particular market). Evolutionary algorithms thus provide us with a way to model boundedly rational agent learning and decision making. The popularity of evolutionary algorithms in economic simulations can be attributed to the fact that they allow us to model large systems of boundedly rational agents from the bottom up. Evolutionary algorithms were originally designed for and used in optimization problems and there is a vast literature that describes how to tune the algorithm to perform well in optimization problems (Jong 1975, Grefenstette 1986, Bäck *et al.* 1997*a*, Tuson and Ross 1998). Now that evolutionary algorithms are also used in the social simulation domain these guidelines have to be reestablished. Current economic simulations often derive parameter settings for the genetic algorithm directly from the values of the economic model parameters. In this chapter we show that this practice may hinder the performance of the GA and thereby hinder agent learning. Furthermore, by comparing two widely used approaches to evolutionary algorithm learning, we show that, in order to obtain robust results, economic model parameters and evolutionary algorithm parameters should be treated separately.

In the first approach, the evolutionary population is considered as a population of agents (one chromosome equals one agent) whereas in the second approach the evolutionary population is considered as a population of strategies from which the agents can choose. We show that the first approach (direct evolutionary interpretation of the economic model) may lead to premature convergence of the genetic algorithm. Robustness to variations in the parameter settings is particularly important in social simulation since it is often difficult to give a direct economic interpretation of a particular evolutionary algorithm parameter (such as population size or recombination rate). This makes it impossible to interpret the economic meaning of a simulation outcome that changes when a technical parameter is slightly changed. Therefore, in order to derive any results from evolutionary social simulation it is important that results are valid for a wider range of parameter settings, that is that they are robust. In this chapter, the focus is on robustness with respect to the technical evolutionary algorithm parameters. The chapter proceeds by giving a general overview of genetic algorithms in Section 2.2 and a discussion of the interpretation of both genetic algorithms models for the learning behavior of economic agents in Section 2.3. Section 2.4 describes the different experiments that were performed and results are given in Section 2.5. Finally, conclusions are given in Section 2.6.

2.2 Evolutionary algorithms

An evolutionary algorithm (EA) is a technique that uses the concept of "survival of the fittest" to evolve a population of strategies (Holland 1975, Mitchell 1996). Using an EA, strategies are represented as chromosomes and the chromosomes evolve from generation to generation yielding better and better strategies. A typical EA can be described as follows (Mitchell 1996). First, a population of randomly initialized chromosomes is generated. The population is subsequently changed and improved in a number of generations by means of selection, recombination ("crossover"), and mutation. Selection chooses the better chromosomes (with higher accumulated payoff) that will serve as parents for the next generation of strategies. This corresponds to the concept of "survival of the fittest" in nature. Offspring is then formed by pairwise recombination of the parents. Finally, the offspring chromosomes are slightly changed (with a small probability) and the new population replaces the old one.

Thus, an EA has several parameters, being among others the mutation rate (the probability for a gene or bit in a chromosome to change), the population size (the number of chromosomes in the population), the recombination rate (the percentage of chromosomes that will be subject to recombination) and the representation of the chromosome. The values of these parameters have to be carefully chosen in order for the EA to perform well. Although there are no conclusive results on which parameter settings allow effective EA performance, there are some general guidelines available, see for examples (Mitchell 1996, Bäck *et al.* 1997*b*).

2.2.1 Parameter issues

Particularly important in this chapter is the avoidance of premature convergence by allowing the population diversity of the GA to be sufficient, that is a sufficient number of different elements (chromosomes) must be present in the population to avoid sampling errors due to small population size. This is a merely technical condition, under which the highly stylized EA operators (mutation, recombination, selection) and parameters operate properly. Premature convergence restricts the EA in its learning capabilities. This occurs when, early in the search process, the EA focuses on the exploitation of a small selection of rather fit strategies at the expense of the exploration of other regions of the search space. In optimization problems premature convergence causes the EA to get stuck in various local optima (in different runs), while lacking the diversity to explore other regions of the search space and find the global optimum. To preserve sufficient diversity and to avoid premature convergence, the values of the EA-parameters have to be carefully chosen. Known factors that influence the diversity of the population (and thus the occurrence of premature convergence) are among others mutation rate, population size, selection pressure and the chosen chromosome representation (Bäck *et al.* 1997*a*). The length and representation of the chromosome affect convergence. The representation that is chosen should be able to represent all possible strategy values. Furthermore, we consider it good practice to choose chromosome length and representation in such a way that the average outcome of randomly initialized population of the chromosomes does not lie on or close to equilibrium values of the economic model.

2.3 Economic interpretation of genetic algorithms

Two general approaches to social or population learning can be distinguished and each approach represents a different economic interpretation of genetic algorithm learning. In the first approach each agent is represented by a single chromosome in the EA population as in Dawid (1999) and Vriend (2000). In this approach the number of chromosomes equals the number of economic agents. In the second approach the population of chromosomes and the economic model are separated, as in Axelrod (1987) and Riechmann (2001*b*). The population of chromosomes is regarded as a pool of strategies from which the agents can choose. Both approaches consider the recombination operator as a model of information exchange between two agents or strategies, whereas the mutation operator is regarded as a model for error or innovation. Although there thus exists an economic interpretation of the genetic operators the modeler should be very careful in interpreting the economic meaning of simulation results. It is very difficult to attach an economic meaning to social simulation results that were obtained using specific values for the GA parameters, in particular if outcomes change considerably with small changes of the parameters. In general, we propose that in order to make valid economic interpretations based on evolutionary simulations - results should be robust, that is valid for a larger range of parameter settings.

Several authors have commented on the economic interpretation of genetic algorithms (Chattoe 1998, Dawid 1999, Edmonds 2001, Klos 1999). A commonly heard objection against this interpretation is that in population based evolutionary algorithms, strategies are shared by all agents. This means that agents have direct insight into which strategies were used by others. Furthermore, agents can copy (parts of) strategies from (well performing) others by means of selection and recombination. This is not true for many social systems; economic agents (firms, bargaining individuals), for example, often make great efforts to hide their strategies from their competitors. This suggests that the use of GAs as a model for agent learning is only valid when the strategies that are considered are simple actions that are observable in the market. The second modeling approach (chromosome equals a single strategy) is therefore preferred to the chromosome-equals-agent approach of population learning. Below we will illustrate this by showing the robustness results obtained for a large range of values.

2.4 Experimental setup

This section describes the experimental setup that was used to investigate the effect of evolutionary model parameter settings on economic outcomes. As described above two main approaches can be distinguished (see Table 2.1). Using approach I (Dawid 1999, Vriend 2000), each agent is represented by a single strategy in the genetic population. The second approach allows an agent to pick a strategy from a large population of strategies. In approach I the value of the genetic algorithm parameter *population size* (that is the number of strategies in the genetic pool) is directly determined by the economic model. In approach II this is not the case. Using approach II (Axelrod 1987, Riechmann 2001*b*), we have to set the value for the population size in such a way that the economic outcomes are robust to changes of the parameter values (both economic and genetic).

2.4.1 The Cournot oligopoly

The testbed for our simulations is a textbook (Varian 1992, Cournot 1897) Cournot oligopoly market with 4 players. We develop an evolutionary sim-

Approach I	**Approach II**
Chromosome = Agent	Chromosome = Strategy
Population size determined by number of economic agents	*Population size determined by proper EA learning*

Table 2.1: Summarizing the two evolutionary learning approaches.

ulation of this model and use it to compare both approaches. The rest of this section will describe the Cournot oligopoly and the setup of the evolutionary simulation. We use a population learning evolutionary algorithm to model the search and learning behaviour of economic agents in a Cournot oligopoly game with four agents. The Cournot oligopoly market describes a situation where a few firms compete in a single market. The firms have some market influence through a common price demand curve. In this situation, firms have to make a strategic decision, taking into account the decisions of the other firms. Here we focus on the Cournot oligopoly which provides a model for the market when the firms produce a homogeneous good and compete on quantity q. The 4 firm oligopoly model we use is characterized by the following equations:

$$\begin{aligned} &\textit{Market price}: \; P = 256 - Q, \textit{where } Q = \textstyle\sum_{i=1}^{4} q_i \\ &\textit{Profit firm } i: \; \pi_i = Pq_i - 56q_i \end{aligned}$$

The Cournot-Nash equilibrium occurs when each firm's output is a best response to the combined output of the other firms—at output q=40. When firms produce at the competitive outcome, each produces an output of on average 50 and the firms make no profit. Profits are highest when the 4 firms act as a single monopolist and produce the collusive output of on average 25 per firm. Collusion usually does not occur in the one-shot game.

A general outline of the Cournot oligopoly simulation is given below. Each period an agent draws a strategy from the population of strategies. If approach I is used the number of strategies is equal to the number of agents (4 in this chapter). Using approach II we have to find a suitable population size (the number of strategies/chromosomes in the genetic population). In order to do this we incrementally (steps of 4) increase the population size and run the system repeatedly with a large number of values for the population sizes. Another parameter that influences simulation results is the representation of the strategies, in particular the length of the strategy. Longer strategies represent larger possible output values. Initializing the population with a representation that allows agents to produce quantities that greatly exceed demand can be done if the modeler does not want the agents to have any a priori knowledge about the market (everything has to be learned). Again we look for settings that render robust results.

Outline of the simulation model:	
Step 1	**Select strategy:**
	Each agent randomly draws a strategy from the population of strategies
Step 2	**Determine action**
Step 3	**Play game**
Step 4	**Assign payoff**
	The fitness of a strategy equals the payoff gained by the agent using the strategy
Step 5	**If all strategies have been used:**
	Update strategies using EA

Table 2.2 gives an overview of the parameter settings used in the experiments. The smallest population size tested was 4 (i.e, equal to the number of agents - Approach I) and the largest population size tested 200. The chromosome length was varied between 6 (maximum output 64) and 9 (maximum output 512). In all cases the average output of a randomly initialized population does not lie on the Cournot-Nash or the competitive equilibrium. The structure of the initial population is determined by the *initial density* parameter. A population that is initialized with a density of 0.5 (the default value), fifty percent of the bits is set to 1 on average. The values for the mutation rate are within the range of commonly recommended values (Grefenstette 1986). We consider the convergence of the model with respect to the different outcomes of the game for a large range of values of the size of the population of strategies. Each period of the game

agents select a strategy from the population of strategies. This population is subsequently updated by a genetic algorithm. The general outline of the simulation model is described below. A genetic algorithm is said to converge if almost all strategies in the population are identical. Note that there can be no convergence in the mathematical sense, for the mutations disrupt the uniform state over and over again.

	I	**II**
Economic Model Parameters		
Number of Agents	4	4
EA Parameters		
Crossover Rate	1.0	1.0
Mutation Rate (per bit)	0.01	0.01
Population Size	**4**	**4-200** (increments 4)
Chromosome Length	6–9	6–9
Representation	binary	binary
Initial Density	0.1, 0.5, 0.9	0.1, 0.5, 0.9

Table 2.2: Parameter settings used in the Cournot oligopoly experiments. Approach I is the methodology where a chromosome equals an agent whereas in approach II a chromosome equals a strategy.

2.5 Robustness results

This section describes the experiments we have performed comparing approach I and II to social learning. We are especially interested in the robustness properties of the two approaches. Apart from looking at the population size we study the behaviour of both approaches under different initial conditions and different representations. In all simulations we have used a simple GA with mutation rate 0.01 and crossover rate 1.0. First we study the effect of population size on the outcome of the Cournot oligopoly game. We start by initializing the population size at 4 (approach I) and then increasing this value (with steps of size 4) to determine a population size that yields robust results. Figure 2.1 shows the results of our experiments for different representations (chromosome lengths). The first figure (left) represents the relationship between average output and population size. Each point in the graph represents the average over 20 runs of 400 generations each (generation 100-500 to compensate for initial noise effects) that were executed using each value of the population size. The initial population density is 0.5 in these experiments, that is, initially half of the bits is set to one. The second figure (right) gives the standard deviation of average output over those 20 runs for each tested value of the population size. Results are given for different representations of the strategies, that is chromosome length 6,7,8 and 9.

2.5.1 Aggregate results

The outcomes for population size 4 correspond to approach I and we can deduce from the figure that population sizes of around 100 chromosomes would be suitable for approach II. Notice that for small values of the population size the difference between individual runs is very large (high standard deviation) whereas averages converge towards the Cournot-Nash value for larger values of the population size. Convergence is slower for longer chromosome length where genetic drift causes averages to go up (as chromosome length increases, a random mutation is more likely to lead to a large increase in production than with smaller chromosome lengths). Note that simulations using different representations all converge to the Cournot-Nash equilibrium with small standard deviation for sufficiently large population sizes. Furthermore, Figure 2.1 suggests that convergence is not robust with respect to representation for small population sizes.
To gain more insight in the convergence behaviour of both approaches we take a look at the average population behaviour using approach I and II. Figure 2.2 shows the average population behaviour for different representations and different initial conditions over 500 generations. All tested chromosome lengths allow the agents to produce at both the Cournot-Nash and competitive output levels and therefore do not constrain their production capacity in any way. The results show that approach I is not robust with respect to representation, that is different representations yield different economic outcomes. On the other hand population averages do converge towards Cournot-Nash outputs irrespective of representation length if approach II

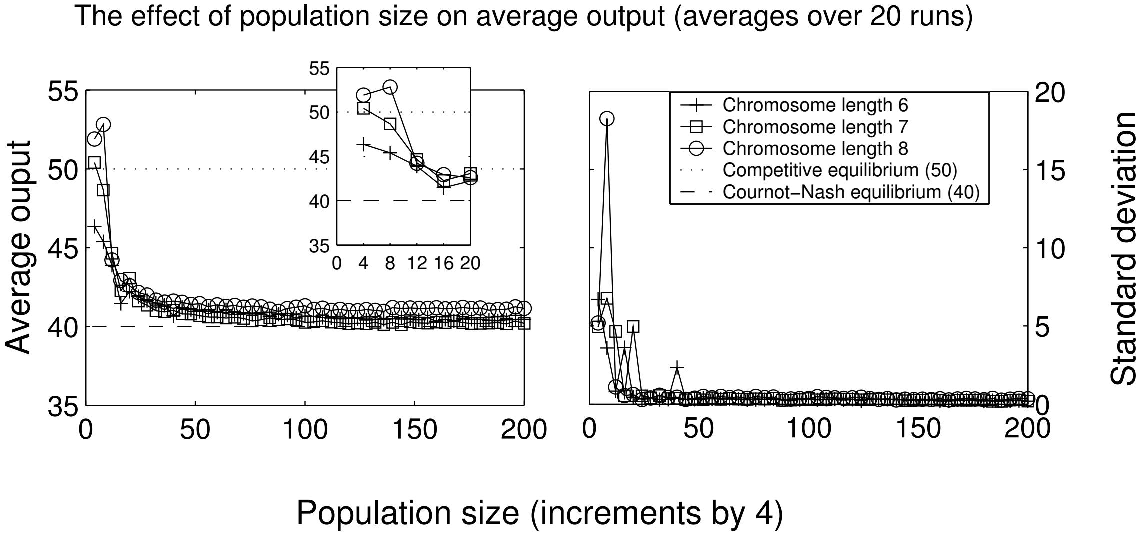

Figure 2.1: Average output versus population size, averages (left) and standard deviations (right) over 20 runs. Each plotted value represents the average output over 20 runs and over the last 400 generations (generation 100 to 500) to eliminate initial noise effects.

is used. If we look at convergence behaviour with respect to initial population density we see again that approach I does not yield robust results for all representations. Only for chromosome length 7 we observe convergence towards the competitive equilibrium for all tested initial conditions. We will take a closer look at the individual runs in the next section to futher study the convergence behaviour. The outcomes in Figure 2.2 suggest that the lack of robustness is caused by premature convergence caused by a lack of diversity in the population. Very small (0.1) or very large (0.9) initial densities lead to an almost uniform population and this effect is of course strongest for small population sizes. While some learning towards profitable outcomes occurs under these conditions, the population converges before a good outcome is reached. Indeed we see that outcomes for different representations show most agreement for initial density 0.5 when population diversity is maximized.

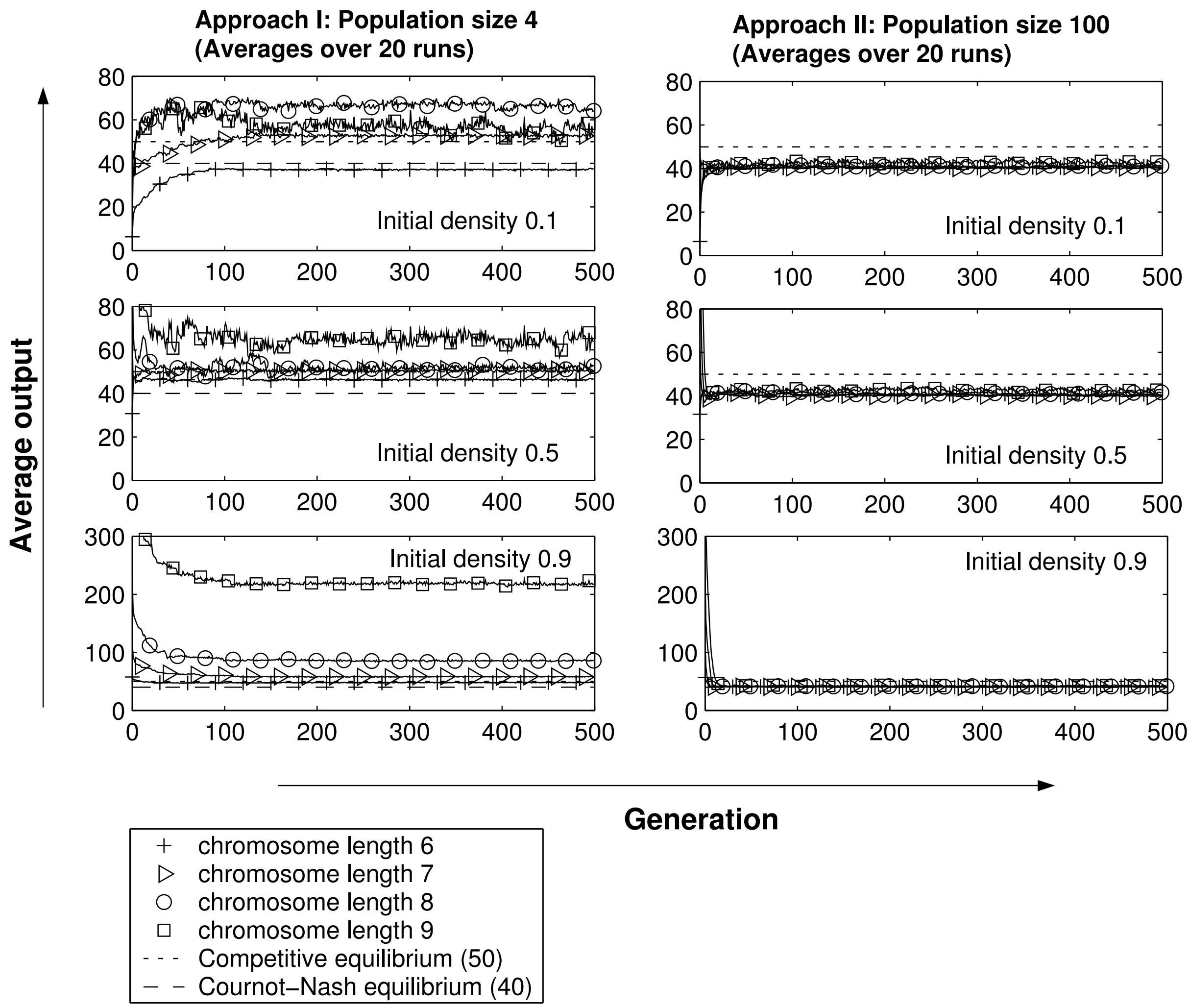

Figure 2.2: Average output for different population sizes - averages over 20 runs. Approach I: population size 4, Approach II population size 100. Note that the y-axis is larger for initial density 0.9 to accomodate all outcomes.

2.5.2 Individual runs

Figures 2.3 through 2.6 show the twenty individual runs for each approach. Notice that there is no uniform convergence when approach I is used. More specifically, if we look at the results for chromosome length 7 in Figure 2.4 (bottom half) we see that the apparent convergence to the competitive equilibrium is only present at the aggregate level. Furthermore, we see that individual runs sometimes converge to very large output values (Figure 2.6) leading to losses for the agents. Here learning stops before profitable outcomes are reached. If approach II is used, however we see that all individual runs converge towards the Cournot-Nash level where each agent's output is a best response to the aggregate output of the other agents. The agents obtain higher profits when their aggregate output is at the Cournot-Nash level than at the competitive level. These results show that a significantly large population of chromosomes is needed to prevent premature convergence to a less profitable outcome. Population learning genetic algorithms can thus be used to model boundedly rational agents in the Cournot oligopoly game. Evolutionary algorithm learning is a sufficiently powerful learning technique to arrive at the Cournot-Nash outcome in a population learning setting. When GA learning is inhibited by small population sizes, the learning outcome depends strongly on the initial population and GA learning is reduced to imitation based learning. We have performed some experiments to illustrate this effect in the next section.

2.5.3 From imitation based learning to GA learning

The evolutionary algorithm is built using selection, recombination and mutation. If we look at each of these components separately, we can gain further insight into the effects of population diversity. An algorithm using only selection (imitation) can be compared to replicator dynamics a simple learning model that is often used in economics (Fudenberg and Levine 1998*b*, Weibull 2002, Mailath 1992). The difference between replicator dynamics and genetic algorithm learning is that replicator dynamics is solely based on imitation while innovation through recombination and mutation are integral parts of genetic algorithm performance. Another difference between genetic algorithms and replicator dynamics is the way fitnesses are compared. In replicator dynamics, agents can imitate strategies used by a direct opponent while in a GA, strategies used in all games are compared population wide. Theoretical results using replicator dynamics in oligopoly settings show that there is convergence towards the competitive equilibrium in the long run as long as the competitive outcome is present in the initial population (Vega-Redondo 1997). Replicator dynamics is based on imitation of more successful strategies. To illustrate the learning capabilities of

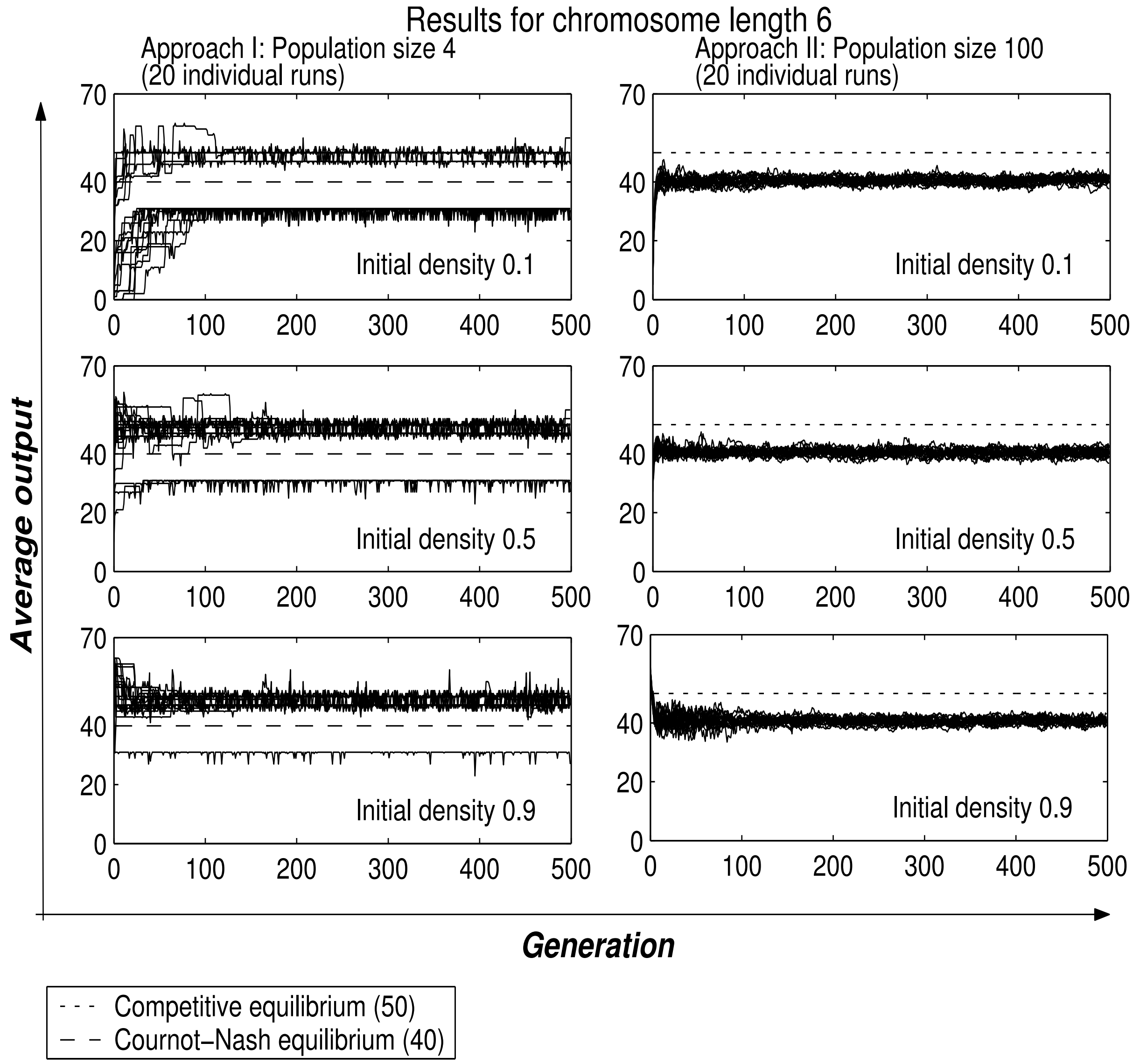

Figure 2.3: Individual runs for chromosome length 6. Note that not all axes are equal in order to accommodate all results.

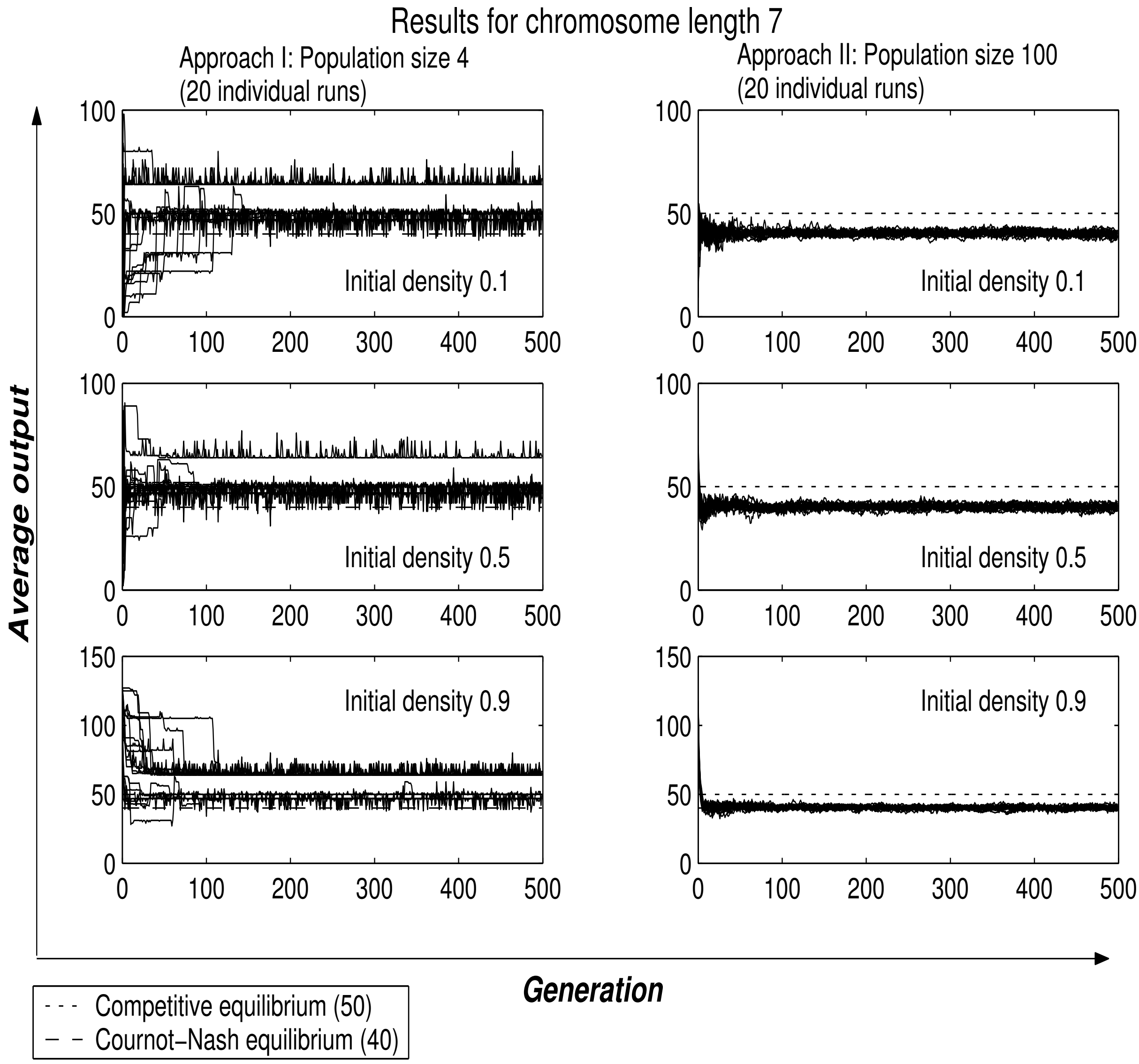

Figure 2.4: Individual runs for chromosome length 7. Note that not all axes are equal in order to accommodate all results.

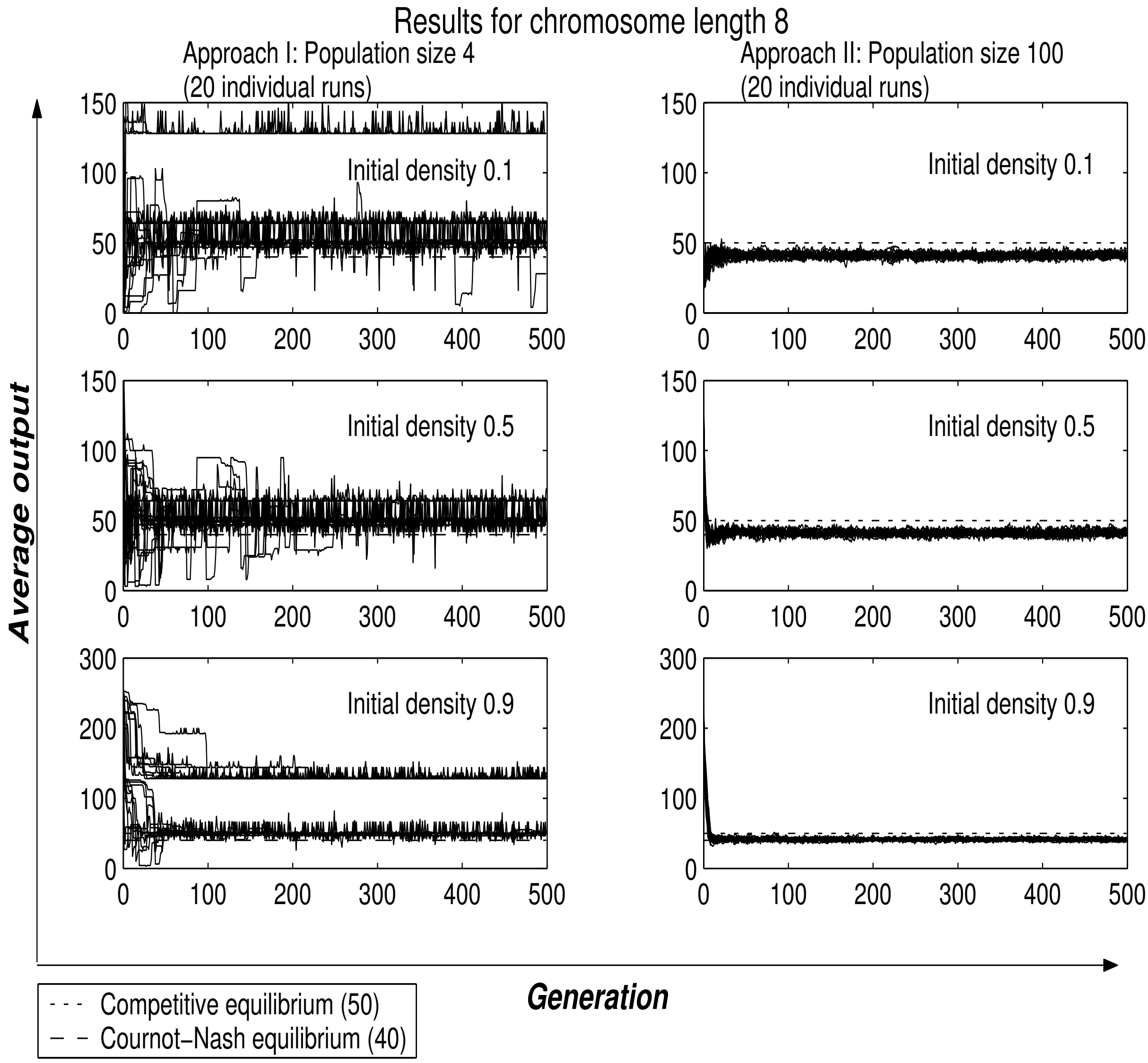

Figure 2.5: Individual runs for chromosome length 8. Note that not all axes are equal in order to accommodate all outcomes.

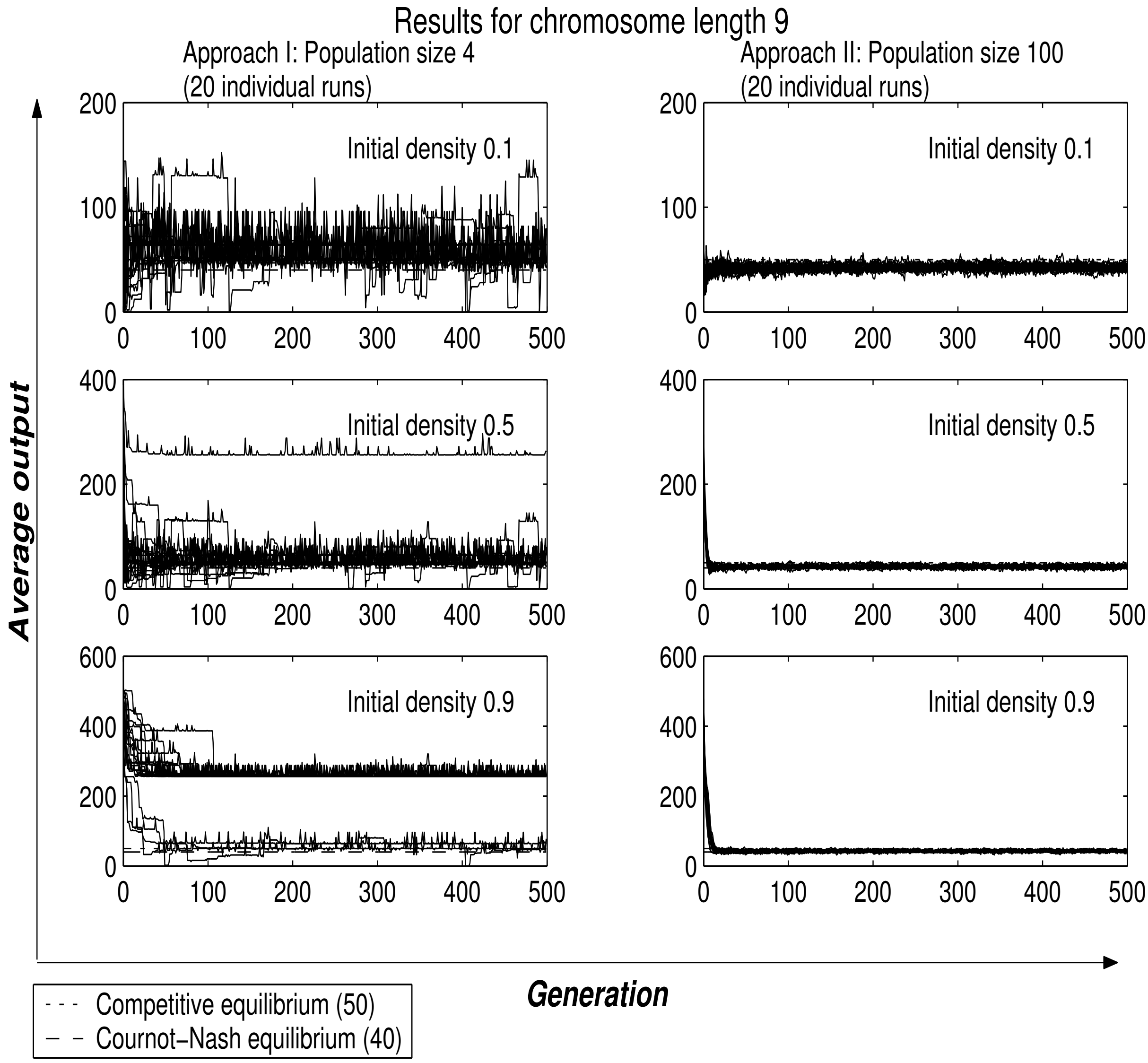

Figure 2.6: Individual runs for chromosome length 9. Note that not all axes are equal in order to accommodate all outcomes.

a properly set up GA, we have extended a replicator dynamics version of the Cournot oligopoly with recombination and mutation. We have used the method of fitness comparison used in replicator dynamics (only imitate strategies of direct opponents). Furthermore, we have inserted the competitive outcomes in the initial population, as in Vega-Redondo (1997) . These experiments can give us some insight in how the recombination and mutation operator of the GA are influenced by small population sizes. Averages over 20 runs are given in Figures 2.7 and 2.8. Results are shown for only selection (imitation) and imitation plus recombination and mutation (mutation rate 0.1) for chromosome length 7. We see that the convergence behaviour of the imitation based learner is less influenced by the population size than that of the GA. Using a sufficiently large population size, GA convergence towards the Cournot-Nash outcome can be observed even when a relative fitness measure is used. This shows that approach II is indeed robust with respect to initial conditions, representation and also transformation of the fitness function. This is important since many economic applications are characterized by relative rather than absolute fitness measures.

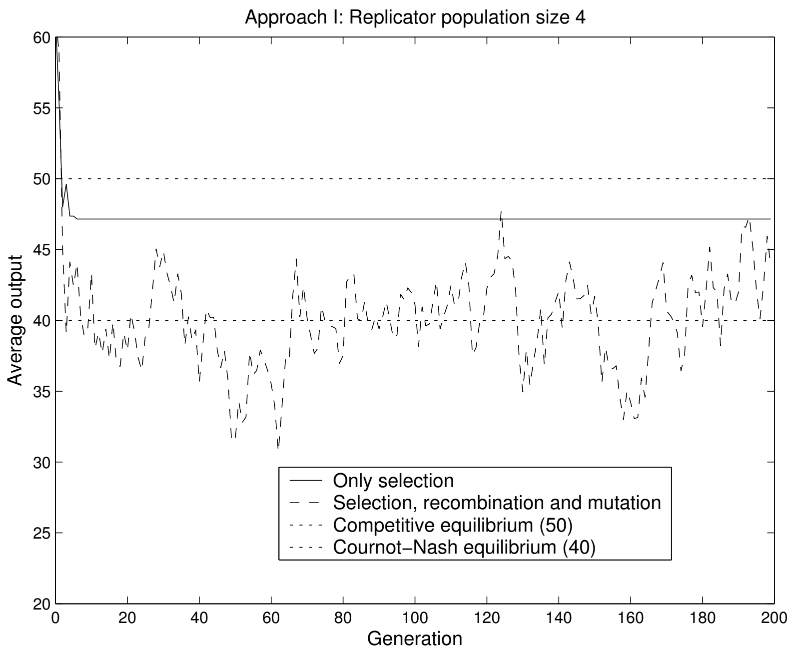

Figure 2.7: The influence of population size on imitation based and GA learning. Averages over 20 runs.

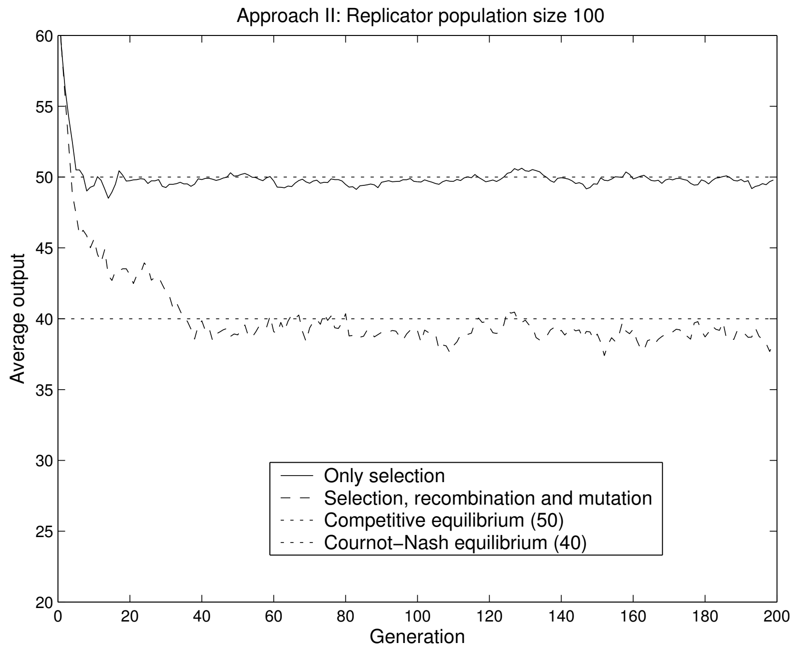

Figure 2.8: The influence of population size on imitation based and GA learning. Averages over 20 runs.

2.6 Conclusions

We have compared two widely used approaches to evolutionary population learning. In the first approach population size is determined by the number of agents in the economic model while in the second approach these economic and technical parameters are treated separately. We show that the first approach can lead to premature convergence, inhibiting the learning capabilities of the agents, related to the abstract learning technique. The second approach leads to robust outcomes with respect to population size, initial density, representation of the strategies and fitness evaluation. These experiments show that economic model parameters and evolutionary algorithm parameters should be treated separately to allow for robust results. Furthermore, the modeler should make sure that simulation outcomes are robust with respect to changes in the experimental setup.

Chapter 3

Heterogeneous, Boundedly Rational Agents

3.1 Introduction

The duopoly market describes a situation where two firms compete in a single market. Both firms influence the market through a common price demand curve. In this situation, firms have to make a strategic decision, taking into account the decision of the other firm. In this chapter we focus again on the Cournot (Cournot 1897) duopoly which provides a model for the market when the two firms produce a homogeneous good and compete on quantity.

In most theoretical models of the Cournot duopoly game, agents are fully rational: they know their own reaction curves, have exact information about the actions of the other firm and take rational decisions. In this chapter we examine different types of so-called *boundedly rational* agents (Simon 1984, Albin 1998, Sargent 1993). These agents do not have access to all the information needed to make a rational decision. Agents have to make a decision based on incomplete and uncertain information such as, for example, their expectations about the behavior of the other agent. Some types of agents use very simple expectation formation techniques, while other agent types exhibit more sophisticated behavior.

We investigate the outcomes the different types of agents will achieve in the repeated Cournot game under both stable and changing market conditions. We are especially interested to see whether agents will achieve the socially desirable competitive market outcome or the Cournot-Nash equilibrium outcome, and under what circumstances the agents learn to sustain the inherently unstable cartel outcome. To model this learning, search and coordination process, we use a genetic algorithm.

The chapter proceeds as follows. Section 3.2 gives a detailed description of the Cournot model. In Section 3.3 a brief overview of the genetic algorithm that is used is given. Also, the evolutionary model used in our experiments will be explained. Section 3.4 will then describe the different types of boundedly rational agents we study in our experiments. Section 3.5 describes the performance of the different types of agents under stable market conditions. Section 3.6 investigates which agents perform best under changing market conditions and conclusions are given in Section 3.7.

3.2 The Cournot duopoly game

The Cournot duopoly is a simple economic model that describes the competition on quantity between two firms, say Firm 1 and Firm 2, see for example Mas-Colell *et al.* (1995). The firms produce a homogeneous good and know the price demand curve. Each firm must decide how much to produce, and the two firms make their production decision at the same time. When making its production decision, a firm takes its competitor into account. The firms know that their competitor is also deciding how much to produce, and the market price they receive depends on the total output of both firms. The essence of the Cournot model is that each firm treats the output level of its competitor as fixed, and then decides how much to produce.

The profit-maximizing output of Firm 1 depends on how much Firm 2 will produce. If Firm 1 thinks Firm 2 will produce nothing, its demand curve is the market demand curve. Firm 1's profit maximizing output is thus a decreasing function of how much it expects Firm 2 to produce. A firm's reaction curve tells it how much to produce, given the production quantity of its competitor. In equilibrium, each firm chooses its production quantity according to its own reaction curve, so the equilibrium output levels occur at the intersection of the two reaction curves. The resulting set of production levels is called a Cournot equilibrium. In this equilibrium, each firm correctly assumes how much its competitor will produce (a fulfilled expectations equilibrium), and it maximizes its profit accordingly. Note that this equilibrium is a Nash equilibrium. In a Nash equilibrium, each firm is doing the best it can given what its competitors are doing. As a result, no firm has any incentive to change its behavior. This equilibrium is also known as a Cournot-Nash equilibrium.

In the one shot game (i.e., one-time decision of the production levels) the firms typically reach the Cournot-Nash equilibrium. However, in a repeated game, when firms interact more than once with each other and thus can try to influence each other's behavior, theoretically other outcomes are also possible. The *collusive* or *cartel* outcome occurs when firms "cooperate" and attempt to set prices and production so as to maximize total industry profits. A cartel is a group of firms that jointly collude to behave like a single monopolist and maximize the sum of their profits. A cartel will typically be unstable in the sense that each firm will be tempted to sell more than its agreed-upon output if it believes that the other firm will not respond.

When firms do not cooperate but each competes for the biggest possible market share, the firms can also end up in the competitive equilibrium, where each price equals marginal costs, and thus the firms make no profits. In a typical Cournot-Nash equilibrium, profits are higher than in a competitive market but lower than in the collusive or cartel situation. The price demand function the firms face in the simulations where market conditions remain stable is given in Equation 3.1. This function changes when performance under changing market conditions is investigated. The (symmetric) Cournot model we use in our experiments is defined by the following equations:

$$\textit{Market price}: \; P = 128 - q_1 - q_2 \tag{3.1}$$

$$\textit{Profit firm 1}: \; \pi_1 = Pq_1 - 56q_1 \tag{3.2}$$

$$\textit{Profit firm 2}: \; \pi_2 = Pq_2 - 56q_2 \tag{3.3}$$

Figure 3.1 depicts the reaction curves for both firms under these market conditions.

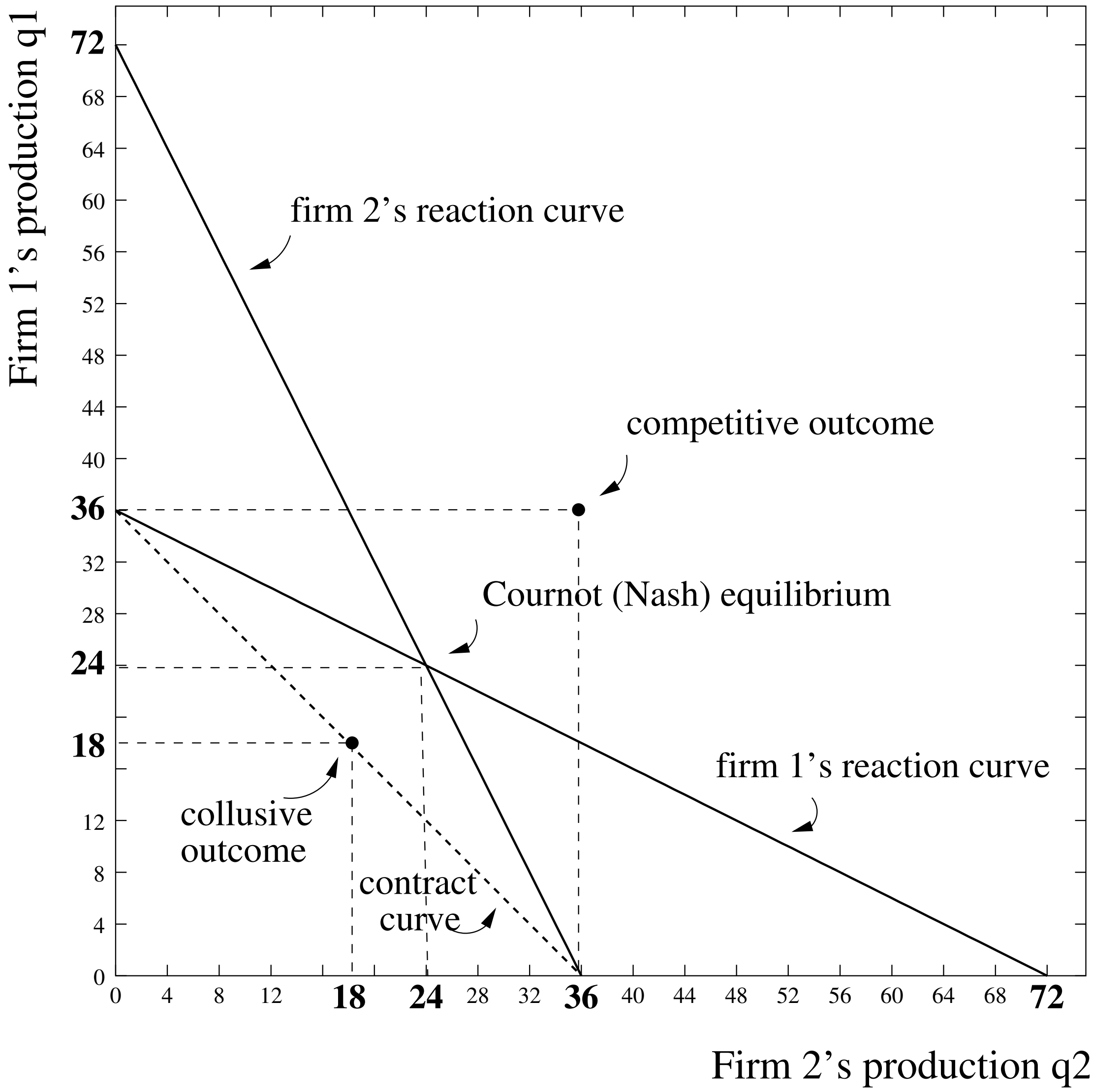

Figure 3.1: The Cournot duopoly.

The reaction curves can be calculated from Equations 3.1, 3.2, and 3.3 by setting marginal revenue equal to marginal costs. To maximize profit, the firms set marginal revenue equal to marginal costs:

$$\textit{Revenue Firm 1}: R_1 = Pq_1 = (128 - q_1 - q_2)q_1 = 128q_1 - q_1^2 - q_1q_2.$$
$$MR_1 = \Delta R_1/\Delta q_1 = 128 - 2q_1 - q_2 \textit{ and } MC_1 = MC_2 = 56.$$

Now setting MR_1 equal to MC_1 and solving for q_1 gives Firm 1's reaction curve:

$$\textit{Firm 1}: \; q_1 = 36 - \frac{1}{2}q_2 \tag{3.4}$$

The same calculation applies to Firm 2 and gives Firm 2's reaction curve:

$$\textit{Firm 2}: \; q_2 = 36 - \frac{1}{2}q_1 \tag{3.5}$$

The Cournot-Nash equilibrium occurs where the reaction curves cross and both firms have an output level of 24. The corresponding equilibrium profits are 576 for each firm. The competitive equilibrium that occurs when price is equal to marginal cost (56 in our model) occurs when both firms set their output to 36, the corresponding profits are then zero for each firm. The profit maximizing collusive output level is reached when total production is equal to 36.

3.3 An evolutionary Cournot duopoly model

Firms in a market learn about market conditions and the nature of their competitors by operating in that market. By searching and learning they find their best strategy. To model this search and learning process, we use a genetic algorithm. Various researchers have used genetic algorithms to simulate the behavior of a population of interacting agents (Dawid 1999, Price 1997). Arifovic (1994), for example, showed that in a simple cobweb model, a genetic algorithm provided a better approximation to experimental data than traditional econometric learning rules such as least squares learning. The genetic algorithm enables us to model agent behaviour as evolving ideas or strategies, whereas the mechanism of "survival of the fittest" allows for the most successful strategies to be maintained and spread throughout the population. Using a genetic algorithm, strategies are represented as chromosomes, and the chromosomes evolve from generation to generation yielding better and better strategies.

Every string in the population can be seen as an economic strategy (in our Cournot duopoly case a *production rule*). The selection operator and, in particular, fitness proportional selection can be seen as a type of learning by imitation. Individuals with a low payoff will imitate the strategies used by successful agents. This imitation process may lead to the spreading of the strings encoding strategies with a high payoff and the vanishing of strings encoding poor strategies. The crossover operator is usually interpreted as a model for communication, or information exchange between strategies. The crossover operator differs from the selection operator in the sense that only part of the strategy of others is imitated. Finally, the mutation operator incorporates the effect of innovation or mistakes of the agents. Individuals change their actions randomly, either by mistake (in copying or imitating due to a lack of information or computational capacity) or because they think the best actions have not been discovered yet.

In our simulation we use a multi-population model; a model of co-evolution. This means that each firm has its own population of strategies (the strategy base) and that a genetic algorithm is separately applied to each population. Strategies then compete only against strategies in the same population. A consequence of the multi-population model is that there is no direct competition between the two firms, and that the firms themselves will not vanish. The multi-population model is graphically depicted in Figure 3.2.

Each generation, a firm selects a strategy from the strategy base associated with that firm. The strategy base thus represents all the knowledge and ideas present in the firm at a particular time. The firm then uses the chosen strategy in the market and a certain payoff is associated with the use of that strategy. After all strategies have been tested, the strategy base is updated by a genetic algorithm, yielding new strategies for the next generation.

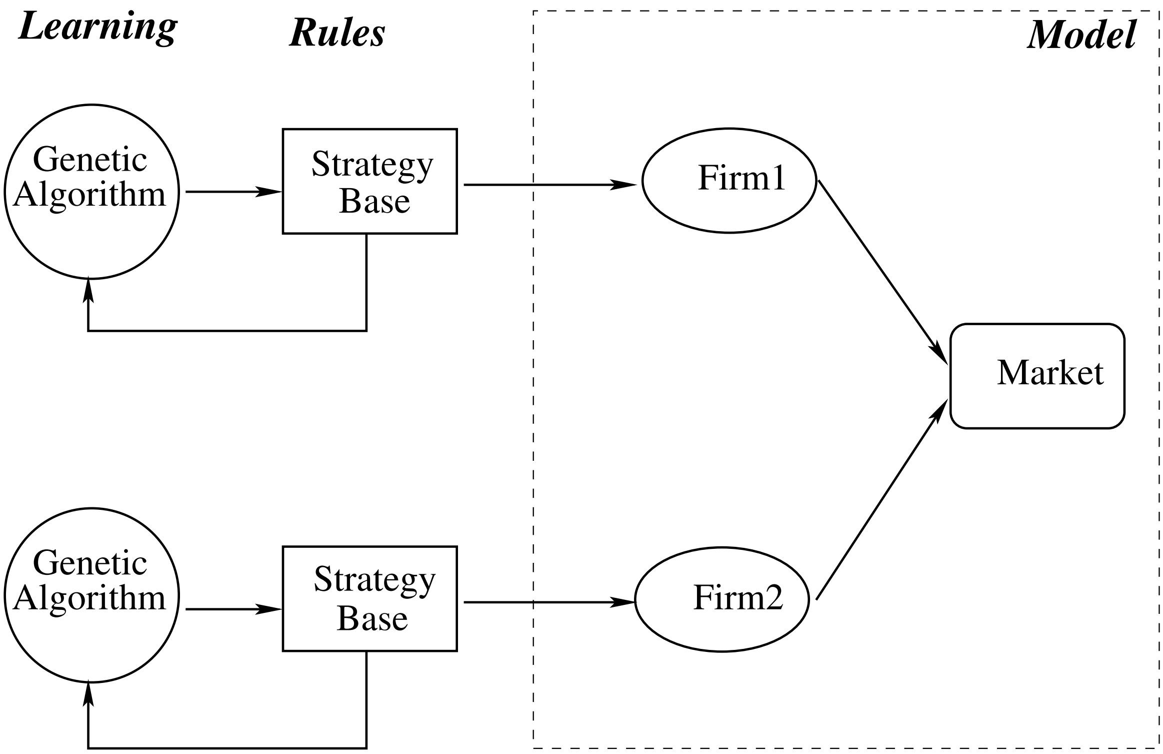

Figure 3.2: The evolutionary model. In each Cournot duopoly game, a chromosome from each firm's strategy base is tested in the market. After each generation (i.e, when all rules have been tested at least once), each firm population is separately updated by the GA.

Table 3.1 gives an overview of the parameter setting for the EA we use in our experiments. The next section describes the different types of agents that were tested in the simulations.

Parameter	*Value*
type EA	GA
selection scheme	fitness proportionate sigma scaling
recombination operator	1pt/2pt/uniform crossover
mutation rate	fixed 0.01...0.1 chromosome length dependent $1/l$ for chromosome length l
crossover probability	1
population size	60
encoding	gray coding
game length	10 iterations
# generations	100
# runs	10

Table 3.1: Evolutionary model parameters.

3.4 Heterogeneous agents

This section describes the different types of boundedly rational agents we have implemented and tested. Agents are boundedly rational because they do not have all the information needed to make a rational decision. Instead, they use heuristic rules or strategies to arrive at a production decision. The different types of strategies have different informational and computational requirements. The goal of our experimentation is to investigate which strategies are most successful in a repeated Cournot duopoly game and what outcomes they will yield. The strategies we have tested differ in the amount of information they need to be executed and the computational capacity they require. Some strategies use a very simple heuristic to arrive at a production decision while other strategies make extensive use of data from past periods. In our simulation, all the agents within one firm use the same type of strategy, so we can identify agents with strategy types.

Some of the agents use simple strategies that have also been analytically studied, see for example Young (1998). We start by describing these simple strategies and gradually move on to more complex agents.

3.4.1 Static agents

The first agent type that we have tested is the *static expectations agent*. Agents that use static expectations, have a fixed assumption (during the repeated game) about the quantity their competitor will produce. The chromosome of such an agent contains its static expectation about the production level of its competitor. This expectation is the only evolving value in the case of static agents. In the game, the agents determine their own output by calculating the best reply to this expected quantity (using the Cournot model from Figure 3.1). Because expectations are static, they are not even updated when the agents play an iterated Cournot game. Therefore, the (average) profits of agents playing the Cournot game are equal irrespective of the number of iterations played.

3.4.2 Adaptive and naive agents

As opposed to static agents, adaptive agents adjust their expectations about the behavior of their opponents on the basis of the results from previous iterations. "Adaptive expectations" means that the expected quantity at time t ($Q^{exp}(t)$) is a weighted average of yesterday's quantity ($Q(t-1)$) and yesterday's expected quantity ($Q^{exp}(t-1)$), that is:

$$Q^{exp}(t) = wQ(t-1) + (1-w)Q^{exp}(t-1), \text{where } 0 \leq w \leq 1. \tag{3.6}$$

The parameter w of the forecasting function is learned and updated by the genetic algorithm, as is the expected quantity of the competitor in the first iteration ($Q^{exp}(0)$). The expected quantity thus always lies between yesterday's quantity and yesterday's expected quantity, that is w has to satisfy $0 \leq w \leq 1$. A special case of adaptive expectations occurs when $w = 1$, in that case the agent assumes the competitor will produce the exact same amount he produced in the previous period. These are called naive expectations.

3.4.3 Imitate agents

The imitate agent is the first agent that does not have any knowledge about its reaction curve. The imitate agent plays a *tit-for-tat*-like (Axelrod 1984) strategy that mimics the competitor's strategy. Each iteration the imitate agents produce the exact same quantity that their competitor produced the previous iteration. The agents thus do not need to know their reaction curves. Since this agent reacts to its opponent's move it is theoretically capable of eliciting collusion. The learned parameter in case of imitative agents is the initial belief about the quantity the competitor produced.

3.4.4 Incomplete information agents

The incomplete information agent uses adaptive expectations in the same way as the adaptive agent. But in addition this agent does not know his reaction curve beforehand. The parameters of the reaction curve are encoded on the chromosome, and the agents learns its best replies through evolution. Furthermore, this agent has the possibility to react to its opponent's behavior, which makes collusion possible. As with the adaptive agents, the chromosome of the incomplete information agent describes its initial belief about the quantity the opponent will produce and a weight that is used to update expectations. Furthermore, three parameters that define a linear reaction curve are encoded on the chromosome. The reaction curve is defined by the three learned coefficients a, b and c as described below:

$$(a+1) - \left(\frac{b+1}{c+1}\right) \cdot expectation \tag{3.7}$$

In addition to the information contained on their chromosomes, incomplete-information agents also have a memory that contains the quantity produced by their competitor in the previous round of the repeated Cournot duopoly game. The memory is not placed on the chromosome and is updated each iteration. The memory of the agents is initialized randomly at the start of the game.

3.4.5 Conditional agents

Conditional agents make their output decisions conditional on some information. They do not, however, possess detailed information about the actions or strategies of their competitor. The conditional agent makes his production decision based upon the profit he received in the previous round. The conditional agent is the first agent of all agents described that does not need to know the exact quantity his opponent produced in the previous round. The quantity the conditional agent produces depends on the profit he received in the previous period. This agent has three values encoded on his chromosome: *quantity1*, *quantity2*, and *reservation profit*. If the actual profit the agent receives is lower than his reservation profit, the agent will produce *quantity1* in the next period, otherwise he will produce *quantity2*. This mechanism gives the agent the opportunity to influence his opponent's behavior and to guarantee a minimum acceptable profit.

3.4.6 Autoregressive agents

The autoregressive agent uses price-data from previous periods to make a production decision. The strategy for an autoregressive agent with history n is shown in Equation 3.8:

$$Q_{n+1} = a_0 + a_1(P_n) + a_2(P_{n-1}) + ... + a_n(P_1) \tag{3.8}$$

where the a_i's ($a = 0, 1....n$) are the coefficients that determine the weight of the results of a certain period and are encoded on the chromosome.

3.5 Results for stable market conditions

Table 3.2 shows some characteristics of the different agent types. The first column states whether the agents know their own best-reply curves. The second column denotes what the agents know about the history of the game. A memory of n indicates that the agent has information about the quantities that were produced in the past n iterations. In addition to the information contained on their chromosomes the agents also have a memory that contains the quantity produced by their competitor in the previous round of the repeated Cournot duopoly game (except for the static expectations agent).

Type of Agent	Reaction Curve	Memory (periods)	Learned Parameter (by EA)
Static Expectations	known	none	$Q^{exp}(0)$
Naive Expectations	known	1	$Q^{exp}(0)$
Adaptive Expectations	known	1	$Q^{exp}(0), w$
Imitate	not known	1	$Q^{exp}(0)$
Incomplete Information	not known	1	$Q^{exp}(0)$, Reaction Curve
Conditional	not known	1	conditional outputs
Autoregressive	not known	n	weights

Table 3.2: Some characteristics of the different agent types. Column 2 states for each agent type whether agents know their own reaction curves. Column 3 states the knowledge the agents have about the history of the game.

The profits obtained by the agents in a round robin tournament are shown below in Table 3.3 (averaged over 20 runs). Theoretically, agents producing their Cournot-Nash outputs will obtain a profit of 576, while colluding agents will obtain the average cartel profit of 648. An example of a single run is shown in Figure 3.3 for two static expectations agents. We see that the profits for the static agents are close to the Cournot-Nash outcomes under stable market conditions. Agent performance was tested and found robust under a wide variety of parameter settings. Table 3.3 shows us that the simple imitate agents perform very well and are able of colluding with other agents to obtain outcomes close to the cartel profits. Furthermore, we see that the performance of an agent against its own type is a good indicator of its performance against other agent types. Also, in order to obtain high profits it is important to have a "well-performing" competitor, that is, some amount of cooperation and coordination is necessary to arrive at Cournot-Nash or cartel outcomes.

Average Profits	*Stat.*	*Naiv.*	*Adap.*	*Imit.*	*In.Inf.*	*Cond.*	*AR.*
Static	503	516	467	598	478	432	443
Naive	454	465	449	607	432	429	433
Adaptive	499	328	144	560	330	351	156
Imitate	601	598	572	605	458	442	151
Incomplete Info	362	416	334	379	425	410	388
Conditional	401	413	344	374	378	398	213
Autoregressive	400	248	233	164	148	150	153

Table 3.3: Average Profits over 20 runs, 1000 generations in a round robin tournament under stable market conditions. Entries denote profits obtained by the row player.

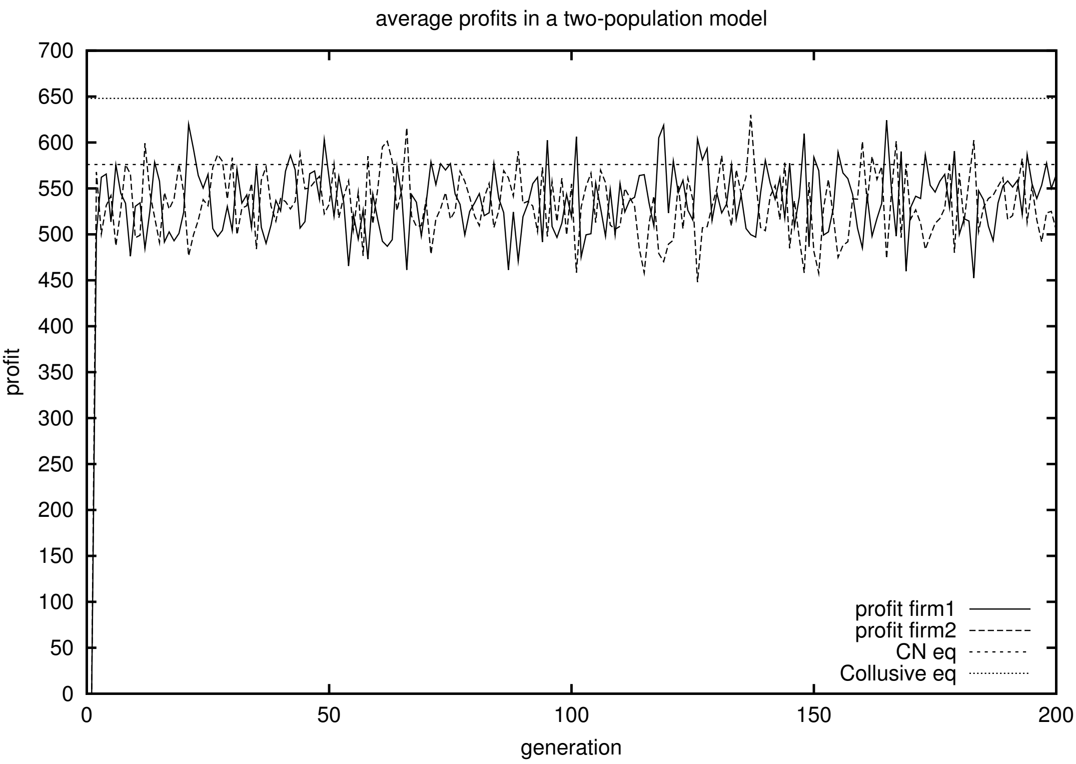

Figure 3.3: Profits of static expectations agents under stable market conditions, a typical run.

3.6 Performance under changing market conditions

So far we have only considered a stable market situation. In such a static environment, the duopoly game is an optimization game. We have seen that a genetic algorithm is capable of solving this optimization problem, especially when chromosomes are short. The results from the previous two sections suggest that the simple agents outperform the more sophisticated agents. One obvious reason for this is the longer chromosome length of the sophisticated agents. With longer chromosomes, the search space is much bigger and the evolutionary algorithm may take longer to converge to and find good strategies. This process becomes more difficult as there are more dependencies between the different parts of the chromosome. The strength of the sophisticated agents is that they have adaptive and learning capabilities as part of their strategies. To see whether this learning capabilities actually enable the agents to make higher profits we have to test the agents in a dynamic environment. In this section we perform experiments where the agents not only have to deal with changes in their environment due to endogenous factors, i.e., the agents they compete against change from generation to generation, but the market is also subject to exogenous changes i.e., in our case; shifts in demand. Thus not only do the agents operate in a changing environment, but they do so under a changing fitness function: The price demand curve slightly changes *every generation* in a continuous fashion. This allows us to test which types of agents adapt faster to the new market situation.

We have chosen a model where demand increases or decreases depending on the time step (iterations or generations). This is done by adjusting the change parameter δ in the inverse demand curve $P = 128 - \delta(q_1 + q_2)$. At the start of the simulation the change parameter is set to one, that is the simulation starts from the static market situation discussed in the previous section. The parameter δ changes as follows: $\delta = 1 \pm gen \div (x + 1)$ where x is a number between 0 and 100. The simulation continues for 100 generations. Performance will be measured by comparing average profits with Cournot Nash and Collusive profits in each situation. Since circumstances change in these models, we do not test agent types with static elements in their strategies such as the static, adaptive and naive strategies discussed in Section 3.4. Furthermore, in the dynamic experiments each strategy is tested only once each time step. This way the simulation can be seen as a time series. Table 3.4 summarizes the outcomes for these experiments.

Agent type	*Profit*	*Std*	*Cournot Nash*	*Collusive*
Conditional	1505	49	1760	1980
Incomplete Information	1466	47	1760	1980
Imitate	1441	68	1760	1980

Table 3.4: Averages over 100 generations, 20 runs in a dynamic market. Entries denote average profits obtained in a round robin tournament.

Figures 3.4 and 3.5 show two individual runs of the experiments. Figure 3.4 describes the profits of conditional agents in a dynamic environment where market circumstances change every generation under a fluctuating scenario. The conditional agent is able to adapt to the changing market condition. Figure 3.5 describes the profits of incomplete information agents in a dynamic environment where market circumstances change every generation.

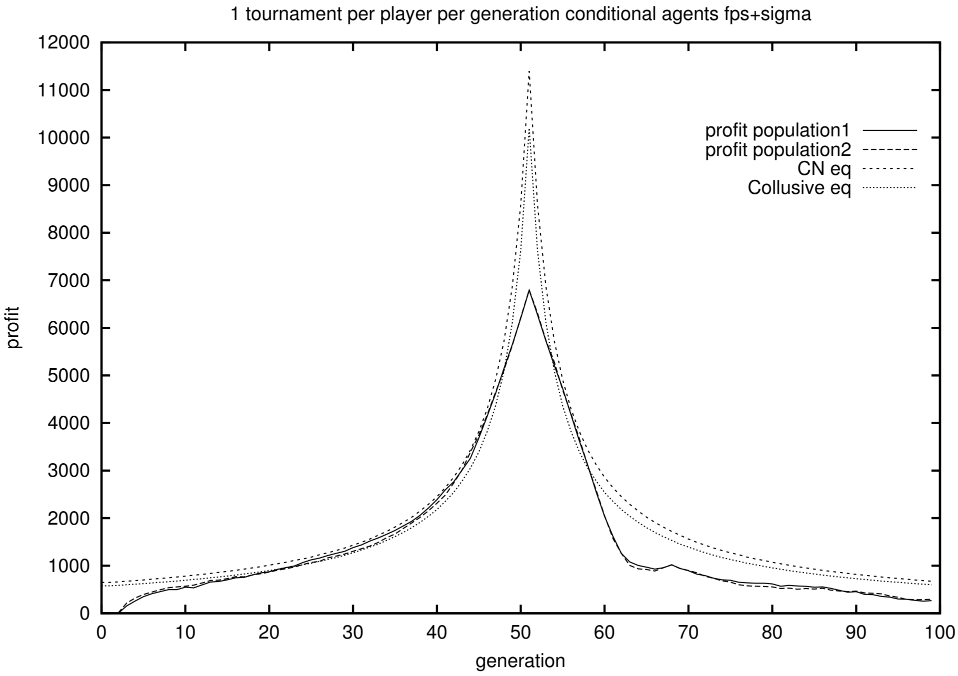

Figure 3.4: Profits of conditional agents in a dynamic environment where market circumstances change every generation: fluctuating scenario. Averages over 20 runs.

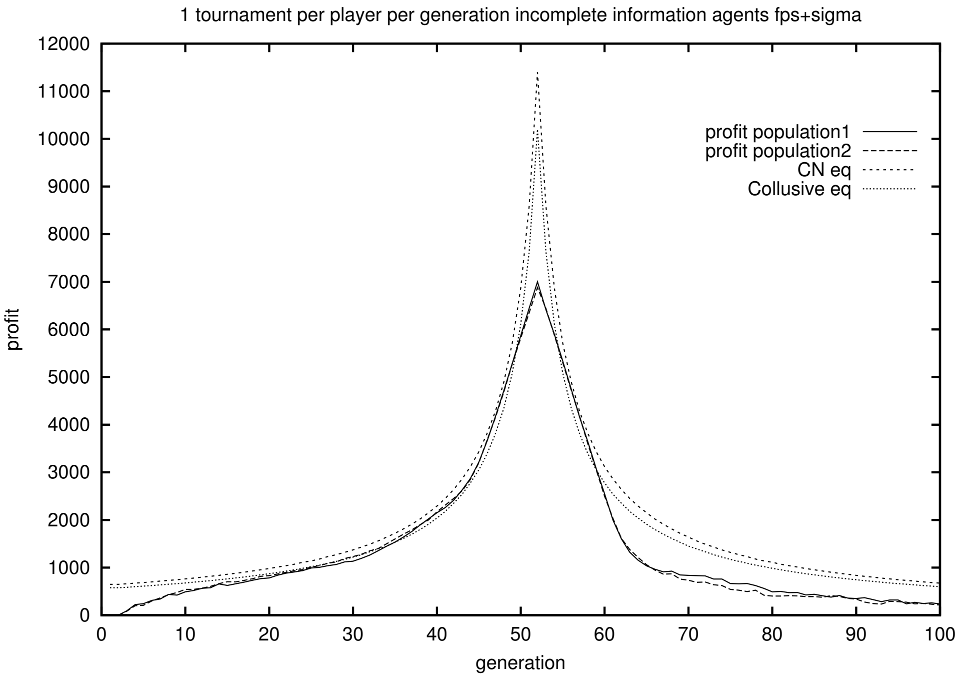

Figure 3.5: Profits of incomplete information agents in a dynamic environment where market circumstances change every generation: fluctuating scenario. Averages over 20 runs.

3.7 Conclusions

We investigate a Cournot duopoly market modeled as a co-evolving system of autonomous interacting agents. We present results for different types of boundedly rational agents. Agent types differ both in the complexity of their strategies and the information they have available to make their production decision. Some types of agents use very simple strategies to make a production decision, while other types use a more sophisticated decision rule. We consider the evolutionary stability of the evolving populations, especially with respect to the different equilibria of the Cournot game. Furthermore, we investigate the performance of the different agent types under changing market conditions.

Most evolutionary simulations study agent behavior and adaptability under endogenous changes in the environment, due to the evolving strategies of the other agents in the simulation. Here, we also consider responses to changes in the exogenous variables, i.e., the model parameters. Under these changing conditions, agents learn production rules that apply to many situations instead of just one output value. We evolve several types of more-sophisticated agents that perform well under changing circumstances. Specifically, agents are able to obtain Cournot-Nash and for some settings even collusive outcomes. In a static situation however, simple agents using traditional learning rules such as naive or adaptive updating, performed better than the sophisticated agents. The simulations were performed using genetic algorithms: we have investigated the performance under a wide variety of parameter settings for the GA.

Chapter 4

Tag-Mediated Interaction

4.1 Introduction

The previous chapter addressed boundedly rational strategies that allowed agents to make decisions when they had only limited information available. This chapter focuses on another mechanism that allows agents to reduce the complexity of their decision environment, namely selective interaction. Using selective interaction, agents are not equally likely to interact with all other agents, as is often assumed in economic theory. Furthermore, agents can make their actions conditional on the identity, appearance or reputation of their opponent. We study the effects of selective interaction by looking at the problem of the emergence of cooperation under difficult circumstances.

An important topic in the fields of biology, economics, and sociology is the "spontaneous" emergence of cooperation in large systems of selfish agents (Axelrod 1984), such as animals in ecosystems or economic agents in markets. Again, we use an agent-based computational economics (ACE) (Vriend 1995, Tesfatsion 2001) approach in this chapter. ACE researchers attempt to model economic systems "bottom up". That is, by modeling the behaviour of individual economic agents and their interactions, the emergence of market behaviour is studied and simulated. ACE thus approaches an economy of agents as a complex adaptive system (Holland 1995). The evolution of such a complex multi-agent system is simulated in this chapter by an evolutionary algorithm (EA) (Mitchell 1996), a recent tool to simulate learning in multi-agent systems. Note that a mathematical analysis of such systems adressing learning behaviour (for instance using Markov chains) is in general hardly feasible and has only been possible for some well-fitted or simple problems (Kandori *et al.* 1993, Young 1993, Axtell *et al.* 2000).

It has been suggested (Holland 1995, Riolo 1997*a*) that under *difficult* circumstances (i.e., when agents are tempted to display uncooperative behavior), "tags" might be useful in promoting the evolution of cooperation. Tags (visible marks) enable agents to distinguish to some extent between other players and bias their behavior accordingly. The research on tags is motivated by the observation that humans often make use of visible marks to quickly classify the other party and in this way to guide the interaction. Axelrod (1984) mentions, for instance, that tags can help to establish stable forms of stereotyping and the formation of groups, even when these tags are not based on objective differences. The reason is that they facilitate selective interaction. In other words, they allow agents to discriminate among agents or objects that would otherwise be indistinguishable.

The key idea is that tags become associated with certain types of behaviour and that agents then learn this association. An agent who discovers such an association between tags and behaviour could profit from that information. As evolution proceeds, more agents discover the profitable rule, and the stereotype becomes soundly established. With the establishment of a stereotype, however, also comes the emergence of mimics (Holland 1993): agents that use the tag associated with a certain behaviour but in fact display different behaviour. If the proportion of mimics in a population becomes very large, however, the other agents will probably react to this and the tag will lose its meaning. An interesting example is the role of fashion as a "tag" to identify people as members of a certain social group.

Previous studies on related topics are the following. Nowak and Sigmund (1998) investigate the role of "images" in the emergence of indirect reciprocity. Similar to tags, images are parts of an agent that are visible to the other agents. However, the images have a fixed meaning and truthfully reveal information, while tags do not have an explicit and predetermined meaning. Axtell *et al.* (2000) investigate a simplified "dividing-the-dollar" game where agents adjust their strategy on the basis of the tag of their opponent. They investigate a Markov-chain model of two fixed-size tag groups where the agents can condition their strategy upon the tag of their opponent. This model does not allow for choice and refusal of partners and the emergence of tags. As in our research, Stanley *et al.* (1994) also study and IPD game with selective interaction. In their model however, partner selection is based on a continuously updated expected payoff function and not on tags (as in our model). The research of Arifovic and Eaton (1995) focuses on the use of tags as type-signals.

Our work is based on the work of Riolo (Riolo 1997*a*, Riolo 1997*b*). Riolo studied a multi-agent system, modeled by a simple evolutionary algorithm (EA) in which agents play a short iterated prisoner's dilemma (IPD) game (of only four rounds) against each other. Under these conditions, it is very difficult for cooperation to emerge. When Riolo added a simple tagging mechanism, however, population dynamics changed dramatically and the agents were able to reach mutual cooperation earlier and over extended periods of time. However, results in (Riolo 1997*a*, Riolo 1997*b*) indicate that the evolving tag-using populations are still relatively unstable.

Riolo's pioneering work is extended in this chapter by investigating the stability of the evolving populations, together with the role of the evolutionary algorithm (EA) that updates the agents' strategies over time. EAs are stochastic search methods based on the principles of natural genetic systems (Mitchell 1996). These algorithms deal with a population of strategies. The strategies are used by the game-playing agents and determine their behaviour in each period of the game. Riolo used a very simple EA in his experiments. In particular, the reproduction of the strategies was modeled as an *asexual* process (i.e., each parent produces one offspring with mutation as the only genetic operator). This model does not allow for the exchange of parts of strategies between agents (recombination). This recombination process may play an important role in the evolutionary search process, however. Earlier experiments by Axelrod (1987) have demonstrated, for instance, that cooperative societies form more frequently if recombination of the agents' strategies occurs.

Recombination of strategies is also called *sexual* reproduction. During this process, well-performing agents (the "parents") exchange parts of their strategies to produce offspring. We will show that the sexual reproduction mechanism has a remarkable stabilizing effect on the evolving populations, as compared to asexual reproduction. This monotonicity has as a side effect that the cooperative populations, on the average, emerge after a longer period of time than in the asexual case. In previous studies on the IPD without tagging (Axelrod 1987, Bragt van *et al.* 2001), it was discussed that cooperative societies fall down because cooperative agents become naive in their behaviour after a while (for a definition, see Section 4.2), focusing on cooperation only, and thus being exploited by defecting agents. In this chapter, we show that when sexual recombination is used with tagging, most cooperative agents do not become naive (in contrast with the asexual case), and that the number of mimics (thus) remains low. This can clarify the emergence of the stability of cooperation, where the recombination prevents the emergence of too many naive agents by enabling the exchange of parts of strategies.

Recombination can be seen as learning parts of other strategies by agents after periods of interaction (IPD) with other agents. Thus, in a social setting, our results could indicate that the possibility of learning about other agents' strategies prevents cooperative agents from becoming naive and thus sustains the existing cooperation.

To obtain a speed-up of the emergence of cooperation, we therefore also consider a natural extension of EAs, incorporating a model for tag-directed parent (mate) selection. Tag-directed mating replaces the standard (random) mating of parents with a more sophisticated matching algorithm, in which parents can select their own co-parent (based upon exterior mating characteristics). Results with this extended model already indicate an increasing number of robust cooperative populations.

This chapter is organized as follows. First, we give a brief outline of our problem in Section 4.2 and our computational experiments in Section 4.3. Our extension of Riolo's work is then presented in Section 4.4. The influence of sexual reproduction is investigated first in Section 4.4.1. Section 4.4.2 then discusses the influence of tag-directed mate selection. In Section 4.4 the two-person IPD is considered. The N-person IPD (NIPD), which is a more suitable model for the so-called "social dilemmas" (Dawes 1980, Hardin 1968), is evaluated in Section 4.5. Both Boyd and Richerson (1991) and Glance and Huberman (1994) have done previous research on the NIPD and on social dilemmas. They note that it is difficult to obtain and sustain cooperation in large groups. Conclusions are drawn in Section 4.6.

4.2 Problem description

We investigate the influence of tagging and recombination on the emergence of cooperation under difficult circumstances. We use the Iterated Prisoner's Dilemma Game (IPD) to model a situation where cooperation is desirable but difficult to achieve. We study both the two player and the N-player game. In an IPD game a player can choose to either cooperate or defect. For the two-player game this means that if both players cooperate they will receive a reward payoff of R=3, if both players defect they will receive the punishment payoff P=1. If one player cooperates and the other defects, the cooperator receives the "sucker's" payoff S=0, and the defector gets the temptation payoff T=5. A payoff matrix for the N-player game is shown in Table 4.3.

We study an evolutionary agent system where agents repeatedly play the IPD game. The agents in our system have both a tag, a mark that is visible for the other agents, and a strategy that determines the next action of the agent based upon some limited memory of the game. The agents decide with which other agents to play the IPD: when two agents meet and their tags are sufficiently similar (dependent on the tag-bias of the agent) they will play an IPD game. The algorithm for tag-based opponent selection is explained in Section 4.3. We use a population based evolutionary algorithm (see Chapter 1) to model the evolution and learning of the agents. All potential solutions are modeled as binary strings; the chromosomes. In our model the chromosomes contain the agents' strategies and their tags, which we further describe in Section 4.3. As usual, the three principal stages of the evolutionary algorithm are selection, recombination (optional), and mutation. In the selection stage, well-performing chromosomes are selected to be the parents of the next generation. This new generation (the offspring) is then created using recombination and finally a mutation is applied to the offspring with a small probability.

Finally, we mention some types of agent behaviour. A cooperative agent is an agent that reciprocates the cooperation of other agents. We subsequently distinguish between naive and discriminating cooperative agents. A naive agent practically always cooperates. A discriminating agent, such as for example, Tit-for-Tat (Axelrod 1984), rewards cooperations while punishing defection.

4.3 Experimental setup

The EA that we implemented consists of a canonical fitness-proportional selection scheme. Three different recombination operators were implemented: single-point crossover, two-point crossover and uniform crossover (Mitchell 1996). Because we are mainly interested in the *relative* performance of the agents, the raw fitness f_i (the average payoff over all played rounds) is normalized by taking $\hat{f}_i = (f_i - \mu)/\sigma + 1$, where μ is the mean population fitness (with standard deviation σ). This implies that a player performing one standard deviation above the mean will (on average) get two offspring. Negative and very low fitness values ($\hat{f}_i < 0.1$) were reset to 0.1 as in (Mitchell 1996) so that individuals with a very low fitness still have some small chance of reproducing.

The representation of the agents' strategies differs from the representation Riolo used in his experiments. Here, we use pure (i.e., deterministic) strategies that are encoded as binary-valued chromosomes, whereas Riolo used mixed (i.e., probabilistic) strategies that were encoded as real-valued chromosomes. We adopt the genetic representation of IPD players as proposed by Axelrod (1987). Repeating Riolo's experiments with this setting led to similar results. Each agent has a memory capacity of three previous moves (one move of his own, and two of the opponent) in our model. There are thus $2^3 = 8$ possible histories and each history uniquely points to an action on the strategy part of the chromosome $[s_0, ..., s_7]$ (see Fig. 4.1), where cooperate is coded by 1 and defect by 0. Thus, a history of 010=2 will lead the agent to perform the action specified by strategy bit s_2 which is defect.

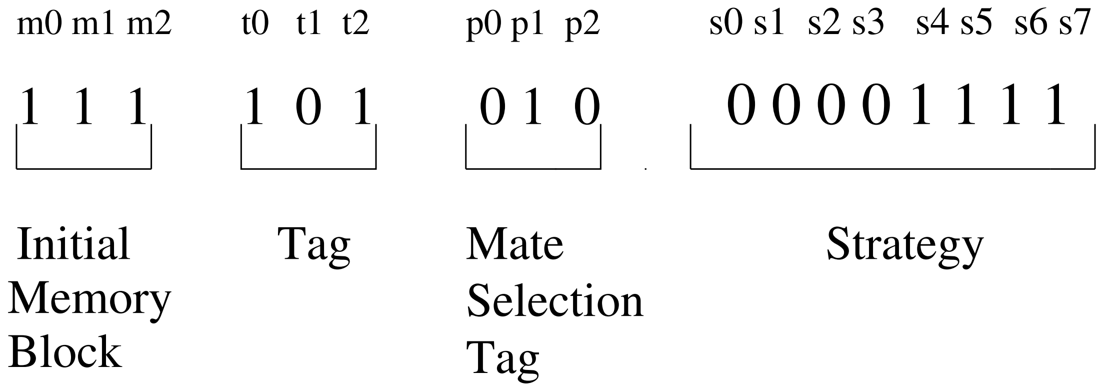

Figure 4.1: The structure of an agent's chromosome when tag-based opponent selection and mate selection are both used.

Three additional bits $[m_0, m_1, m_2]$ are present on this chromosome that form an 'artificial pre-game history' to determine the agents' first moves. In the case depicted in Figure 4.1, this means that the agent will begin by playing cooperate. The initial memory bits are all one (the binary representation of seven) and thus point to the last strategy bit (s7) will be played which depicts a 1 and thus means cooperate. Because the IPD game is very short in our simulations, only four rounds as in (Riolo 1997*a*), the agents only have a memory capacity of three previous moves (instead of a memory of six moves in Axelrod's 151-round simulations).

An agent's tag consists of three tag bits $[t_0, t_1, t_2]$ (see Fig. 4.1). We adopted the same algorithm for tag-based opponent selection as Riolo (1997*a*). In this algorithm (see Fig. 4.2) each player can inspect the tag of a limited number of opponents and tries to find an opponent with a matching tag. This algorithm will be repeated until ten opponents have been selected from the population of 400 individuals. An opponent search cost (of 0.02, see Table 4.1) is associated with each tag trial. When the allowed number of tag search trials is exceeded, the player is matched with a random opponent. After an opponent has been selected, the IPD game is played and the opponent search costs are subtracted from the player's average payoff. In the experiments described here, the tag bias, which specifies the maximum allowed Hamming distance between two tags, is set to 0. In this case, the agent in Fig. 4.1 will only accept opponents with a similar [1,0,1] tag.

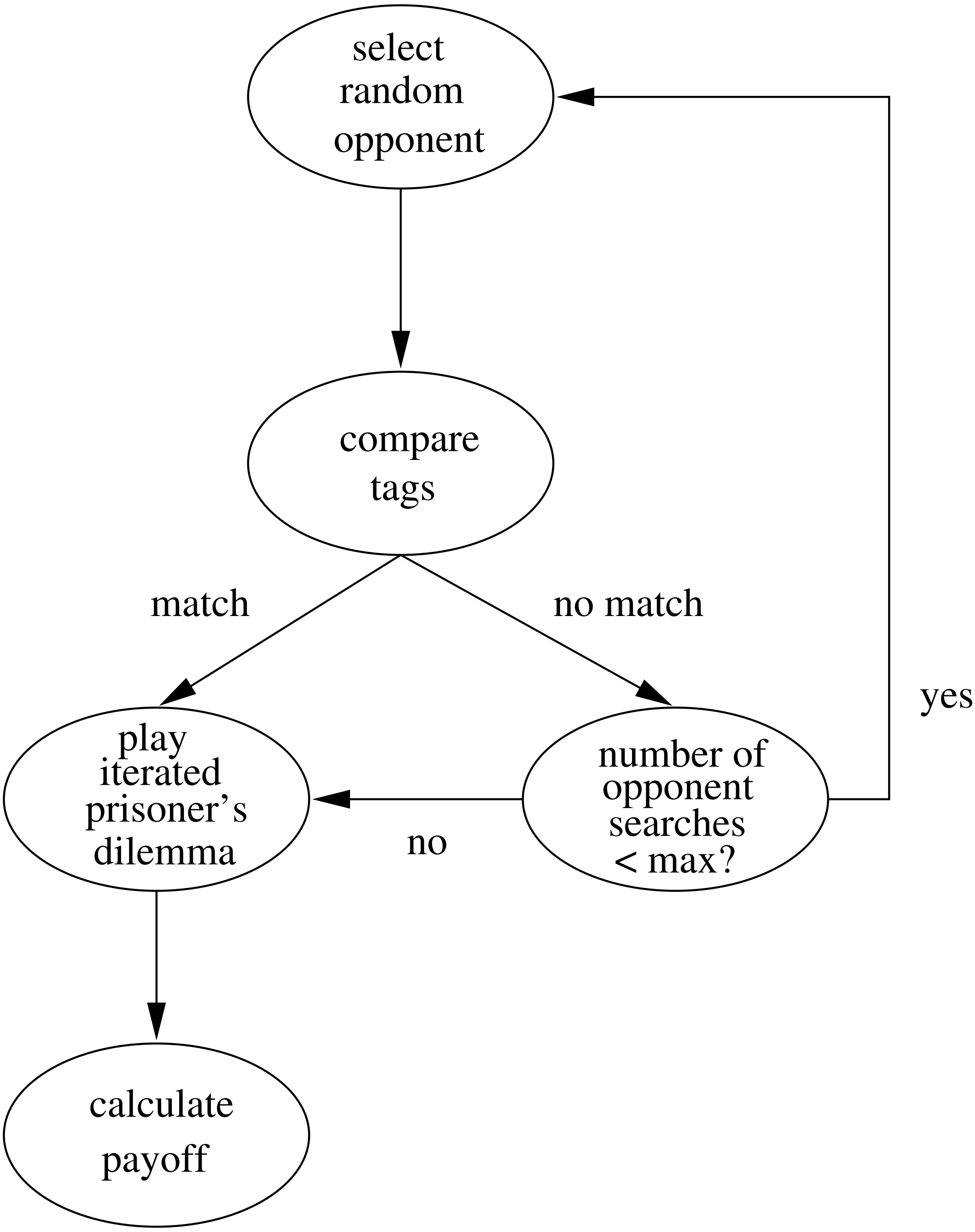

Figure 4.2: The algorithm for tag-based opponent selection.

The agent's chromosome, depicted in Fig. 4.1, also has a mate selection tag. The mate selection mechanism is implemented analogously to the opponent selection algorithm. After calculating the fitness of all agents in the population, a pool of parents is generated (using fitness proportional selection). Each parent then compares his mate selection tag with the mate selection tags of other parents in order to find a matching co-parent. When the maximum number of mate tag trials is exceeded, the parent will mate with a randomly selected co-parent. Note that the mate search costs are set equal to zero (see Table 4.1).

An overview of the model settings is given in Table 4.1. Whenever possible, similar settings as in (Riolo 1997*a*) were chosen. Increasing or decreasing the mutation rate only had a small influence on the results when the mutation range was varied between 0.001 and 0.1, a range of commonly recommended values (Bäck *et al.* 1997*b*).

Parameter	*Value*
Number of Agents	400
Population size (Number of strategies)	400
Number of tag trials	5
Tag size (in bits)	3
Tag bias	0
Opponent search costs	0.02
Mate search costs	0.00
Mutation rate (per bit)	0.025
Crossover probability	0.9/0.0
Number of moves per game	4
Number of games to play	10

Table 4.1: Model settings. We adopted the same terminology as in (Riolo 1997).

The pseudo-code for the evolutionary system is given below. Parameter settings for the computational model are taken from Table 4.1.

```
MAIN
begin
  for each agent do                          (Initialization of the strategies)
      create random bitstring od
end
begin
  for each generation do
      for each agent do
          while number of games <  number of games to play do
                tag-based opponent selection          (see function below)
                play IPD game
                calculate payoffs od
      od
      apply genetic algorithm                          (see function below)
  od
  replace current population with new population
end

TAG-BASED OPPONENT SELECTION
while no opponent has been found do
        candidate opponent = select random opponent
        if number of searches < max
           then
                number of searches ++
                compare tags
                if match then opponent = match fi
        fi
        if number of searches = max
           then
                opponent = random opponent
        fi
od
end
```

GENETIC ALGORITHM
begin
  for *each agent* **do**
    The fitness of an agent is its average payoff over all
    the games played
  od
end
begin
  Scale all fitnesses using sigma-scaling
  repeat
    do *choose two mating parents from the current*
      population by roulette wheel selection **od**
    do *with crossover probability Pcross, cross over*
      the pair at a randomly chosen point **od**
(if no crossover takes place, form two offspring that
are exact copies of their respective parents)
    do *mutate the two offspring at each locus with*
      probability Pmut **od**
    do *place the resulting individuals (chromosomes)*
      in the new population **od**
  until *new population created*
end

4.4 Results and discussion

4.4.1 Asexual vs. sexual reproduction

Figure 4.3 shows a typical run of a population of *asexual* agents playing the IPD using the settings described in Table 4.1. The oscillatory pattern of the mean population fitness indicates that the population alternates between a state of mutual cooperation (when the fitness is close to the reward level $R = 3$) and a society of defectors (when the fitness is close to the punishment level $P = 1$). Societies of cooperators are frequently undermined by "mimics": defecting agents with a tag associated with (a group of) cooperators. These mimics are not recognized as being defectors and can

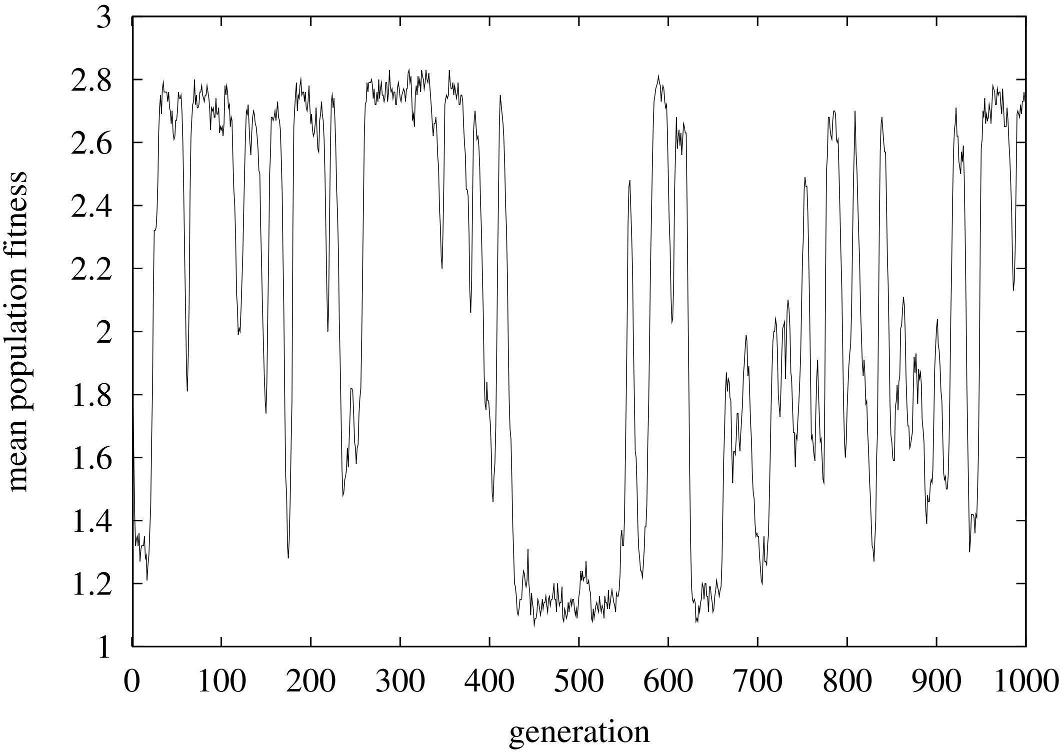

Figure 4.3: Mean fitness of a population of *asexual* agents playing the IPD. The fluctuations in fitness show that stable societies of cooperating agents do not form.

therefore successfully exploit the naive cooperating agents (Riolo 1997*a*), which emerge as in (Axelrod 1987, Bragt van *et al.* 2001).

After introducing sexual instead of asexual reproduction, we obtain a significant change in population dynamics, see Figure 4.4. The oscillatory behavior visible in Figure 4.3 disappears, and the individual runs can now be classified as (1) runs in which a high mean fitness level is achieved and sustained, and (2) runs in which a society of (mainly) defectors forms. Examples of both cases are shown in Figure 4.4. An important aspect is the monotonicity of the observed behavior: once cooperation emerges it persists over long periods of time. For instance, we extended some runs for as long as 10,000 generations without the mean fitness dropping a single time below the 2.3 level after a cooperative period occurred. A society with cooperative periods is defined here as a society in which the mean population fitness remains above 2.3 for 20 successive generations at least once during

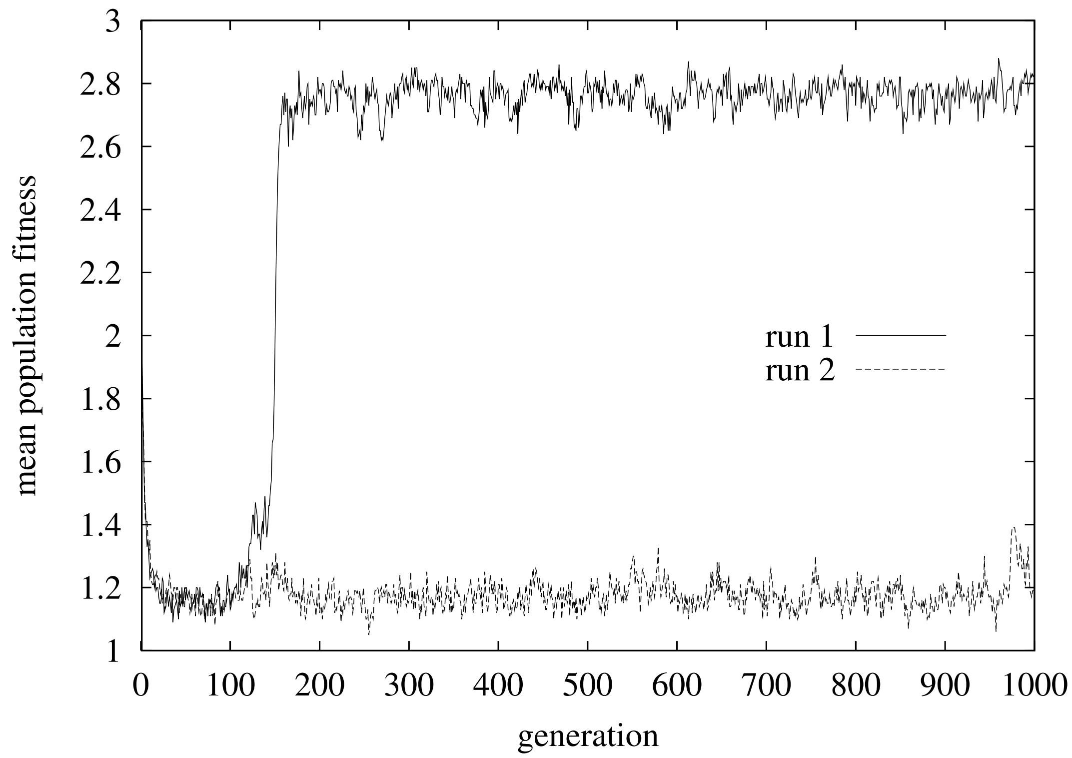

Figure 4.4: Mean fitness of a population of *sexual* agents playing the IPD: two typical runs. Notice that the oscillatory behavior observed in Figure 4.3 disappears.

the entire run. Average results (over 30 runs) are presented in Table 4.2.

	Asexual	*Sexual*	
		(two-point)	*(single-point)*
Runs with cooperative periods (RCP)	0/30	8/30	24/30
Sustained stable cooperative runs (RSSC)	0/30	8/30	24/30
$G_{over}2.3$ $G_{under}1.7$ $G_{to}2.3$	0.59 (0.09) 0.21 (0.06) 22 (21)	1.0 (0.0) 0.0 (0.0) $7.7 \cdot 10^2$ $(7.6 \cdot 10^2)$	1.0 (0.0) 0.0 (0.0) $2.5 \cdot 10^2$ $(2.1 \cdot 10^2)$
Mean historical fitness (MHF)	2.24 (0.1)	1.54 (0.5)	1.95 (0.44)

Table 4.2: Average fitness performance for asexual experiments and sexual experiments. (Numbers are calculated for 30 runs of 1,000 generations; standard deviations in brackets.)

In case of sexual reproduction, the mean fitness over the entire evolution history (MHF) remains rather low. This is due to the fact that it takes longer for cooperative societies to emerge when recombination is used. Notice for instance in Table 4.2 that, when two-point crossover is used, the average number of generations it takes before the fitness first exceeds a value of 2.3, $G_{to}2.3$, increases to $7.7 \cdot 10^2$ in the experiments with sexual reproduction. But once a population has exceeded this fitness level, the population fitness never drops below 1.7 ($G_{under}1.7$=0) and in fact always remains above 2.3 ($G_{over}2.3$=1). $G_{over}2.3$ is defined as the fraction of generations in which the mean fitness is above 2.3, counting only generations after $G_{to}2.3$. Analogously, $G_{under}1.7$ counts the fraction of generations in which the mean fitness has dropped below 1.7. When one-point crossover or uniform crossover was used instead of two-point crossover, the same stabilizing effect was observed. However, cooperation was achieved more often using the less disruptive single-point crossover operator. Populations using single-point crossover evolved to cooperative societies in 24 out of 30 runs compared with only 8 out of 30 runs for two-point crossover. The mean fitness increased to 1.95 (0.44) compared with 1.54 (0.5) in case of two-point crossover (see Table 4.2). In case of uniform crossover, the mean fitness was only 1.34 (0.33) and cooperative societies emerged in 5 out of 30 runs.

The striking stability of the evolving cooperative societies could firstly be explained by the influence of sexual reproduction on the number of mimics. Our experiments show that in cooperative societies distinct tagging groups form, i.e., for each tag one agent type becomes dominant. A mimic is then defined as an agent with the same tag as the dominating agent type, but with at least 5 different strategy bits. Defined this way, the proportion of mimics is 20-30 % in the experiments with asexual reproduction. This large proportion of exploiting agents contributes to the large fluctuations in mean fitness observed in Figure 4.3. The proportion of mimics is much smaller (below 10 %) in the experiments with sexual reproduction. This can be further explained by the absence of naive agents. We found that without recombination, a substantial and increasing part of the cooperating agents used naive strategies: these naive agents are particularly vulnerable to exploiting mimics. The number of naive cooperative agents is reduced by sexual reproduction: in the experiments with sexual reproduction the majority of the cooperative agents have (various) discriminating strategies. Such a majority can be sustained by the recombination operator that allows parts of the strategies to be included in a new (cooperative) agent. Thus, the probability of break-down of one of the cooperative tag-groups is substantially reduced. We remark that, on the other hand, there is always a larger number of naive cooperators present in the asexual populations: this can help to achieve a first cooperative (though unstable) group faster in the asexual case than in the sexual case.

4.4.2 Tag-directed mate selection

Results obtained using the relatively disruptive two-point crossover operator slightly improved when selective mating was introduced. Mate selection yields an increase in mean fitness (from 1.54 (0.5) to 1.68 (0.5)), mainly because, on average, mutual cooperation is discovered earlier ($G_{to}2.3$ decreases from $7.7 \cdot 10^2$ to $7.2 \cdot 10^2$). Again we found that, once cooperation was established, average fitness stayed above the 2.3 level. We also performed experiments with an evolving tag bias for mate selection (located on the chromosome). In this setup, a mate was only accepted if the Hamming distance between the two mating tags was equal to the tag bias. We found, in general, that agents have a strong preference for partners with a similar mating tag (the average mate tag bias converged to a small value), while the average fitness remained the same as in the experiments with a fixed bias for mate selection.

4.5 Tagging in the N-person IPD game

The two-person IPD game can be used to model many social processes where cooperation is desirable but not easily obtained or sustained. There is an important class of cooperation problems, however, which can not be modeled adequately by the two-person IPD game. These problems are the so-called social dilemmas (Dawes 1980). Social dilemmas can, however, be modeled by the N-person IPD (NIPD) game. Therefore, we extend our research on tagging to the NIPD game. Previous computer simulations of the NIPD with evolutionary algorithms (Yao and Darwen 1994) have shown that it becomes substantially more difficult to evolve cooperative societies if the number of players increases (i.e., when N > 2). To investigate whether the tagging mechanism also fosters stable cooperation in the NIPD, we performed a series of experiments for $N > 2$.

The NIPD game is described in detail in (Yao and Darwen 1994). Like in the 2-person IPD, an agent can only choose between cooperation or defection in each round. The payoff (per round) is then determined as follows. If an agent cooperates, his payoff is equal to $2n_c - 2$, where n_c is the total number of cooperating agents. If he defects, he earns a payoff of $2n_c + 1$. In the 4-person IPD this payoff scheme would lead to a payoff of 1 in a society of defectors and a payoff of 6 in a cooperative society. The payoff matrix for an agent in the 4-person IPD is shown in Table 4.3.

Number of cooperators among the other $N-1=3$ players		*0*	*1*	*2*	*3*
agent A	*Cooperates*	0	2	4	6
	Defects	1	3	5	7

Table 4.3: Payoff matrix for an agent in the 4-person IPD (Yao 1994)

In our computational model, the strategy of an agent is now depending on (1) the agent's previous moves and (2) the number of cooperating agents in these rounds. If we consider the 4-person IPD, the length of the strategy block (see Fig. 4.1) is therefore equal to 2^9= 512 bits (3 bits to denote the previous 3 moves of the player, and 3 times 2 bits to denote the number of cooperators in the previous 3 rounds). The initial memory block then also has a size of 9 bits. As in the previous experiments with the 2-person IPD, the length of the tag is equal to the size of the initial memory block (i.e., 9 bits). Without a mating tag, the total chromosome length for agents in the 4-person IPD is therefore equal to 530 bits. An agent in the 4-person IPD repeatedly applies the algorithm for tag-based opponent selection as is shown in Fig. 4.2. The agent is thus allowed 5 tag trails per opponent.

In the evolutionary algorithm, the mutation probability (per bit) is reduced to 0.002 (from 0.025, see Table 4.1) to avoid an excessive increase in the number of mutations due to the much longer chromosome length for

$N = 4$. (With this mutation rate, on average one bit per chromosome is mutated for $N = 4$ agents.) Values for the remaining parameters were kept the same as in the 2-person IPD (see Table 4.1).

Computational results for $N = 4$ are reported in Table 4.4. The degree and stability of the emerging cooperation is measured by monitoring the mean historical fitness (MHF), the number of runs with cooperative periods (RCP), and the number of runs with sustained stable cooperation (RSSC). Figure 4.5 shows a typical run with asexual reproduction. The horizontal lines in Fig. 4.5 indicate the mean population fitness for different values of the number of cooperators n_c (accounting for tag search costs). Notice that it is very difficult to achieve cooperative societies (the MHF remains low). Although the tagging mechanism increases average fitness levels, population-wide cooperation does not emerge. Remember that the average population fitness would still be equal to 2.25 if there is only one cooperator in each round (i.e., $n_c = 1$, corrected for the tag search costs). Note from Table 4.4, however, that the number of runs with cooperative periods (RCP) and the number of runs with sustained stable cooperation (RSSC) significantly increase in case of tag-using agents.

To gain more insight in the nature of the cooperation that occurs, we examined the number of cooperators per tag group and in each round of the game. As in the 2-person IPD, distinct tag groups emerge after approximately 100 generations. Most of these tag groups exhibit defective behaviour. Sometimes, however, a tag group discovers cooperative strategies. In most runs, this "cooperating" group was of a substantial size, periodically increasing average fitness levels to 3 or even higher. The maximum fitness measured during the experiments was approximately equal to 5, which indicates the emergence of a large group of cooperators (also given the fact given that the agents have to pay tag search costs).

The MHF data presented in Table 4.4 suggests that sexual cooperation does not help the emergence of cooperation. Notice for instance that the MHF decreases from ≈2.08 in the asexual experiments to ≈1.14 in the experiments with single-point crossover (when tag use is allowed). A more careful analysis however shows that the higher mean fitness in the asexual experiments is due to the fact that in these experiments incidentally a very high fitness is achieved. This cooperation level cannot be sustained however. The experiments further show that (due to slow convergence), fitness levels in the runs without sexual reproduction decrease very slowly, but once all cooperation is completely lost, it is very difficult for the asexual agents to reestablish it (at least not within the 10,000 generations we have examined). Figure 4.6 shows that this is not the case for experiments with sexual agents: after an initial period of low fitness levels a transient towards an increased level of cooperation occurs (after approximately 600 generations). This increased level of cooperation is then sustained in the remainder of the experiment.

Sexual reproduction	*Tags*	*MHF*	*RCP*	*RSSC*
No	No	1.08 (0.02)	0/30	0/30
(single-point)	No	1.11 (0.03)	0/30	0/30
(two-point)	No	1.11 (0.02)	0/30	0/30
No	Yes	2.08 (0.47)	25/30	3/30
(single-point)	Yes	1.14 (0.13)	8/30	4/30
(two-point)	Yes	1.62 (0.50)	20/30	14/30

Table 4.4: Influence of tagging and sexual reproduction in the 4-person IPD. Note that both the mean historical fitness (MHF), the number of runs with cooperative periods (RCP) and the number of runs with sustained stable cooperation (RSSC) increase if the agents can use tags. (Statistics are calculated for 30 runs of 10,000 generations; standard deviations in brackets.)

Sexual reproduction	*Tags*	*MHF*	*RCP*	*RSSC*
No	No	6.8 (0.2)	5/10	2/10
(single-point)	No	5.7 (0.3)	4/10	2/10
(two-point)	No	6.2 (0.2)	5/10	3/10
No	Yes	9.5 (2.4)	8/10	4/10
(single-point)	Yes	12.9 (1.3)	7/10	7/10
(two-point)	Yes	10.6 (1.5)	8/10	8/10

Table 4.5: Experimental results for the 8-person IPD when the number of iterations per game is increased to 10. Note that the mean historical fitness (MHF) increases when the use of tags is allowed. (Statistics are calculated for 10 runs of 10,000 generations; standard deviations in brackets.)

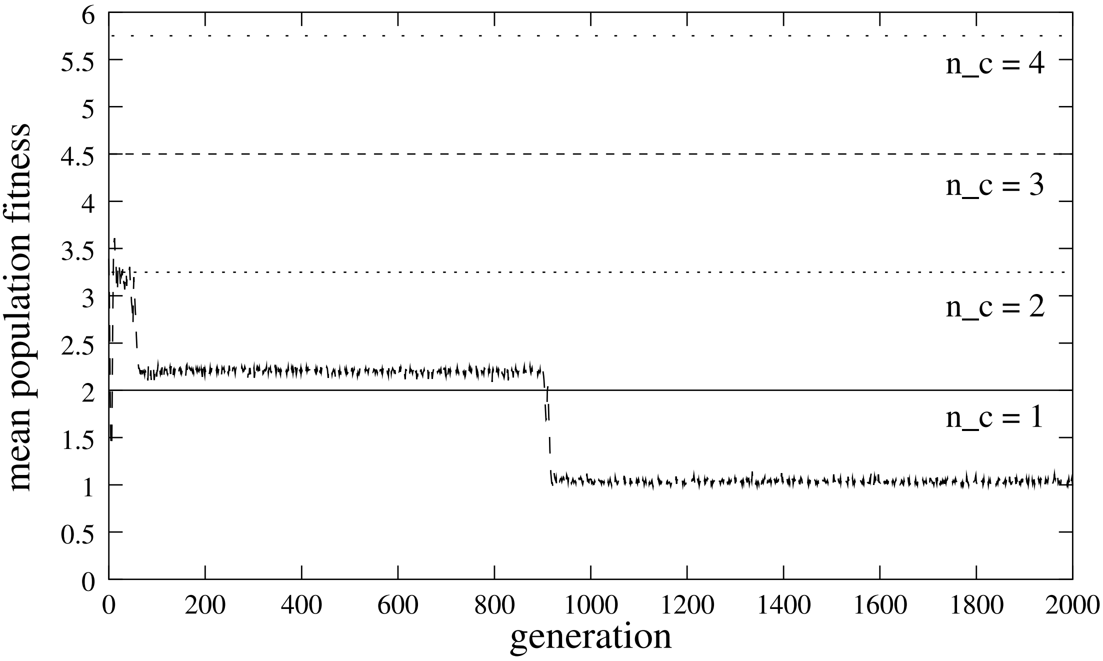

Figure 4.5: Influence of tagging and asexual reproduction in the 4-person IPD, one typical run with cooperation. Note the step-wise decrease of fitness as evolution proceeds. The horizontal lines indicate the fitness of populations with on (average) 1, 2, 3, or 4 cooperators.

When the number of players was increased to 8 or 16 players, only defective societies where observed, with fitness levels always lower than 1.5. (In the 8-person IPD, the mutation probability is set equal to 0.0002; in the 16-person IPD this probability is set equal to 0.00003) We found in additional experiments that this difficulty in achieving cooperation was caused mainly by the small number of rounds (namely 4, see Table 4.1) in the game. When the game length increases, average fitness levels rise, and cooperation is achieved more often. As an example, Table 4.5 shows the results for the 8-person IPD when the number of iterations is increased to 10. Remember that fitness values in the 8-person IPD lie between 1 (for $n_c = 0$) and 14 (for $n_c = 8$). Again we see that tags help to establish cooperation in societies of agents playing the NIPD. If we look at the population after 10,000 generations we also observe a strong convergence per tag group as was the case in the 2-person IPD.

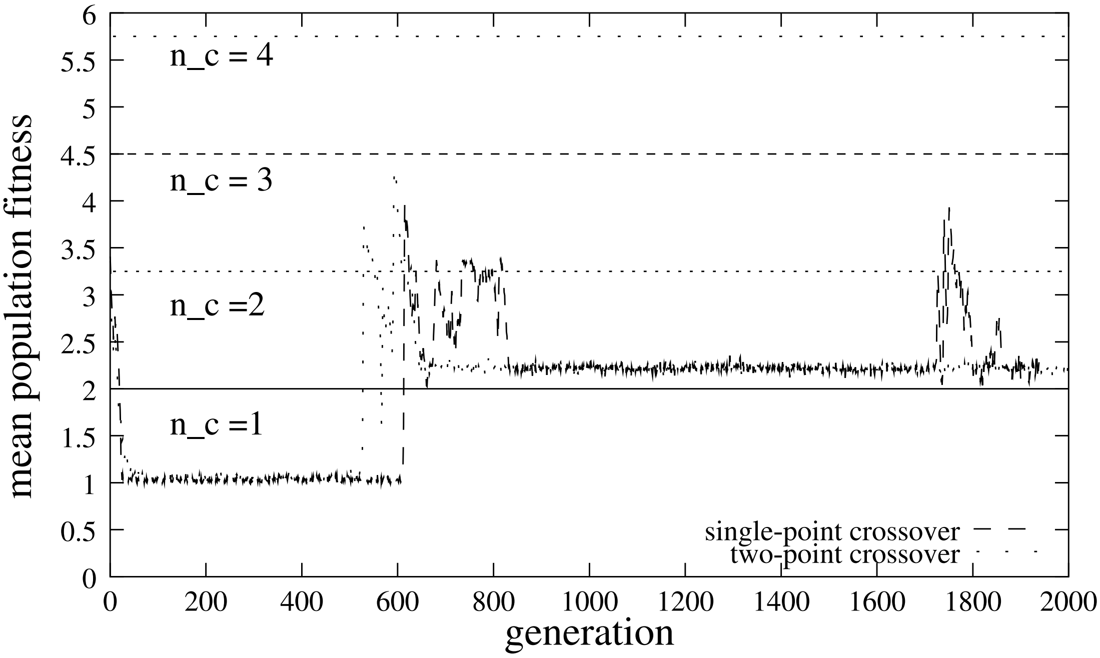

Figure 4.6: Influence of tagging and sexual reproduction in the 4-person IPD. Two typical runs with cooperation. Notice that, once some cooperation is achieved, the population stays out of the defective zone throughout the entire run.

In the 16-person IPD, increasing the number of iterations to 10 causes a small increase of the level of cooperation. In 10 runs of 10,000 generations each, cooperation (i.e., *fitness* > 1) emerged only once with tag-using agents. These results are roughly compatible with experiments from Yao and Darwen (Yao and Darwen 1994), where no cooperation was found in a 16-person IPD without tags (with 100 iterations and agents with a memory of size 2). The failure to reach cooperation can be caused by the large search space (the chromosome length without tags is 32,783 in the 16-person IPD without tags).

4.6 Conclusions

We have studied evolutionary processes in multi-agent systems. In this chapter, we have investigated the "evolution of cooperation" in a population of agents playing the tag-mediated iterated prisoner's dilemma (IPD). Computational experiments have been performed using evolutionary algorithms (EAs). We have shown that the tagging mechanism and the reproduction process of the agents play a major role in the formation of stable cooperative societies.

In the 2-person IPD, the population alternates between a state of mutual cooperation and a society of defectors in a model with asexual reproduction (i.e., when children are simple copies of their parents and mutation is the only genetic operator). A distinct behavior emerges if reproduction of the agents is sexual (i.e., when the parental strategies are recombined during the reproduction process). We observed, for instance, the formation of very stable societies of cooperative agents, a phenomenon not observed in the experiments with asexual reproduction. Furthermore, we found that cooperative societies emerge more frequently when the recombination operator is not too disruptive (e.g., a single-point crossover scheme). Finally, we proposed a tagging mechanism to enable biased partner selection. Results for this extended model were presented.

Results for the N-person IPD showed that (1) it becomes more difficult to evolve cooperative societies if the number of players is increased (i.e., $N > 2$), and that (2) tagging does help to achieve cooperation in the N-person IPD game. Furthermore, stable long-term cooperation emerges more frequently when sexual recombination of the agents' strategies occurs (as in the two-person game studied in this chapter). This is due to the fact that recombination prohibits the emergence of large numbers of naive cooperative strategies and thus reduces the impact and success of mimics. Strategic information exchange thus helps to build robust strategies, while tagging creates favorable circumstances for cooperation.

Chapter 5

Agent behavior and interaction in a trade network

5.1 Introduction

Electronic commerce and trading of information goods significantly impact the role of intermediaries. Because the services of intermediaries are costly to consumers, the question arises why intermediaries are necessary. This question is particularly relevant in electronic commerce where consumers can decide to buy directly from the producer. Electronic markets facilitate direct contact between consumers and producers, reducing the influence of intermediaries. On the other hand, intermediaries may be able to reduce the information overload consumers face.

The central question of this chapter is whether intermediaries can make a profit in an information economy. One reason that traditional intermediaries can make a profit is because they trade a larger volume of goods than the average consumer and can buy at lower prices. When information goods are traded over the Internet, however, this advantage disappears. Information goods differ from traditional goods because they are costly to produce but cheap to reproduce and there are no natural capacity limits for additional copies. Therefore pricing structures for information goods are different than for traditional goods.

One possible role for the intermediary in an electronic market is the role of search expert—intermediaries are in the market for more periods of time and make more transactions than individual buyers and sellers.

Such an intermediary thus gains expertise on where the best deals are to be found. In the age of information overload, intermediaries that reduce consumer search costs may be able to make a profit in electronic markets. We investigate the conditions under which such intermediaries can attract a customer base. We use evolutionary computer simulations, a methodology from the field of agent-based computational economics (ACE) as in (Tesfatsion 2001, Alkemade and Poutré 2002), to study an electronic trade network where consumers, producers and intermediaries trade an information good. Agent-based computational economics studies economic phenomena as they emerge from interactions between individual, boundedly rational agents. In our case, we initialize a trade network with a fixed, user-specified number of consumers, producers and intermediaries. Over time, the agents in the networks learn which links to form and we study the trade network that arises from those repeated local decisions.

We use a network economics approach (Shapiro and Varian 1999) to study electronic trade networks. Network economics holds the view that individual actions, and in turn aggregate outcomes, are in a large part determined by the interaction structure (as in ACE). This stands in contrast with the market view of the economy where buyers and sellers are anonymous and the structure of the interaction is typically considered less important. The network economics view of the economy states that there must be a connection (an information link) between buyers and sellers in order for any trade to occur. A connection between two agents means that there is a flow of relevant information between the two agents (i.e., a buyer that requests price quotes from a seller or subscribes to a mailing list). If a buyer does not know about a certain seller that offers the best price, this price will not influence his purchase decision. Agents trade over the network and buyers have to decide which connections to form to the sellers. Over time, some connections may yield a higher utility than others and consumers can decide to update their link pattern. Links are a model for consumer search behavior and are costly, i.e., the buyer has to invest some resources (time, money) to find and maintain contact with potential sellers. Buyers make this strategic linking decisions based on the information they have available. To model this boundedly rational search and learning behavior of the consumers we use an evolutionary algorithm.

Section 5.2 describes the trading agents and the economic model. In Section 5.3 the use of an evolutionary algorithm as a model for agent learning behavior is explained. A layout of the experimental design is given in Section 5.4. Section 5.5 provides results for the agent-based simulations we have performed. In Section 5.6 we study how an intermdeiary can become an expert. Conclusions will be given in Section 5.7.

5.2 The economic trade network model

We consider a trade network game of cost-minimizing boundedly rational consumers, and profit-maximizing producers and intermediaries. The goal of this research is to investigate the influence of the network structure and information level of the agents on the level of intermediated trade in the model. Each period of the game consumers buy a single unit of an information good, which they can buy directly from the producers or through an intermediary. Production of information goods involves high fixed but low marginal costs. In this chapter we assume that initial production costs are sunk and reproduction is very cheap and easy, therefore we impose no capacity constraints on the number of goods an individual producer can supply. Trade can only occur if there is a connection, a link, between a consumer and a seller (producer or intermediary). Buyers (that is, consumers and intermediaries) strategically decide which links to form by choosing a linking strategy from their associated strategy base, this strategy base is periodically updated by an evolutionary algorithm. Producers strategically decide what prices to charge during a trade period. The trade network thus consist of consumers, producers, intermediaries and the links connecting them. The model is initialized with a fixed, user-specified number of consumers, producers, and intermediaries. Figure 5.1 depicts the economic model and a possible trade network.

We investigate the influence of the initial expertise level of the intermediary, that is, the influence of the number of links the intermediary initially sustains with producers, on the resulting trade network. Well-connected intermediaries have a better chance of finding the best price than inter-

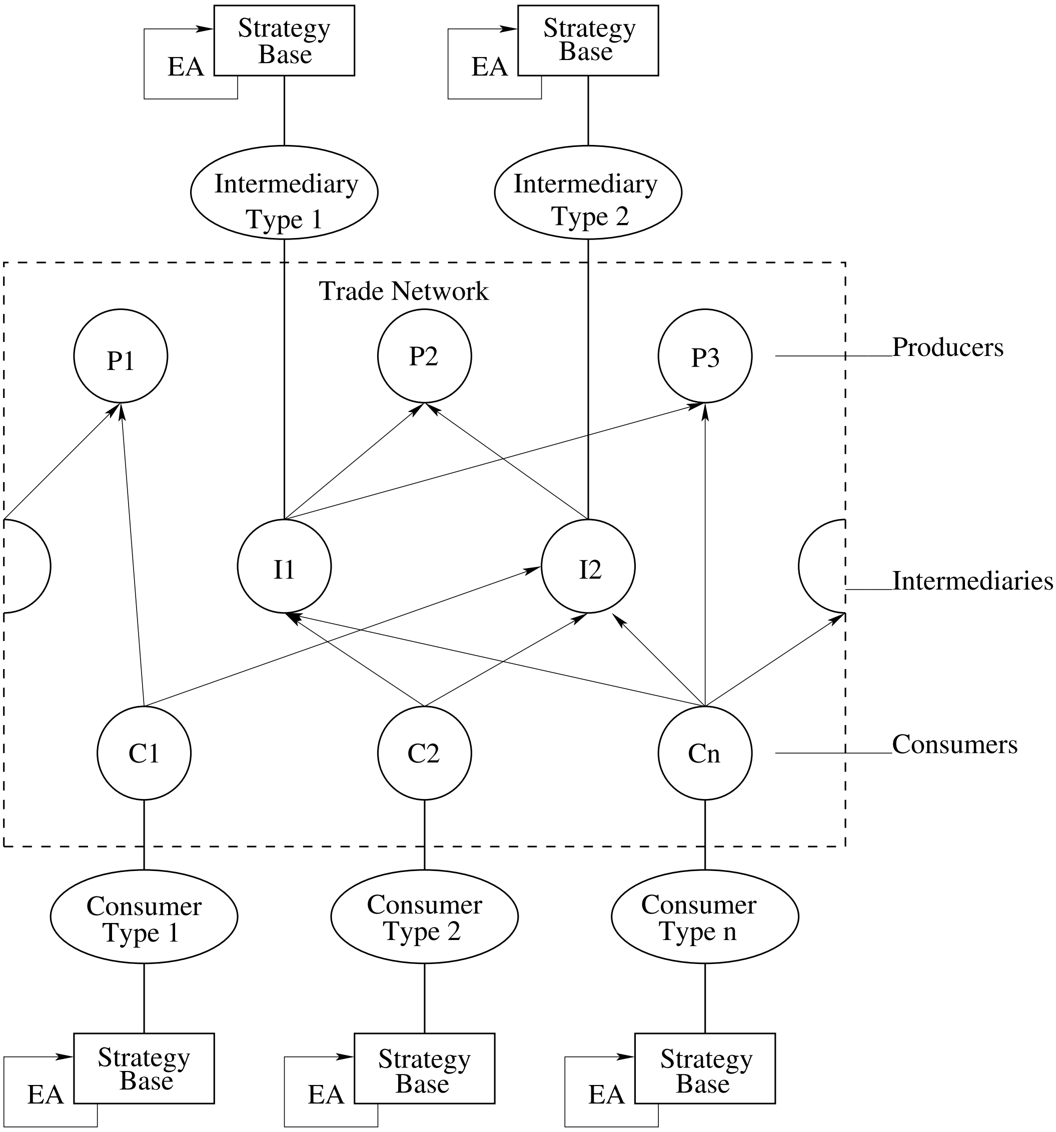

Figure 5.1: The model and example of a possible trade network between consumers, producers and intermediaries. The trade network is depicted within the dashed square.

mediaries without links. We study the level of intermediation for different producer price schedules that lead to different market dynamics. Below, a more detailed description of the different types of economic agents present in the model will be given. An overview of the evolutionary algorithm that is used to learn better linking strategies is described in Section 5.3. Each trade period the flow of goods and information follows the steps depicted in Figure 5.2. Again this will be described in more detail below.

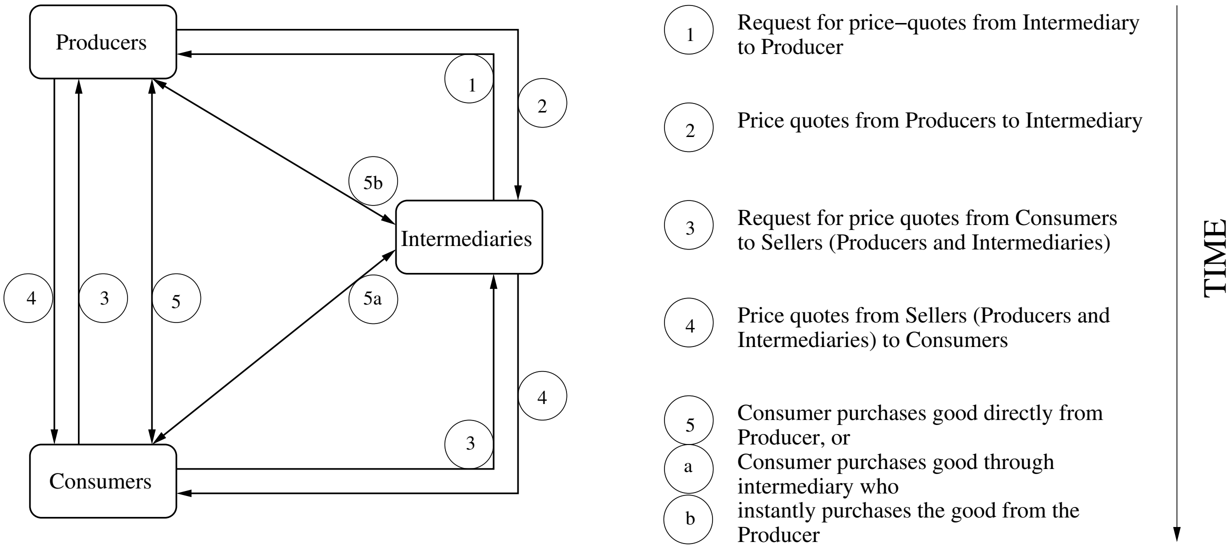

Figure 5.2: Flow of information between Consumers, Producers and Intermediaries in the trade network model.

5.2.1 Heterogeneous, boundedly rational consumers

Consumers in the trade network are cost minimizing consumers. Each trade period, a consumer buys a single unit of a homogeneous information good. Consumers can only buy from a seller they have formed a link to and these links are costly. Each link has a per unit cost associated with it (in the current setup these cost are taken constant for all links). The total search cost (C_s) a consumer incurs during a single trade period is equal to the number of links (l) he maintains times the search cost per link (c_l) (for parameter values see Table 5.1). The utility of a consumer (U_c) is then defined as the negative of this total search cost plus the purchase price (P):

$$U_c = -(l \cdot c_l + P)$$

Consumers in our model are boundedly rational (Simon 1984, Newell and Simon 1972), that is, the consumers do not have the information and the computational capabilities needed to make an optimal decision. To make a perfectly rational decision, an agent needs to know the exact price each producer will charge and then form only one link to the producer offering the lowest price. This is not a realistic scenario since the price depends on the decisions of other consumers as well. Individual consumers have no direct information about the behavior of other consumers. This is particularly true in electronic commerce where shops have no physical location and consumers do not have access to knowledge about other consumers. The consumer thus faces a trade-off between the number of costly links he forms and his utility. More links mean a higher chance of finding the best price while fewer links mean lower search cost. Furthermore, since we study a model where producers change their prices over time, the optimal linking strategy for the consumer can also change. Therefore the consumer has to learn by trial and error (experience) which link patterns yield higher utility. A consumer's strategy determines his search behavior (which links he forms and maintains). This strategy is learned and adapted by an evolutionary algorithm. The fitness of a search strategy is equal to the utility that particular strategy yielded for the consumer.

The search strategy thus determines how many and which links a consumer forms to producers and intermediaries. The boundedly rational consumer then buys the product from his cheapest linked seller. However, with a small error probability ϵ the consumer purchases the information good from a randomly chosen seller. This reflects the fact that the consumer may make a mistake in selecting the cheapest seller or that the physical (Internet) connection with his preferred seller may fail. Search thus occurs at two levels; search over a link to find the best price quote and search at a more abstract level to find the best search strategy (the EA). The algorithm for the consumers is as follows:

Each trade period consumers take the following steps:
Step 1. Decide which links to form to sellers
Step 2. Choose preferred seller:
With probability $1 - \epsilon$ (where ϵ is small)
Preferred seller is the linked seller offering the lowest price
Or, with probability ϵ
Preferred seller is a randomly selected linked seller
Step 3. Buy the good from the preferred seller
Step 4. Calculate utility: $-$ (price + total search cost)
Step 5. Update link strategy

5.2.2 Heterogeneous adaptive producers

At the beginning of each trade period, a producer has to decide which price to charge during that period. The prices set by the producers are the driving force behind the dynamics of the trade network. There are no capacity constraints for the producers since we are dealing with an information good. We investigate two basic adaptive price schedules, where producers adjust their prices on the basis of economic results from the previous period.

First, we consider the case where producers use a **downward sloping price demand curve** (Mas-Colell *et al.* 1995) where price decreases as demand increases. Naive expectations are used to update beliefs about expected demand: $Demand^{exp}(t) = Demand(t-1)$. Expected demand for the first trade period is chosen randomly from a uniform distribution [0..*number of consumers*]. Hence, producers have heterogeneous expectations about demand. In this scenario the sellers use a basic price setting mechanism and do not recognize any market power they may have. Since producers charge the lowest price when demand is high, this price schedule presents a coordination problem to the buyers. Buyers are best off if they all form one and only one link to the same producer. We could think of a good that exhibits network effects (software).

Production of information goods involves high fixed, but low marginal costs. Some economists (Choi *et al.* 1997) argue that such a cost structure requires a pricing schedule where information goods are priced according to consumer value. We therefore investigate a second pricing scenario where producers continually try to increase their profits. Producers use a so-called **derivative follower** (DF) algorithm—see (Greenwald and Kephart 1999, DiMicco *et al.* 2001)—to determine their prices. The DF algorithm is a (local) search algorithm that dynamically adjusts the price for the offered good based on observed profits. This algorithm starts at some user-specified price level (in our case randomly drawn) and then, step-by-step, changes the price in the same direction until the current profit drops below the profit obtained in the previous trading period. In that case, the search direction is reversed and steps in the other direction are made. Every time the profit decreases the search direction is reversed again. The direction of change is randomly set to -1 or 1 at the beginning of the simulation. Similarly, the initial price is randomly chosen from a uniform distribution [*minimum price*, *maximum price*] (heterogeneous producers). This algorithm is able to react very quickly to changes in the profit landscape. Two additional advantages of the DF algorithm are that the underlying idea of the DF is very intuitive and that the DF requires very little problem specific knowledge. The derivative follower algorithm leads to more complex price dynamics than the downward sloping price demand curve described above, so the search problem for consumers is more complex. The algorithm for the producers is as follows (for an overview of parameter values used in the experiments see Table 5.1):

Each trade period producers take the following steps:

Scenario 1: Producers using a downward sloping demand curve

- **Step 1** Calculate *expected demand*,
  using: $Demand^{exp}(t) = Demand(t-1)$
- **Step 2** Determine *Price P*,
  using $P = a - (b * Demand^{exp}(t))$
- **Step 3** Sell the good to buyers
- **Step 4** Calculate actual demand ($Demand(t)$)

Scenario 2: Producers using the derivative follower algorithm

- **Step 1** **If** the profits in the previous period are smaller than profits in the previous period
  Then reverse direction of price adjustment (dpa)
- **Step 2** Determine *Price P*,
  using: $P = P_{old} * (1 + (dpa * stepsize))$
- **Step 3** Sell the good to buyers
- **Step 4** Calculate actual profit

5.2.3 Profit maximizing intermediaries

We investigate whether intermediaries that are experts at searching can exists and make a profit in an electronic trade network when consumers can also buy directly from the producers. To gain more insight in the role of such intermediaries we vary the level of knowledge or expertise that intermediaries have about the trade network and test whether they are able to attract customers. Intermediaries buy goods from producers and sell them to consumers. Intermediaries charge consumers a percentage (markup) of the acquisition price, so that they can make a profit. Profit equals the total number of goods sold to consumers times the markup percentage minus the total link cost of the intermediary. Intermediaries can buy and sell more than one unit of the homogeneous product during a single trade period. On the consumer-side intermediaries function like the consumers described above. That is, they have a search strategy that is periodically updated by an evolutionary algorithm. The fitness of a search strategy is equal to the profit the intermediary obtained using that strategy. We assume that the intermediary only buys a product if there is a consumer he will sell it to (no stocks). The intermediary thus acts as a broker. Price quotes remain valid for the entire trade period. If an intermediary receives an attractive price quote from a producer he can purchase the information good instantly when a consumer arrives that is willing to buy the product at the price quoted by the intermediary. The algorithm for the intermediaries is as follows:

Each trade period intermediaries take the following steps:
Step 1 Decide which links to form to sellers
Step 2 Determine *Price*,
using: $P = Acquisition\ Price * (1 + markup)$,
Acquisition Price is the price from the
linked seller offering the lowest price
Step 3 Calculate profit (number of units sold ∗ markup)
Step 4 Update link strategy

To model the fact that the intermediary may (initially) have expert knowledge about the trade network and can use this knowledge to make a profit, we vary the initial network density of the intermediary (the number of links to producers that the intermediary sustains initially). When the intermediary has complete knowledge about the trade network, he maintains links to all producers. An initial network density of 0.6 means that at the beginning of the simulation the intermediary maintains links to (and can obtain price quotes from) sixty percent of the producers.

5.2.4 Market dynamics

Market dynamics are driven by the prices set by producers. At each timestep of the model, the agent actions described above are executed resulting in the model sequence described below. Initially, the number of producers, consumers and intermediaries are chosen, as well as the initial network density (expertise levels) of the intermediaries. The outcome of the evolutionary agent-based simulation now depends on those initial conditions and the agent-interactions.

Each trade period the economic agents take
the following steps:

1. Producers choose their prices
2. Intermediaries form links to producers based on their search strategy
3. Intermediaries choose their prices
4. Consumers form links to sellers based on their search strategy
5. Consumers buy 1 unit of the good from their preferred linked seller
6. Consumers calculate their utility
7. Consumer search strategies are updated by the EA
8. Intermediaries calculate their profits
9. Intermediary search strategies are updated by the EA
10. Producers calculate their profits
11. Producers update their prices for the next period

5.3 Agent learning

To model the learning and search behavior of the evolutionary agents we use an multi-population evolutionary algorithm (EA), see Chapter 1. A schematic overview of the model is given in Figure 5.1. There are three groups of economic agents present in the model: consumers, producers and intermediaries. There are different types of consumer agents. In the current setup agent types are heterogeneous with respect to the strategies they use. Different agent types select their strategies from a different strategy base. When a consumer of a certain type chooses a strategy, he picks a strategy out of the associated strategy base. The strategy base of a particular consumer type is periodically updated by an evolutionary algorithm. It can therefore happen that agents of a certain type prefer direct trade while other agent types trade through the intermediary. Similarly, each intermediary type draws strategies from a different strategy base.

5.4 Experimental design

The goal of the experiments is to investigate how the initial expertise level of the intermediary influences the level of intermediation. We vary the initial level of expertise of the intermediary—the number of links the intermediary initially maintains to producers—and study the evolution of the trade network over several trade periods. We thus perform an agent-based computational study of an electronic trade network modeled as an evolving system of autonomous interacting agents. The resulting trade network and network dynamics are a result of the local interactions of autonomous agents over time.

The model described above allows us to investigate the role of the intermediary in electronic trade networks where consumers can choose to buy directly from the producer or through the intermediary. The experiments are conducted under different conditions concerning the pricing mechanism for the information good and the level of expertise of the intermediary. The services of the intermediary are costly and a rational consumer with perfect information would therefore prefer to buy directly from the producer. However, in our model it is costly to find the cheapest producer, i.e., the consumer has to invest some resources (time or money) to find a good deal. An intermediary with a good network may be able to take over the search function from the consumer and make a profit. For a consumer there is no visible difference between buying from the intermediary or buying directly from the producer. In this chapter we do not assume that intermediaries are more trustworthy than producers: the only way intermediaries can make a profit is if they are better at finding the cheapest producer than the consumers. To model the fact that the intermediary may have expert knowledge about the trade network and can use this knowledge to make a profit, we vary the initial network of the intermediary. In all simulations, the consumer network is initialized with a density of 0.2. This means that the average consumer initially has links to twenty percent of the sellers (producers and intermediaries). We then introduce intermediaries in the market and the initial network density of the intermediary varies from 0.2 (no difference in knowledge between intermediaries and consumers) to 1.0 (complete knowledge about the market for the intermediary). Table 5.1 gives an overview of the parameter settings used in the experiments. We have performed a sensitivity analysis on the parameter values and found that results are robust for different parameter settings. Furthermore, we have performed 25 runs of each parameter configuration used. The results that are shown in the next section are averages over 25 runs. All experiments were initialized with 10 producers, 40 consumers and 1 intermediary. As described above the expertise level of the intermediary and the pricing mechanism of the producers were varied during the experiments.

We have developed an agent-based simulation environment for testing and visualization of electronic trade networks. A screen shot of one of the output windows of the system is given in Figure 5.3. The system displays the architecture of the trade network at a certain point in time as well as the links that were used to purchase the good. On the right a graph monitoring average consumer utility is shown. All economic and EA parameters can be adjusted by the user, the system can also run in batch mode allowing for more extensive simulations and statistical analysis of the results. Agent-based simulation makes it possible to investigate many new scenarios that may arise as electronic commerce increases. Agent-based computational economics studies economic phenomena as they emerge from interactions between individual, boundedly rational agents. Simulations can give us valuable insights in the market structures that will arise. The simulations discussed in this chapter are a first step in that direction.

5.5 Results and discussion

5.5.1 Producers with a downward sloping price demand curve

The price structure used by the producers in this scenario presents a coordination problem to the consumers. Consumers get the lowest price if

Parameter	*Value*
Economic model parameters	
Producers	
Number of Producers	10
Maximum price	12.0
Minimum price	0.0
Producer price setting mechanism	
Downward sloping demand curve	$P = a - b * Demand^{exp}$
a	12
b	0.3
Derivative follower	$P = P * (1 + (dir * step\ size))$
direction	$\{-1, 1\}$
step size	0.1
Consumers	
Number of Consumers	40
Number of Consumer Types	10
Link costs	1
Error constant ϵ	0.05
Initial network density consumers	0.2
Intermediaries	
Number of Intermediaries	1
Link costs	1
Initial network density intermediaries	0.2...1.0
Intermediary markup	5%
EA parameters	
EA	Simple GA
Mutation rate	0.02 per bit
Crossover rate	1.0
Size of strategy base/population size	20

Table 5.1: Economic parameter values for consumers, producers and intermediaries and the evolutionary algorithm parameter values.

they all purchase the information good from the same producer. The efficient (efficiency here refers to minimum cost or maximum profit) outcome occurs when all consumers maintain one and only one link to the cheapest producer. When a producer has succeeded in attracting a customer base of a certain threshold size, he keeps attracting new customers because he offers the lowest price. Furthermore, when the consumers have solved the

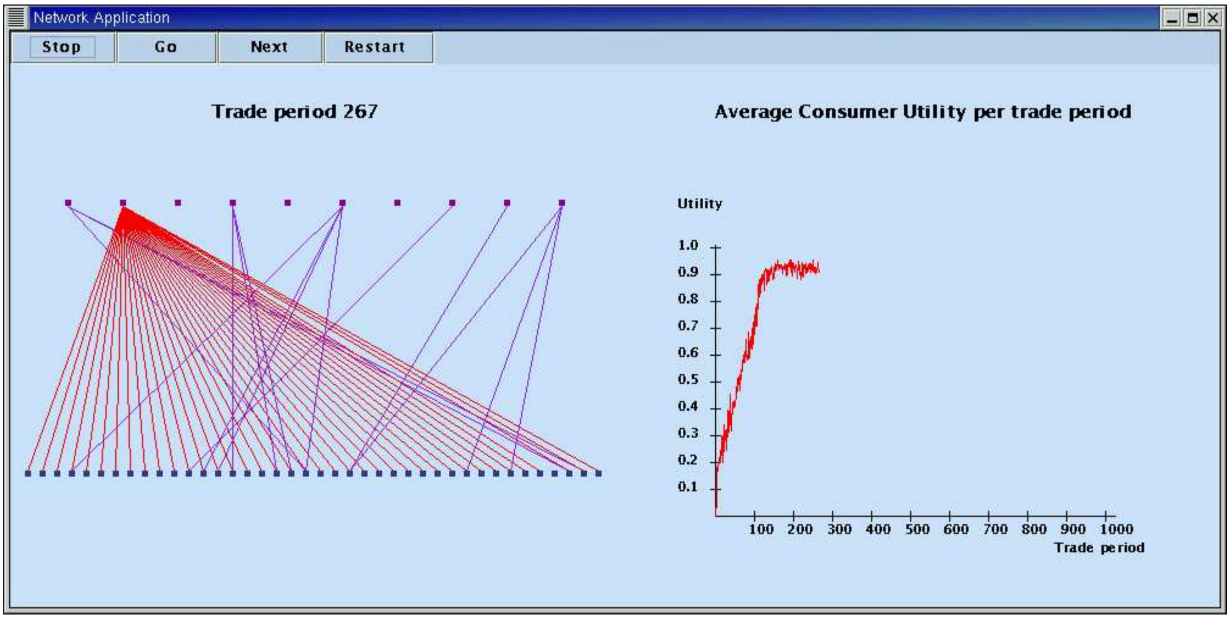

Figure 5.3: Screenshot of the simulation environment.

coordination problem there is no incentive to change their strategy, or even to maintain outside options, so the structure of the trade network as well as the price dynamics stabilize. Figures 5.4 and 5.5 illustrate these effects.

Figure 5.4 shows average consumer utility for different levels of the initial expertise level of the intermediary. The figure shows that consumer utilities are highest when no intermediaries are present in the market. The fraction of sales that occurs through the intermediary is plotted in Figure 5.5. As the intermediary starts off with a better network he attracts more consumers. However, the intermediary is not able to maintain this competitive advantage. This is caused by the fact that at the beginning of the simulation there is a lot of variance and uncertainty—prices vary, search strategies are not yet learned and demand is greatly dispersed. It is not yet obvious which producer will 'win' (path-dependent). Under these circumstances, the intermediary presents an attractive alternative. When price dynamics stabilize, it becomes easier for the consumers to find a good strategy and most consumers learn that they can obtain higher utility by direct trade instead of intermediated trade.

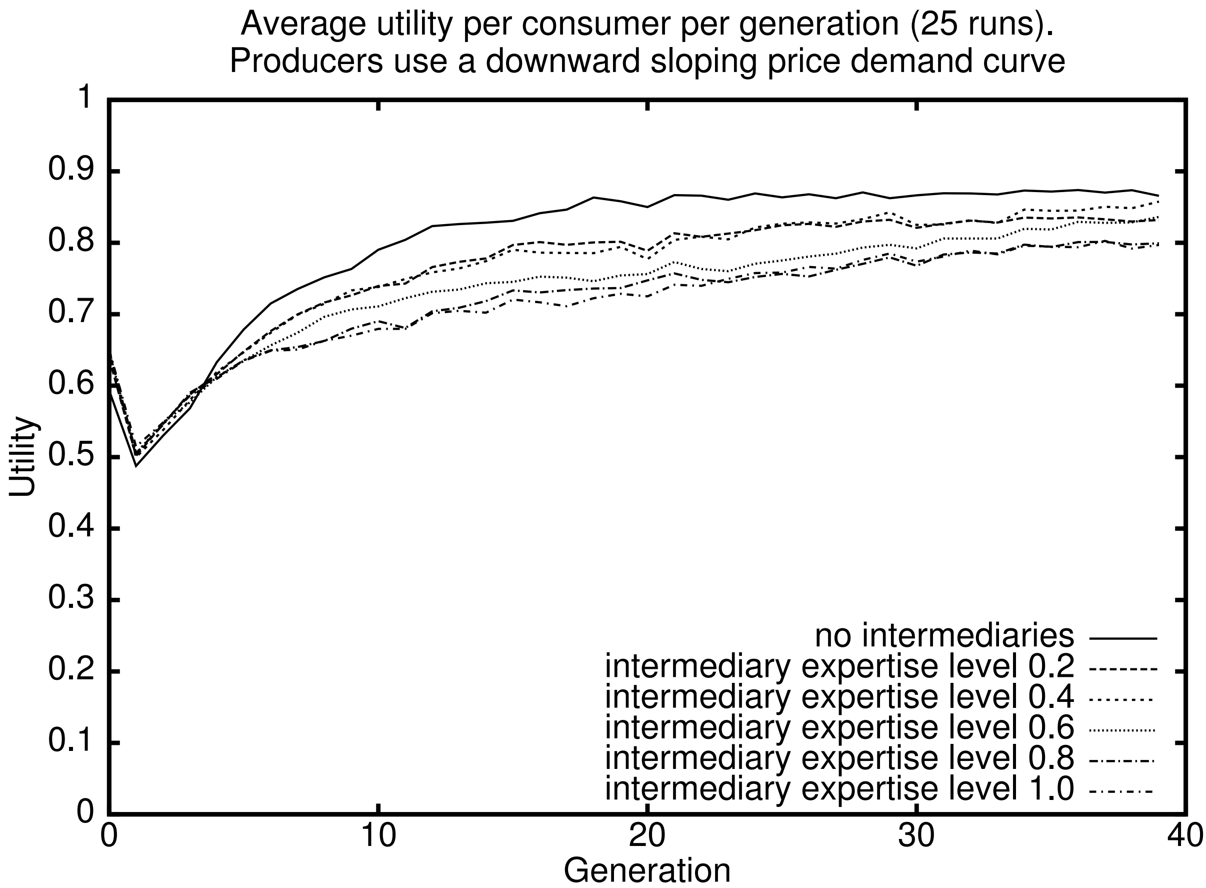

Figure 5.4: Averages consumer utilities over 25 runs for 40 update cycles of the evolutionary algorithm. Results are given for different initial expertise levels of the intermediary.

A comparison of Figure 5.4 and 5.5 reveals that the increase in average consumer utility coincides with a decrease in intermediary activity. This is an indicator that the intermediary increases market inefficiency. Average consumer utility increases over time, which demonstrates that the evolutionary algorithm generates strategies that solve the consumer coordination problem and learn to bypass the intermediary. Note that even when the quality of the initial network of the intermediary is equal to the quality of the consumer network, the intermediary still has some market share. This is caused by the randomness in the system; consumers make mistakes, and not all newly generated search strategies are an improvement. The experiments using these price structure show that consumers are able to find the optimal solution when dynamics are simple. Furthermore, under these conditions intermediaries are not part of an efficient market structure. However, when prices fluctuate, intermediaries are a stabilizing factor and are able to attract some customers. These effects of consumer learning can be observed even when initially all consumers have a link to the intermediary, see Figure 5.6. The experiments conducted using the derivative follower algorithm are described below.

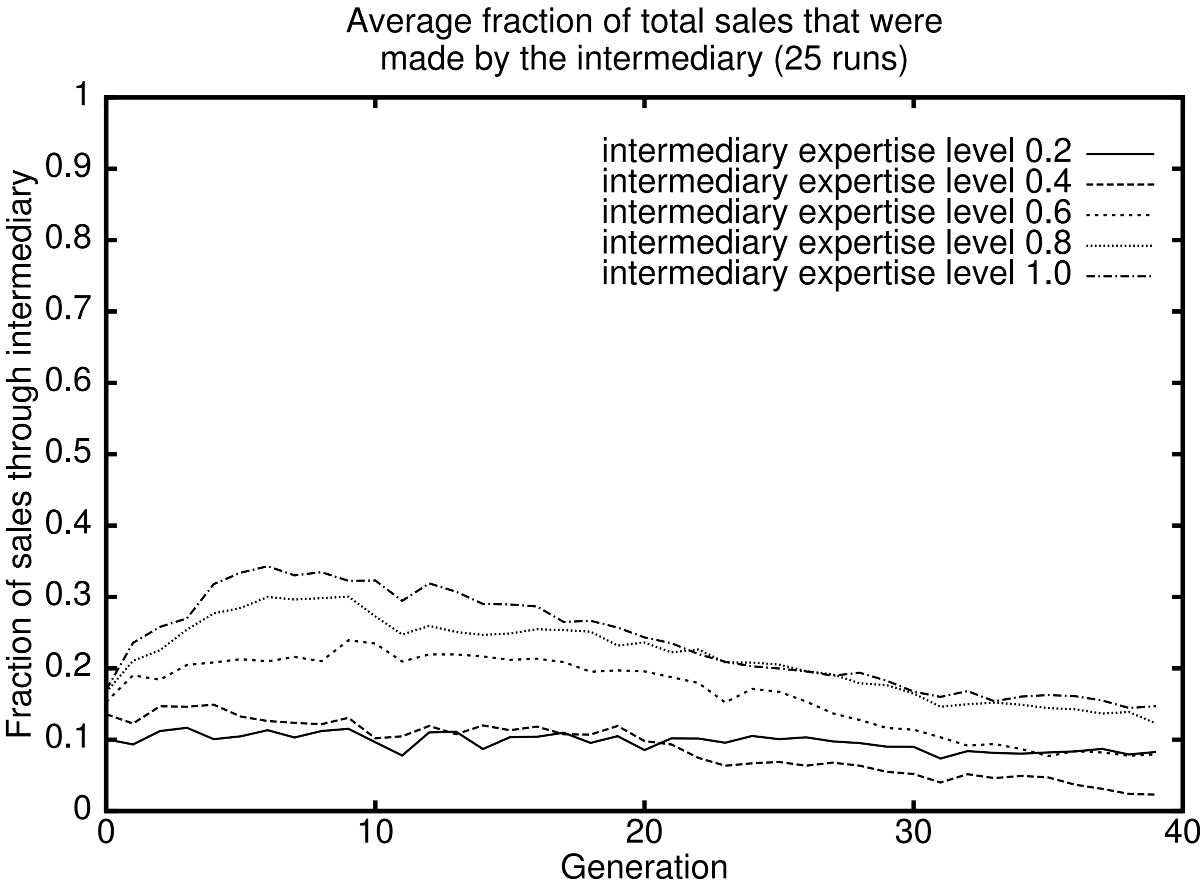

Figure 5.5: Averages fraction of sales by the intermediary over 25 runs. Results are given for different initial expertise levels of the intermediary.

5.5.2 Producers using the derivative follower pricing algorithm

Price dynamics caused by the derivative follower (DF) algorithm are more complex than the price dynamics discussed above. Producers continually seek to increase their profits by adjusting prices. The consumers now face a different problem, a trade off between the number of links they maintain

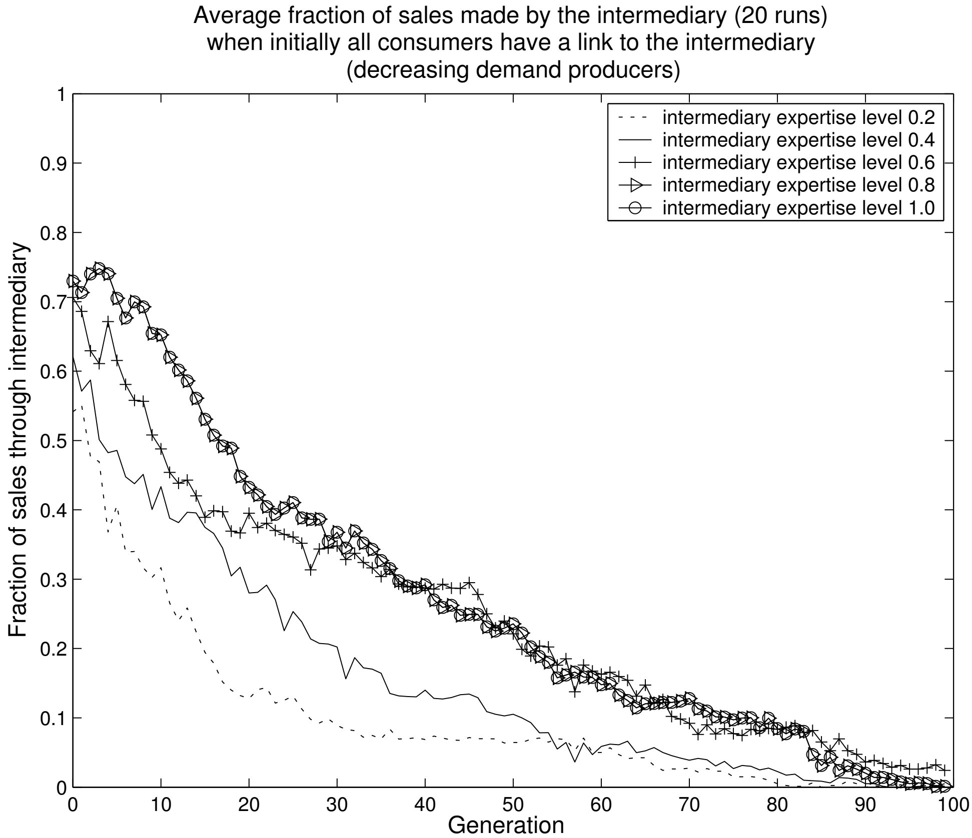

Figure 5.6: Averages fraction of sales by the intermediary over 25 runs when producers use a downward sloping demand curve and all consumers initially have a link to the intermediary. Results are given for different initial expertise levels of the intermediary.

and their total utility, so it is no longer optimal in the long run to maintain just one link. In this scenario a buyer may want to have several trade possibilities, so he can switch when his current supplier raises his price. The results of the experiments are given below. Figure 5.7 shows the average consumer utility when producers use a DF pricing strategy, while Figure 5.8 plots the fraction of sales that occur through the intermediary. Like in the experiments described above, we see that the fraction of intermediated sales increases when the quality of the initial network of the intermediary increases. However, this does not coincide with lower average consumer utility (see Table 5.2). In fact, consumer utility remains fairly constant after an initial training period. This indicates that there is a profitable niche for the intermediary and that intermediaries can play a role in an efficient market structure. If we take a closer look at the trade patterns that arise, we see that the intermediary takes over the search function from the consumers (during the turbulent initial part of the simulation) at the cost of a slightly higher price. Initially consumers try to form and maintain many links to different producers (network density increases from 0.2 to 0.5) but after 10 generations many consumers maintain only a link to the intermediary. After 20 generations the fraction of intermediated sales decreases again and consumers learn high utility strategies for direct trade.

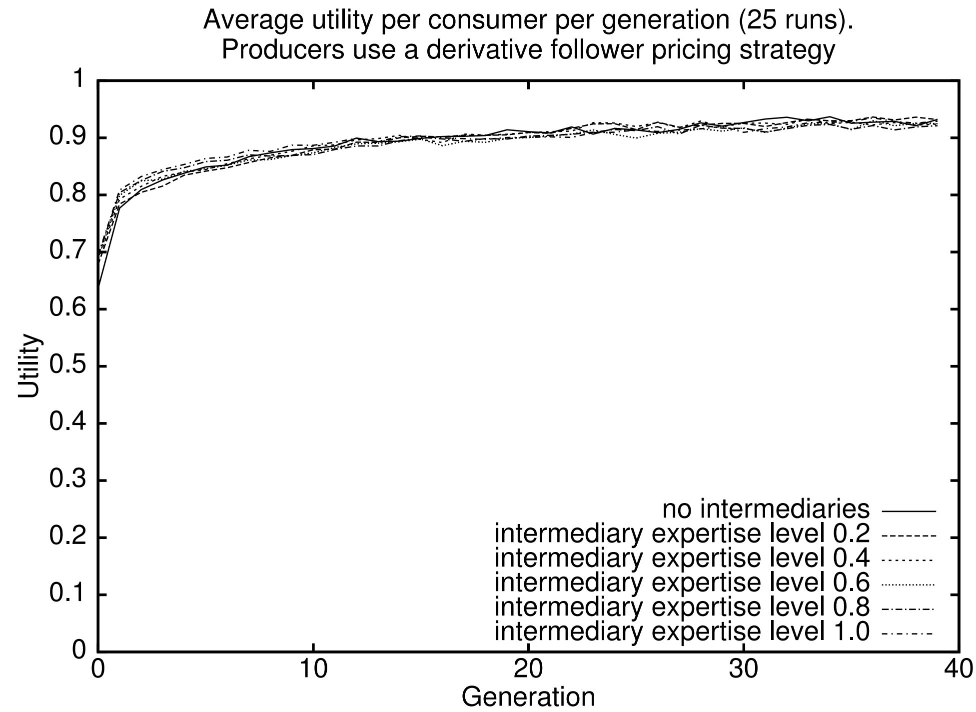

Figure 5.7: Averages consumer utility over 25 runs when producers use a derivative follower pricing strategy. Results are given for different initial expertise levels of the intermediary.

Again we see that the same effect still occurs if initially all consumers have a link to the intermediary, see Figure 5.9.

These experiments show that expert intermediaries can provide a valuable service to consumers when search costs are high. Furthermore, their activities have a stabilizing effect on price dynamics and the structure of the resulting trade network. However, the intermediary has to invest in a high quality network in order to attract customers. As market dynamics stabilize it becomes easier for consumers to find a good search strategy

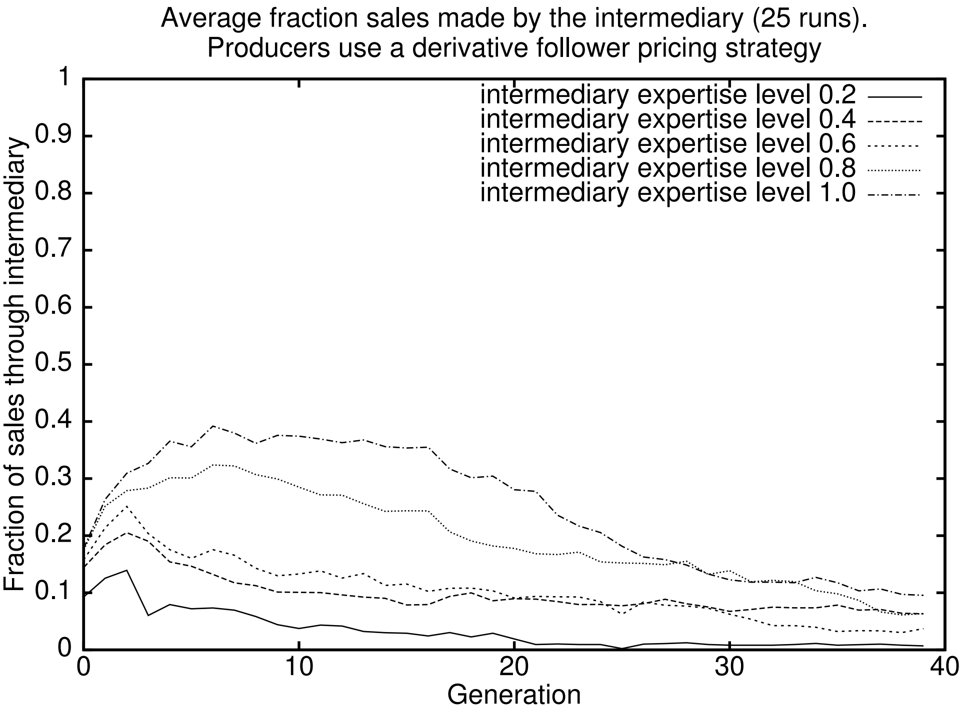

Figure 5.8: Averages fraction of sales by the intermediary over 25 runs when producers use a derivative follower pricing strategy. Results are given for different initial expertise levels of the intermediary.

Expert level Intermediary	*Fraction of Intermediated Sales*	*Average Consumer Utility*
no intermediary	0	0.89 (0.05)
0.2	0.03 (0.04)	0.89 (0.05)
0.4	0.09 (0.03)	0.89 (0.05)
0.6	0.10 (0.06)	0.89 (0.04)
0.8	0.19 (0.08)	0.89 (0.05)
1.0	0.25 (0.10)	0.89 (0.04)

Table 5.2: Averages over 25 runs of 40 generations when producers use a derivative follower pricing strategy

and bypass the intermediary. This even happens when initially all consumers have a link to the intermediary, as illustrated in Figure 5.9. That is consumers learn to bypass the expensive intermediary as they learn more about the structure of the market through experience. Electronic markets for information goods are characterized by frequent price changes, hence we

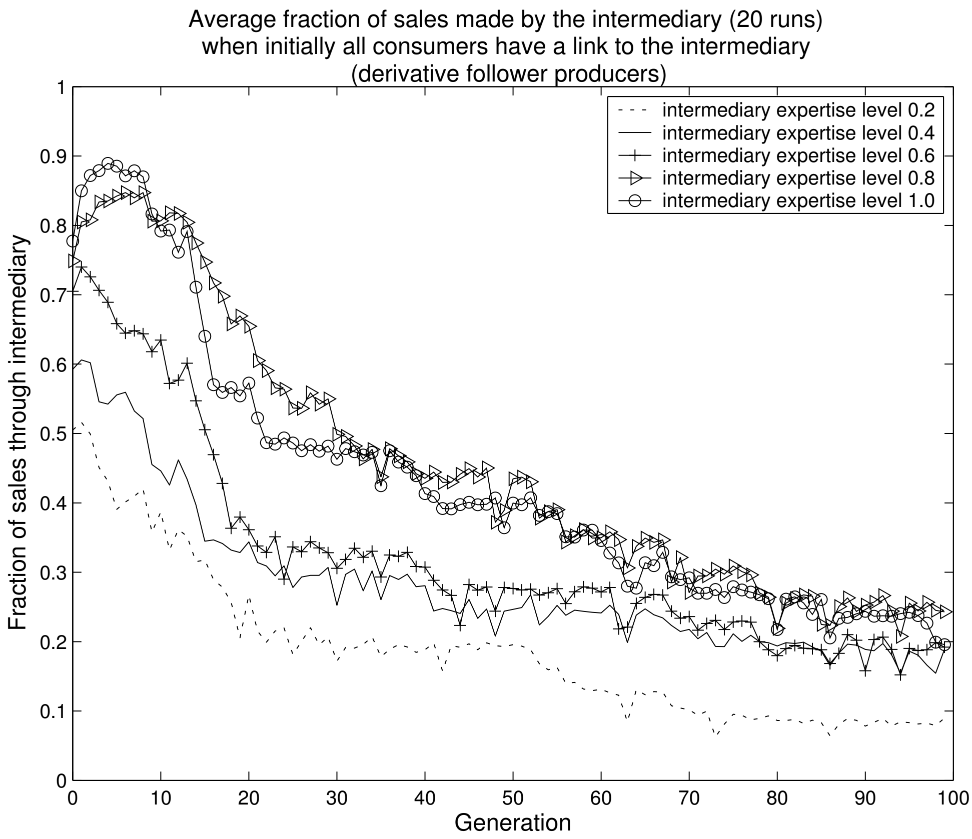

Figure 5.9: Averages fraction of sales by the intermediary over 25 runs when producers use a derivative follower pricing strategy and all consumers initially have a link to the intermediary. Results are given for different initial expertise levels of the intermediary.

expect that specialized search intermediaries will play a role in electronic trade networks. The fact that intermediaries can occupy a greater niche in the market when uncertainty is high, is confirmed if we consider random production in Section 5.5.3.

5.5.3 Random production

The effect of uncertainty in the market on the level of intermediated trade can be clearly illustrated by looking at the situation where production is random, see Figure 5.10. Here we see that well connected intermediaries are able to obtain and maintain a substantial market share.

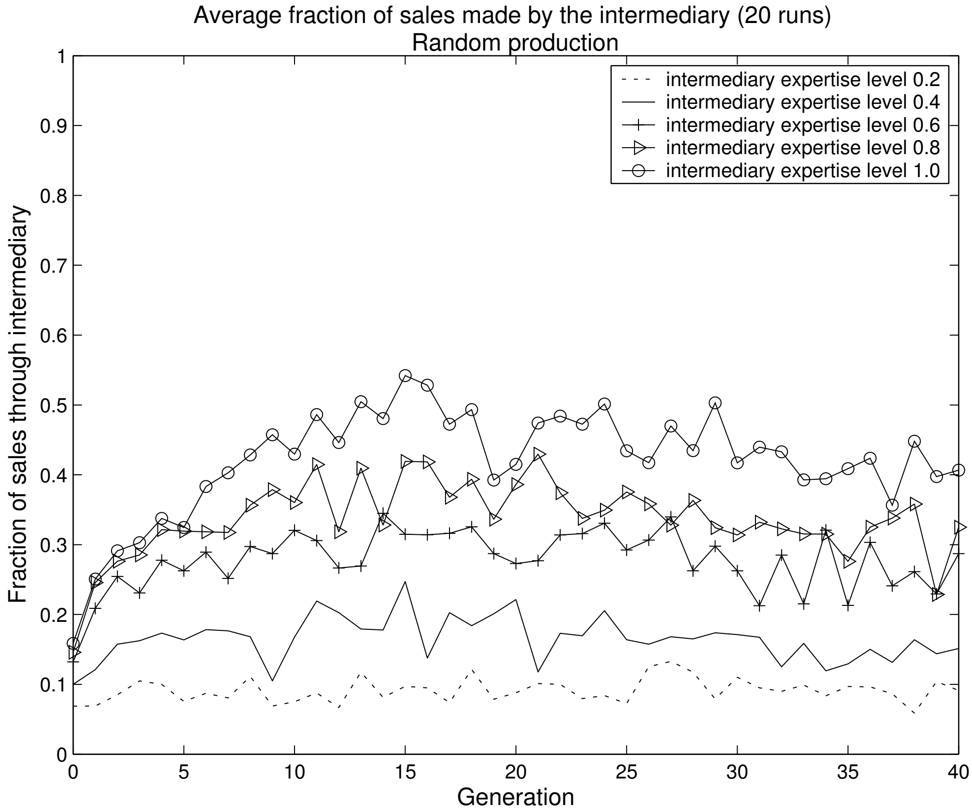

Figure 5.10: Averages fraction of sales by the intermediary over 25 runs when production is random. Results are different initial expertise levels of the intermediary.

5.6 Learning to become an expert intermediary

In the previous sections we have investigated the conditions under which expert intermediaries can obtain a market share in markets where consumers can also choose to buy directly from producers at a lower cost. The expertise of the intermediary was reflected by the fact that the intermediary has a better knowledge of the market than the average consumer. In this section we consider how an intermediary might gain such expertise. According to Spulber (1999) one reason why intermediaries have better knowledge than the average consumer is the fact that they simple perform much more transactions and thus have an increased opportunity to learn about the market. In order to test this hypothesis we have increased the *learning rate* (*lr*) of the intermediary, that is the strategy base of the intermediary is now updated more often than the strategy base of the consumers. Figure 5.11 presents the results for this model with *learning rate* 4 for the intermediaries and *learning rate* 1 for consumers.

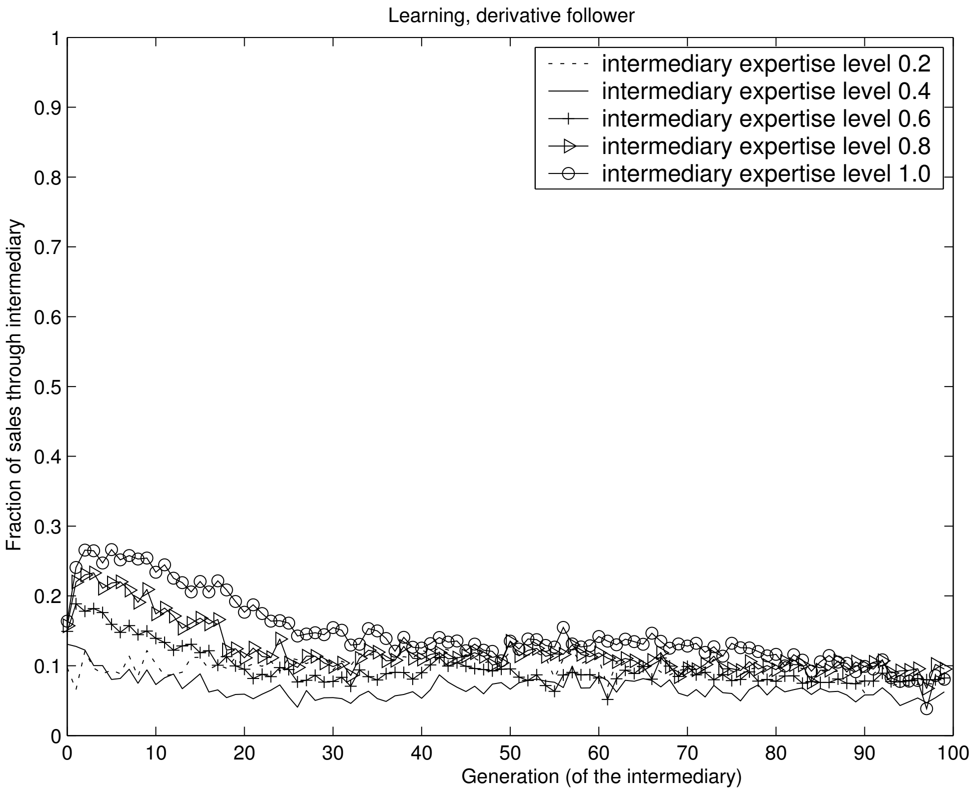

Figure 5.11: Averages fraction of sales by the intermediary over 25 runs when intermediaries (*lr 4*) learn faster than consumers (*lr 1*). Results are given for different initial expertise levels of the intermediary.

First we note that the intermediaries are able to obtain a higher market share than in the previous section although the same overall tendency can be observed (decreasing intermediary market share as the trade simulation continues). However, if we look at the individual runs this is no longer true. Without fast intermediary learning, all runs were similar, as is demonstrated by the low standard deviations in Table 5.2. When intermediaries have a higher learning rate than consumers this is no longer the case. More specifically, we find that in 20-25% of all runs intermediaries are able to obtain and maintain a substantial niche in the market. Two such runs are depicted in Figure 5.12. These results show that higher expertise levels of the intermediary can indeed be explained by a faster learning rate for those intermediaries.

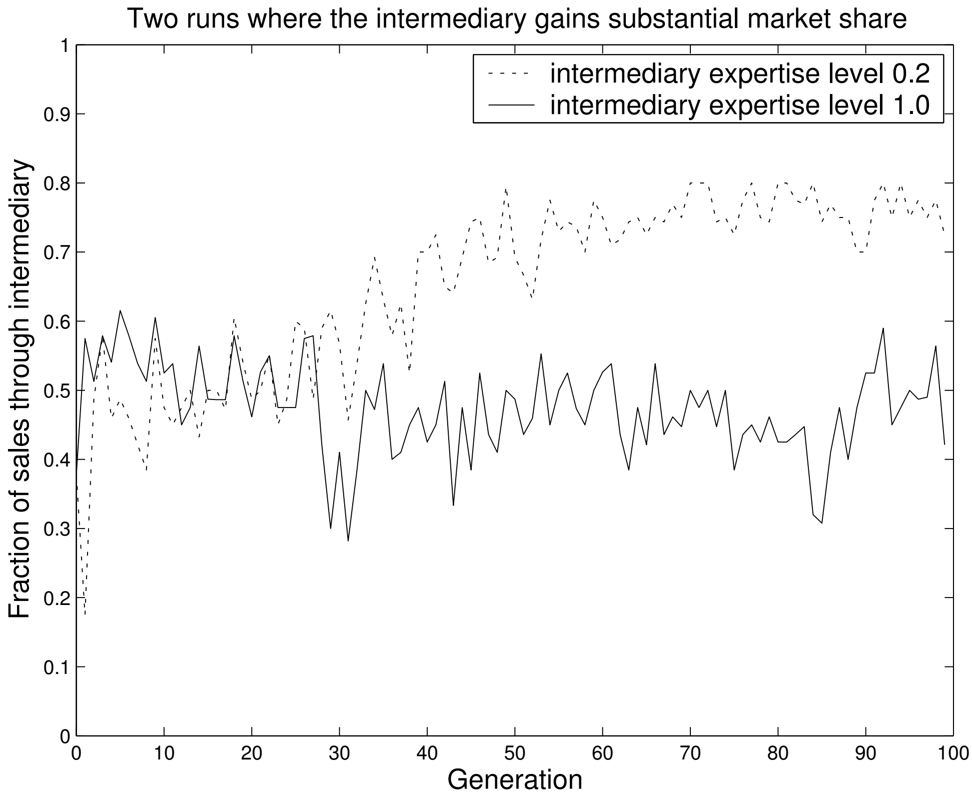

Figure 5.12: Two examples of individual runs where intermediaries succeed in obtaining and maintaining a substantial market share.

5.7 Conclusions

In this chapter we have addressed the question whether intermediaries will still exist and be able to make a profit if consumers can make direct connections with producers, as is often the case in electronic commerce. We have performed agent-based simulations to study the performance of intermediaries under different market conditions. We modeled an electronic trade network where an information good is traded over the network. Each trade period cost-minimizing consumers have to decide which links to form to sellers (i.e., producers and intermediaries), while the good can only be purchased if a link between the buyer and the seller exists. Links thus represent trading possibilities, a flow of relevant information (i.e., price quotes in our case between potential buyers and sellers) and the consumers have to make a strategic decision about which (costly) links to form. We have used an evolutionary algorithm to model the search and learning behavior of the buyers. Our main finding is that if market dynamics are sufficiently complex, intermediaries that have better knowledge about the market than the average consumer are initially able to increase their market share and make a profit.

In accordance with the theory our simulation showed that ultimately most consumers bypass the intermediary if direct trade is more profitable. We have investigated the conditions under which intermediaries can still make a profit and simulated different scenarios. Based on our simulation we can make the following observations. Intermediaries that are experts at finding the best price quotes can initially survive (and even increase their market share) in an electronic trade network where consumers can also form direct links to producers, although ultimately most consumers bypass the intermediary if direct trade is more profitable. However, when producers change their prices in an adaptive way to increase their profits (derivative follower) consumers face a trade-off between maintaining many costly links and getting the best price. In those types of markets there is a profitable niche for the intermediary and we find that many consumers choose to trade through the intermediary. Consumers compensate for the higher purchase price by maintaining less links—this has a stabilizing effect on the architecture of the electronic trade network. And finally, the expertise levels of the intermediary can be explained by the fact that intermediaries learn faster than consumers because they perform more transactions.

These findings suggest that intermediaries will still be needed when consumers trade on complex dynamic electronic trade networks. Agent-based simulation thus makes it possible to investigate many new scenarios that may arise as electronic commerce increases. Simulations can give us valuable insights in the market structures that will arise.

Chapter 6

Diffusion of information on a social network

6.1 Introduction

Network economics holds the view that individual actions and, in turn aggregate outcomes, are in large part determined by the interaction structure between heterogeneous economic agents. The structure of the social network between agents is particularly relevant if we consider the market for so-called fashion goods, i.e., products for which consumer value depends strongly on the consumption decisions of others. For the diffusion of such fashions, "word of mouth" plays an important role, consumers tell each other about the product and with that it gains popularity. Firms operating in such a market need to take the properties of the social network between consumers into account when they make marketing decisions.

The central question of this chapter is whether firms can learn about social network structure and consumer characteristics when only limited information is available, and use this information to evolve a succesful marketing strategy. We consider the decision problem of a firm that wishes to choose an advertising strategy to successfully introduce a new product in a network of consumers. Consumers base their purchase decision on the behaviour of other consumers. It may be of critical importance for this firm to get insights into the structure of the social network. However, marketing research charting consumer relations is typically expensive and difficult to perform. We investigate whether firms can learn targeted advertising strategies, taking the social network structure into account if only aggregate sales data are available. Is such a strategy of targeted advertising more effective than a random advertising strategy? And does the best strategy change with respect to different topologies of networks? To address these question we use insights from the existing literature on diffusion phenomena in networks, both in economics and epidemiology. We extend existing models to allow for more realistic modeling of consumer behaviour and we study the diffusion of the innovation through agent-based simulation.

The simulation model allows us to study how word of mouth about an innovation (i.e., a new product or idea) spreads throughout a social network. More specifically, we look at the situation where the decisions of the consumers are strongly determined by the decisions of their neighbors in the network. There are two major situations in which this type of behavior can be considered a good strategy for consumers. First, when agents do not possess any reliable information about the new good, they look at the consumers around them as a way to extract information. Some of the other consumers may hold private information about the new good or simply, in the case of a dynamic setting, they may have purchased the good already and then inform the people to whom they are connected . Second, consumers may in fact assign a relatively low weight to the actual characteristics of the good itself and instead attach a higher value to the number of people purchasing the good.

The rest of the chapter dicusses the details of our model and a discussion of obtained results. Section 6.2 provides an overview of the relevant literature. Section 6.3 describes the model used for our agent-based simulations. The experimental setup is described in Section 6.4. Results are given in Section 6.5 and Section 6.6 draws conclusions.

6.2 Relation to existing literature

The literature on social interactions has throughly studied how a preference for conformity can explain herd behavior in consumers or the emergence of fashion, fads and customs, see (Banerjee 1992, Brock and Durlauf 1997, Bikhchandani *et al.* 1992, Bernheim 1994). The diffusion of a new product, or innovation in a network, often follows a gradual pattern. In a first stage a few consumers (the innovators or early adopters) adopt, then consumers in contact with them adopt, then consumers in contact with those consumers adopt, and so forth until the innovation possibly spreads throughout the network reaching also the more conservative consumers (or 'followers'). When the diffusion reaches the majority of the network, we call this a cascade. Such a cascade is associated with the commercial success of the new product. If innovation does not succeed in completely taking off, the firm may decide to file the product as a failure. Thus, we explore whether or not the new product diffuses to the largest part of the network together with the time actually needed for the critical diffusion.

The initial positive feedback mechanism described above, may be offset by a tendency of some consumers to distinguish themselves from the dominant tendency. A form of negative externality may then make a few individuals revise their purchasing decision with the effect of limiting the diffusion of the innovation to the entire system. In this paper we will study both positive and negative feedback. We explicitly take into account that the consumption behavior of other people can have a positive externality ('people like to imitate other people'), but also negative feedback ('people like to be special'). One way for these two opposing forces to co-exist is to have 'imitation effects' being replaced by a tendency towards heterogeneity as soon as some critical level of diffusion of the product is attained.

A further useful distinction in the study of consumption behavior concerns the intrinsic purchasing attitude of consumers. Some may be considered 'innovative' consumers. They are the ones who first choose a new product and are basically responsible for its initial diffusion. Innovators usually represent a small portion of the set of consumers. Most people are instead simply 'followers', in that their strategy is to choose the novel good only after someone else has already tried it. Their strategy is a more conservative one and they are usually responsible for the actual spread of the innovation. These two consumption attitudes have both been shown to play a role in the diffusion processes of many products. One example is new software products (Hippel von 1988), which are usually tried by a restricted group of experimental users and later, eventually, chosen by a higher number of more conservative consumers. In our model we allow for different consumer purchasing attitudes.

Advertising campaigns are costly and different strategies can be employed by the firms. Firms are boundedly rational and are not fully aware of the structure of the communication channels among consumers. We consider the situation where firms only have aggregate sales data. This can be an example for products that are sold over the Internet (for example ringtones) where the firms do not have any information about the characteristics of the consumers. Figure 6.1 illustrates how the topology of the network may ifluence marketing strategies. A so-called 'Star Network' (see Figure 6.1) is easiest to penetrate for the firm. If the firm targets the center of the star, word of mouth about the product reaches all agents in the network very fast. The most difficult network for the firm is the regular network, where pathlengths are very long (Figure 6.1). However, if consumers differ in their tendency to adopt new products there is a trade-off between targeting consumers that are well connected (such as the center of a star) versus targeting consumers that have certain characteristics, such as for example the 'opinion leaders' in Valente and Davis (1999). Our model allows us to investigate this trade off and analyze good advertising strategies for the firm under different circumstances regarding both the social network topology and consumer characteristics.

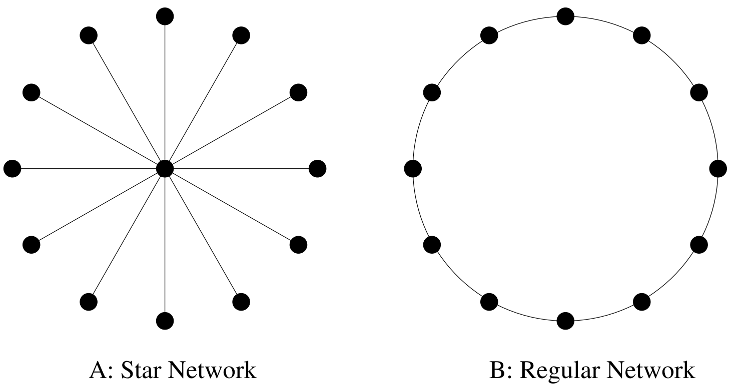

Figure 6.1: A Star Network (left) and a Regular Network (right)

Next, we relate our model to the recent literature on diffusion in networks and we provide a few empirical examples that we want to study. Models of informational cascades analyze how the decisions of first movers in a sequential decision problem can lead to herd behavior (a cascade) of the whole system of agents, see Banerjee (1992). Herd behavior occurs when agents do not use any private information but instead only value the information provided by the decisions taken by the other agents. It should be noted that in these models, decisions are made sequentially, so that agents look at the actions taken by the agents who decided before them. In this chapter we take a different perspective and analyze the diffusion of information in networks of consumers, as in Watts (2002). A number of economic models exist, that investigate how innovations diffuse in social networks (Ellison 1993, Brock and Durlauf 1997, Young 2003). The focus is on local interactions and in particular on positive feedbacks. An exception is the work of Cowan *et al.* (1997) were negative externalities are also considered, but in their model the influence of the topology of the social network is not investigated. Famous studies have been concerned with studying the choice between competing technologies, the specific properties of 'network technologies' entailing compatibility issues and the local self-reinforcement processes that allow rapid dominance of some new standard or product or institution (Arthur 1994, David 1985, Katz and Shapiro 1985).

Most of the cited theoretical contributions consider regular lattices, defined as symmetric structures where all nodes have the same 'degree', i.e. the same number of links departing from each node. Instead, empirical evidence suggests that the degree distribution in social networks is highly right skewed. Specifically, social networks often display the properties of small-world graphs. As originally defined in (Watts and Strogatz 1998), these networks are obtained from regular lattices by rewiring randomly chosen edges. In another version (Newman and Watts 1999), they are graphs whose vertices are connected together in a regular lattice, with the addiction of a small number of connections bridging randomly chosen vertices. Small-world graphs show a higher level of clustering than random graphs. Their pattern is not as ordered as in a regular lattice, but they preserve short average path lengths, proportional to the logarithm of system size. Small-world networks are thought to be a good model for many types of real social networks.

Many analytical results for diffusion phenomena on networks are available from the epidemiological literature that studies the influence of the topology of the underlying network of individuals on the dynamics of disease propagation, see for example (Strogatz 2001). These models actually correspond to percolation problems on graphs. Specifically, they investigate threshold values for actual epidemic outbursts as opposed to limited localized spreading. Some models (Callaway *et al.* 2000) also explore the effectiveness of vaccination strategies that try to inactivate some of the nodes in the network. One limitation to the application of the mentioned exact solutions is that they are valid for very large networks, but they do not hold for small networks. They also require rather strict assumptions, see (Moore and Newman 2000) for a definition of the general site and bond percolation problem. There are some analytical results on models of diffusion on networks, see for example (Newman 2000). However, as the agents become more complex and we introduce a firm with a slightly more sophisticated advertising strategy, results soon become intractable. Therefore, we use an agent-based model to simulate the information and product diffusion process.

6.3 The diffusion model

We consider the diffusion of an innovation over a network of consumers. The goal of this research is to investigate whether a firm can learn directed advertising strategies that increase the size and speed of the diffusion. Consumer valuation for the product depends only on the fraction of its neighbours that has already purchased the innovation. The consumers are connected through a social network and the innovation spreads through this consumer network. Furthermore, the firm that is pushing the product can advertise and thus inform consumers about the existence of the product, however advertising is costly and firms can only target a limited number of consumers. Therefore firms have to learn advertising strategies that will cause the most effective diffusion of the product. Figure 6.2 gives an overview of the simulation model, the components will be discussed in more detail below.

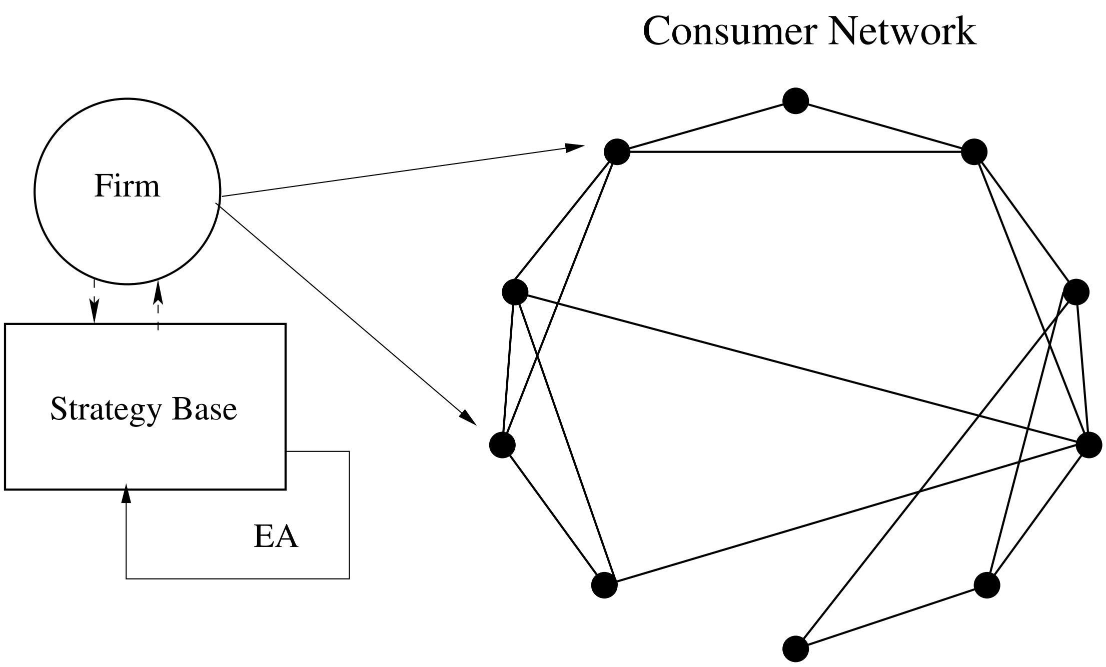

Figure 6.2: Outline of the simulation model. In this example the consumers are connected through a small world network structure.

6.3.1 Consumers

In this chapter we look at products for which the consumer value depends strongly on the number of other consumers that use the product. If consumer value increases when the number of other consumers that has adopted the innovation becomes larger we call this **positive (network) externalities**, see for example Katz and Shapiro (1985), Farrell and Saloner (1986). **Negative (network) externalities** occur if consumer value decreases as number of adopters becomes larger. In this chapter we investigate both positive and negative externalities. For example, if we consider fashion goods it might be unrealistic to assume that the positive externality continues to grow until all consumers use the product. Consumers may have a desire to be fashionable on the one hand, while they also want to be special on the other hand, this is particularly true for the innovator who always adopts the latest fashion or fad. These innovators may move on to the next fashion while more conservative consumers are still following a previous hype. To be more precise, we also investigate the situation where the innovation is no longer attractive once it becomes adopted by too many people.

A consumer is characterized by its *exposure threshold (et)* and by its neighbors. These threshold and the structure of the social network are exogeneously given at the start of the simulation. Consumer that have already adopted the innovation talk about it to their neighbours, so consumers are aware of the purchase decisions of other consumers they are linked to. The *exposure threshold* models the tendency of a consumer to adopt the new product. A consumer with an exposure threshold of 0.5 will buy the product if and only if at least half of its neighbors have also bought the product. Note that this threshold is more easily reached in sparse (but connected) networks than in dense networks. In Section 6.5.4 we also consider negative externalities, that is consumers want to be fashionable but still special. In this case consumers have both an *exposure threshold (et)* and an *over-exposure threshold (oet)* threshold. Consumers stop using the product if the fraction of their neighbors that has adopted the product exceeds their over-exposure threshold. This *over-exposure threshold* is used by the agents to discriminate between innovations that are attractive and innovations that are no longer fashionable because their user-base has become too large. The decision rule for the consumer can be characterized as follows:

Each trade period consumers take the following steps:
1. *Consumers* who have already adopted the innovation talk about the product to their neighbors
2. A *Consumer* decides to adopt the product if:
Word of mouth it receives from its neighbors exceeds its Exposure Threshold.
and (in case of negative externalities)
Word of mouth it receives from its neighbors does **not** exceed its Over-Exposure Threshold.

6.3.2 Firms

Firms can engage in an advertising campaign in order to try and increase the size and the speed of the diffusion of their product (the innovation). Firms do not have any a priori knowledge about the structure of the consumer network and have to learn which advertising strategies are best. They have to choose which consumers to target in order to ensure fast diffusion of their new product. Advertising campaigns are costly and different strategies can be employed. Firms are boundedly rational and are not fully aware of the structure of the communication channels among consumers. As a result, they are likely to explore a range of targeted advertising strategies. To model this search and learning process of the firms we use a simple genetic algorithm which will be described in more detail below. An advertising strategy specifies which consumers are targeted at time 0. The success of an advertising strategy depends on (1) the number of consumers that have adopted the innovation after a specified period of time when that strategy was used, and (2) the cost of the advertising campaign. The number of products sold is the only information coming from the market that the firm obtains at the end of each period. The algorithm used by the firms can be summarized as follows:

Firms take the following steps:
1. Select an advertising strategy
2. Calculate fitness of the strategy:
Fitness = Sales - Advertising costs
3. Update Strategies:
Update strategy base using a GA
4. Go to **1.**

We investigate the learning capabilities of the firms with respect to two scenarios. First we examine whether firms can learn the best directed advertising strategy when a fixed number of consumers is targeted, i.e., the firm has a fixed advertising budget. This allows us to compare the results to a random advertising strategy targeting the same number of consumers.

Second, we investigate whether firms can learn to decrease their advertising expenses, i.e., to minimize their costs. We use different genetic operators to update the strategy base in each scenario. These GA operators are described below.

6.3.3 The genetic algorithm used by the firms

Strategies are updated by a genetic algorithm. A strategy is represented by a bitstring of length l, where, in this case, l is the number of consumers in the network. If the i^{th} bit on the chromosome equals 1 this means that the strategy represented by the chromosome targets consumer i (if the bit equals 0 the consumer is not targeted). We use a simple genetic algorithm to update the strategies for the consumers. Since we consider consumers with a fixed budget however, we use adapted mutation and crossover operators (see below) in order to ensure that a strategy targets exactly m consumers, where:

$$m = \left\lfloor \frac{\text{Budget } b}{\text{Marginal cost of advertising } c} \right\rfloor$$

The fitness of a strategy is determined solely by the sales after t timesteps, where t is the **training time**. The trainingtime t was taken between 10 and 50 timesteps. We have adapted the crossover and the mutation operator in order to ensure that a chromosome contains exactly m ones. Each chromosome thus consists of l bits representing the consumers, m of those bits are set to 1 and we will call thos bits the *1-bits*, similarly, *0-bits* are the $l - m$ bits that are set to 0. The algorithms for the adapted operators are given below, which we call one-preserving mutation and one-preserving crossover respectively.

One-preserving mutation:
For each chromosome that is selected for mutation:
1. Randomly select one of the *1-bits*
(one of the currently targeted consumers)
2. Change this bit to **0**
3. Randomly select one of the old *0-bits*
4. Change this bit to **1**

One-preserving crossover:
For two parent chromosomes (parent1 and parent2) we create two offspring chromosomes (offspring1 and offspring2):
0. Set all offspring bits to zero.
First we consider the k *1-bits* that the two parents have in common
1. (Both strategies agree on those *1-bits)*
These k *1-bits* are copied onto the offspring chromosomes
(as would be the case with a regular crossover operator)
Second we consider the $m - k$ remaining *1-bits*
2. Select a random number C_{cross} between $[0..m - k - 1]$
C_{cross} is the crossover counter
3. Copy the first C_{cross} *1-bits* from parent1 to offspring1
Copy the remaining bits from parent1 to offspring2
4. Copy the first C_{cross} *1-bits* from parent2 to offspring2
Copy the remaining bits from parent2 to offspring1

In words, one-preserving mutation replaces a currently targeted consumer and replaces it by a consumer that is currently not targeted. Note that one-preserving mutation thus works on the entire chromosome, instead of on a single bit. An example illustrating both operators is given in Figure 6.3 for a chromosome of length 6. The first parent chromosome thus specifies that consumers 1, 3 and 5 receive targeted advertising (for example in the form of a free sample product).

6.3.4 The social network

We investigate three types of social network architectures; the k-regular network where all consumers have exactly k neighbors. In this chapter

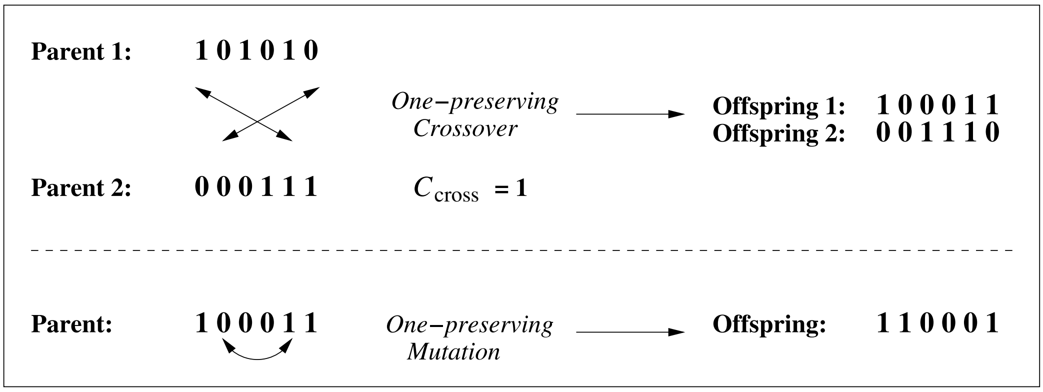

Figure 6.3: Example of one-preserving crossover and mutation

we restrict ourselves to regular networks that are modeled as a (one dimensional) ring lattice where each agent is connected to its k nearest neighbours by undirected edges. The random network, where links between consumers are constructed randomly, and the Small World network. A Small World network is constructed by taking a regular network and then rewiring a small fraction of the links (usually between 1 and 10 percent of all links (Watts and Strogatz 1998). This fraction of links is called the **rewiring constant** (rc). The **degree** of a network is defined as the (average) number of neighbors of a given consumer in the network. For k-regular networks the degree thus correpsonds to k. For Small World and random networks, the number of links per consumer may vary and the degree of such networks describes the average number of neighbours of a consumer.

In this chapter we consider a social network of 1000 consumers. This is sufficiently large to observe the dynamics that occur for different network topologies (Cowan and Jonard 2004). A **cascade** on the network occurs if, starting from a small fraction of the consumers (the initially targeted consumers), the diffusion spreads out to a large part of the population. In this chapter, we define a cascade as a diffusion that reaches at least 80 percent of all consumers, starting from a small initial number of targeted consumers. When the diffusion reaches all consumers in the network we call this a **global cascade**. Note that such a global cascade may not always be possible since random and Small World networks can be disconnected (see Section 1.2.1).

6.3.5 Diffusion dynamics

Diffusion dynamics are driven by the topology of the network, the advertising strategy that is used by the firms and the consumer characteristics. At each timestep in the model, the agent actions described above are executed resulting in the model cycle described below. Initially, the number of consumers as well as their characteristics, the network topology and the advertising strategy used by the firm are exogeneously given. The outcome of the agent-based simulations now depends on those initial conditions and the agent interactions.

Each cycle:
1. *Firms* choose an advertising strategy
(from their strategy base)
2. *Consumers* who have already bought the product
talk about the product to their neighbors
3. *Consumers* decide whether to (still) adopt the product
Go to 2. (Repeat for a given number of timesteps)
4. Firms calculate their profits
5. Firms update their strategies for the next period

We are especially interested in two aspects of the diffusion dynamics: (1) the size of the diffusion, that is how many consumers eventually adopt the innovation, and (2) the speed of the diffusion, that is, how long it takes to reach this diffusion level. An important measure in assessing the properties of the diffusion is the **critical diffusion threshold**. The **critical diffusion threshold** in a network of homogeneous consumers is the highest *exposure threshold* for which a cascade is observed. All thresholds below this critical threshold will lead to cascades on the network.

6.4 Experimental setup

The goal of the experiments is to investigate whether firms can learn directed advertising strategies to increase the diffusion of their products. Furthermore, we want to investigate the properties of such strategies. The social network and the consumer thresholds are exogeneously given at the start of the simulation. We vary the topology of the network, with respect to degree and network architecture, as well as the exposure thresholds of the consumers, and study the diffusion process over time. We thus perform an agent-based computational study of the diffusion of an innovation over a social network. The diffusion dynamics are a result of the local interactions of autonomous agents over time. The model described above allows us to investigate the speed and the size of the innovation diffusion under different initial conditions. Figure 6.1 gives an overview of the parameter values that were used in the simulations.

Parameter	**Value**
Number of Consumers	1000
Degree	1-20
Rewiring Constant (rc) (Small World Network)	0.05
Exposure Threshold	0.0-0.5
Number of Initially Targeted Consumers	10
Diffusion Time (during learning)	10-50
Generations	20
Number of Strategies	50
Pone-mut	0.1
Pone-cross	1.0

Table 6.1: Parameter values used in the simulations.

6.5 Results and discussion

In this section we present the results of our simulations and compare the diffusion processes under random and directed (learned) advertising.

6.5.1 Homogeneous consumers

In this section we consider the diffusion of information over the network when consumers are homogeneous with respect to their exposure threshold. Figure 6.4 shows critical diffusion thresholds for the different types of networks when consumers are homogeneous with respect to their exposure thresholds and firms have a fixed budget. Each network consists of 1000 consumers. The degree of the network specifies the (average) number of neighbors each consumer has. Points signify the maximum threshold for which an informational cascade was achieved after 1000 timesteps starting from 10 nodes. The averages are over 20 runs that is 20 different initial networks were tested. The consumers that are initially targeted are different in each run. The learning time is 50 timesteps. In the fixed budget experiments we use the one-preserving operators discussed in the previous section. Note that under these conditions the advertising strategy only has to be optimized with regard to the network topology. If we look at Figure 6.4 we see that firms are able to learn effective advertising strategies. In the case of the regular networks such a strategy has to target consumers that are evenly spread out over the network in order to ensure maximum diffusion. Using such a strategy, firms are able to achieve informational cascades even if consumers have a low tendency to buy the product (i.e., a high exposure threshold).

If we restrict our attention to the random strategies, we can observe two different regimes; an upper phase for dense networks, a lower phase for sparse networks. In a sparse network, diffusion of information is limited by the global degree of the network, but cascades occur even when consumers are quite 'resistant' to being convinced by their neighbors, i.e. when their exposure threshold is high. On the other hand, if the network is sufficiently dense, the propagation is limited by the stability of individual nodes. In this case the critical exposure threshold is significantly smaller. Most nodes have a large number of neighbors, but with a random strategy it is unlikely that all these neighbors are buying the product at time 1, so the initial perturbation may not be able to diffuse at all. Note that critical thresholds are similar for small-world and regular networks. We can observe that there are no cascades on the random network for low degree, this is caused by the fact

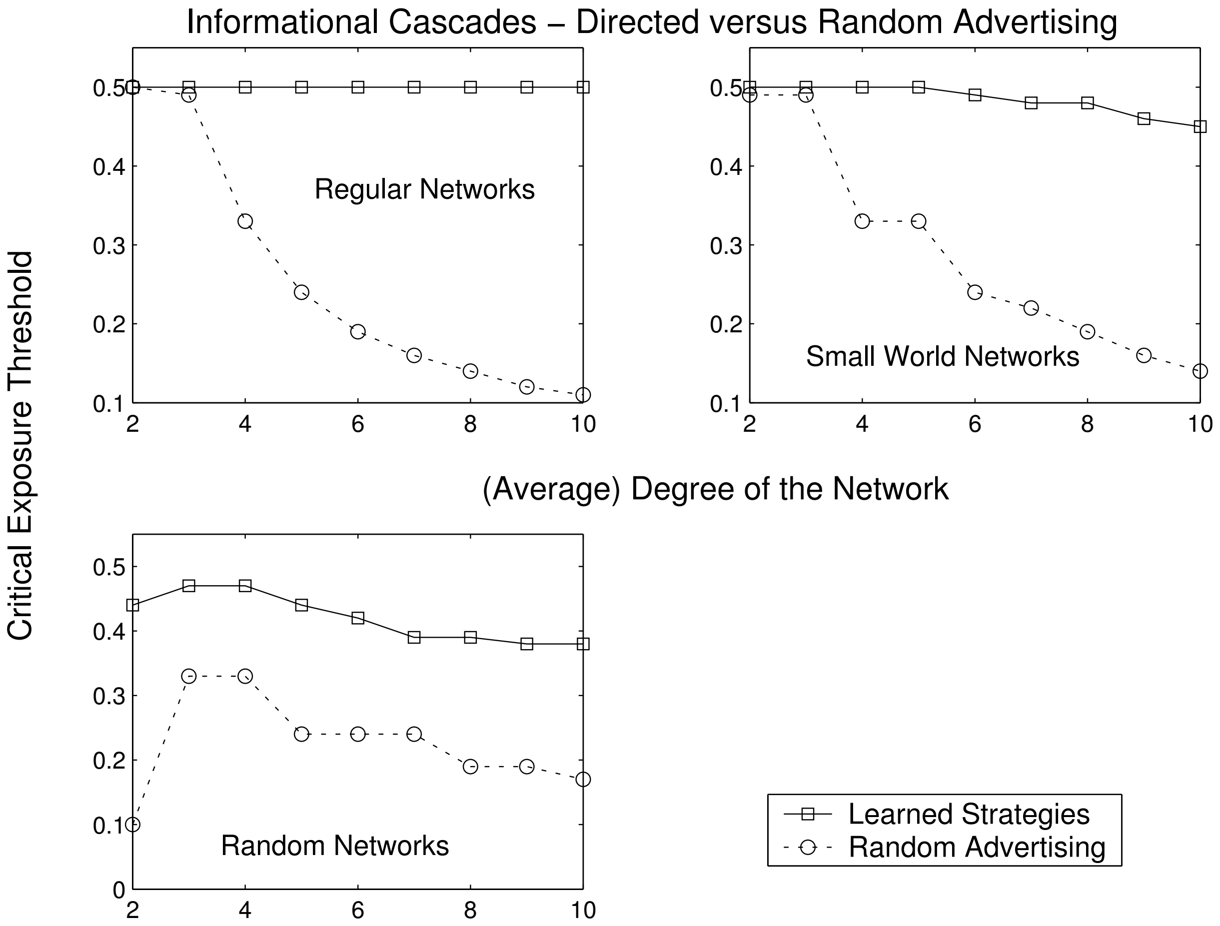

Figure 6.4: Critical diffusion thresholds for the different types of networks. Points signify the maximum threshold for which an informational cascade was achieved after 1000 timesteps starting from 10 nodes. (Averages over 20 runs). The learning time is 50 timesteps.

that the network may not be connected and the diffusion cannot reach all the components of the network. Notice that as the degree increases, it becomes more difficult for a cascade to occur. That is, cascades are only observed in networks where the agents have a low exposure threshold. In a network of degree 2, only one neighbor needs to purchase the product in order to start the cascade, so a threshold of 0.5 is immediately obtained. Starting from one initial consumer it becomes much more unlikely that a large fraction of neighbors has bought the good if the degree is high. Furthermore, we also observed cascades for some runs above the critical threshold. Figure 6.4

shows that in order for a cascade to occur it is necessary that the network is connected, that is that there are not too many components. Furthermore, we see that as long as the network is connected, a cascade spreads more easily over less regular networks. This is in accordance with the literature, for example, Watts (2002).

With directed advertising however, this effect disappears and cascades are achieved most often on regular networks. This can be explained by the fact that on a regular network, firms only have to take into account the position of a consumer in the network. In small world networks and random networks however, not only the position of the consumer is important, but also the number of links a consumer has as well as the type of links (cross-network or only to close neighbors). Summarizing, we can say that firms are able to learn effective directed advertising strategies in a network with homogeneous consumers. Using these strategies, cascades can be achieved in situations where random strategies only lead to limited diffusion. In the next section we consider networks of heterogeneous consumers.

6.5.2 Heterogeneous consumers

In this section we consider the diffusion of information over the static network when consumers are heterogeneous with respect to their exposure threshold. Figure 6.5 shows the difference between random and directed advertising when consumer exposure thresholds are drawn from a normal N(0.3,0.1) distribution. Each point in the graph represents the average diffusion after 50 runs of 1000 timesteps. In each run, a different consumer network was generated.

Note that the learned strategies outperform the random advertising strategies with respect to the size of the maximum diffusion. If we look at Figure 6.6 we also notice that (1) directed advertising strategies are able to achieve cascades when the random strategies are not, (2) the size of the diffusion is larger for directed advertising strategies (even if no cascade is achieved), and (3) the speed of the diffusion is larger for directed advertising strategies.

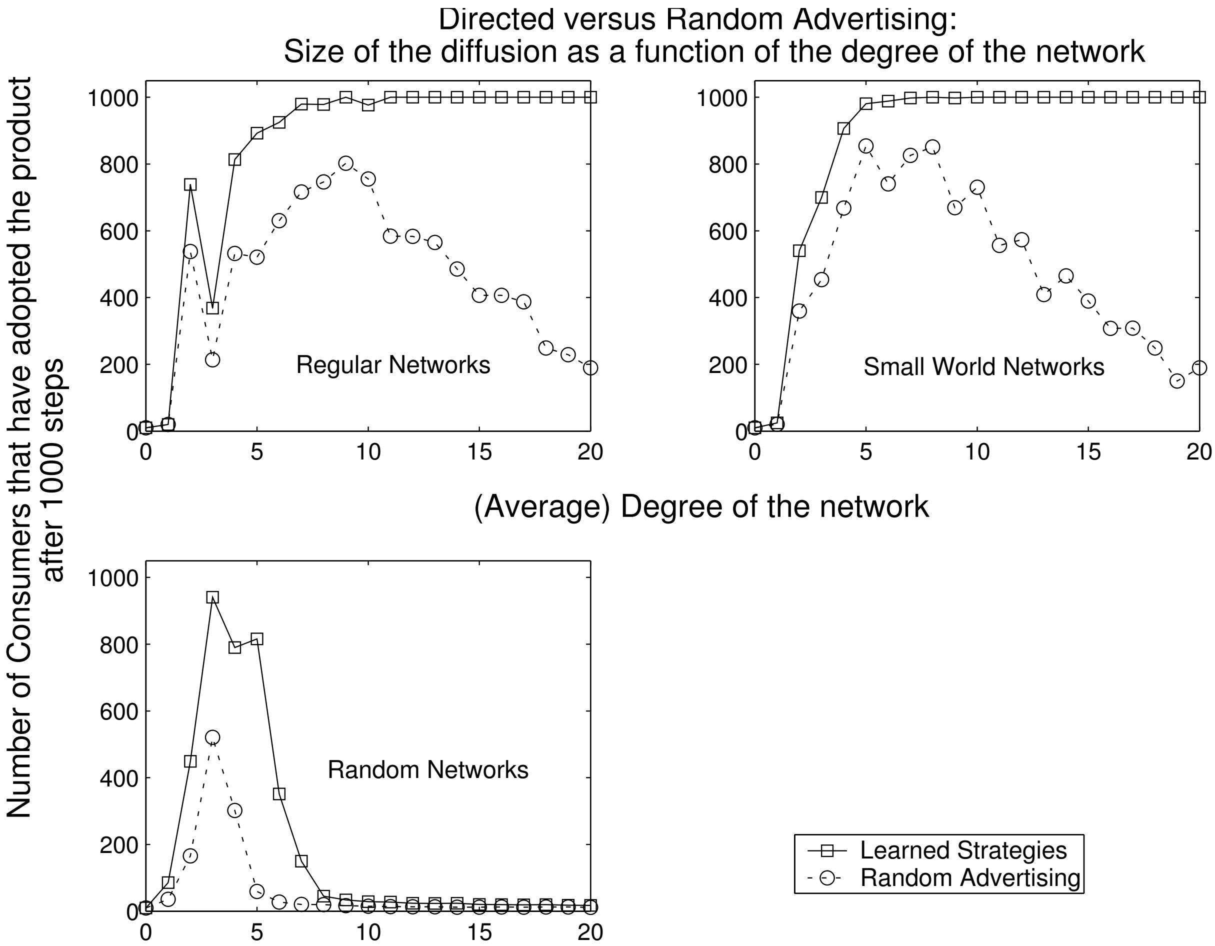

Figure 6.5: Directed and Random advertising when consumers are heterogeneous: Diffusion versus network degree for different types of networks. Each datapoint represents the average over 50 runs. The learning time is 50 timesteps.

6.5.3 Dynamics

To gain more insight in the nature of the directed advertising strategies, Figure 6.7 shows three evolved strategies for a smaller network. We see evolved strategies for three types of networks, with twelve heterogeneous consumers and an average degree of three. The experiments were conducted for a budget of 4. The numbers at the nodes represent the exposure thresholds of the consumers. Black nodes are targeted by the evolved strategies. All three strategies lead to a cascade within 10 timesteps. As we examine the evolved strategies more closely, we notice several interesting features. First,

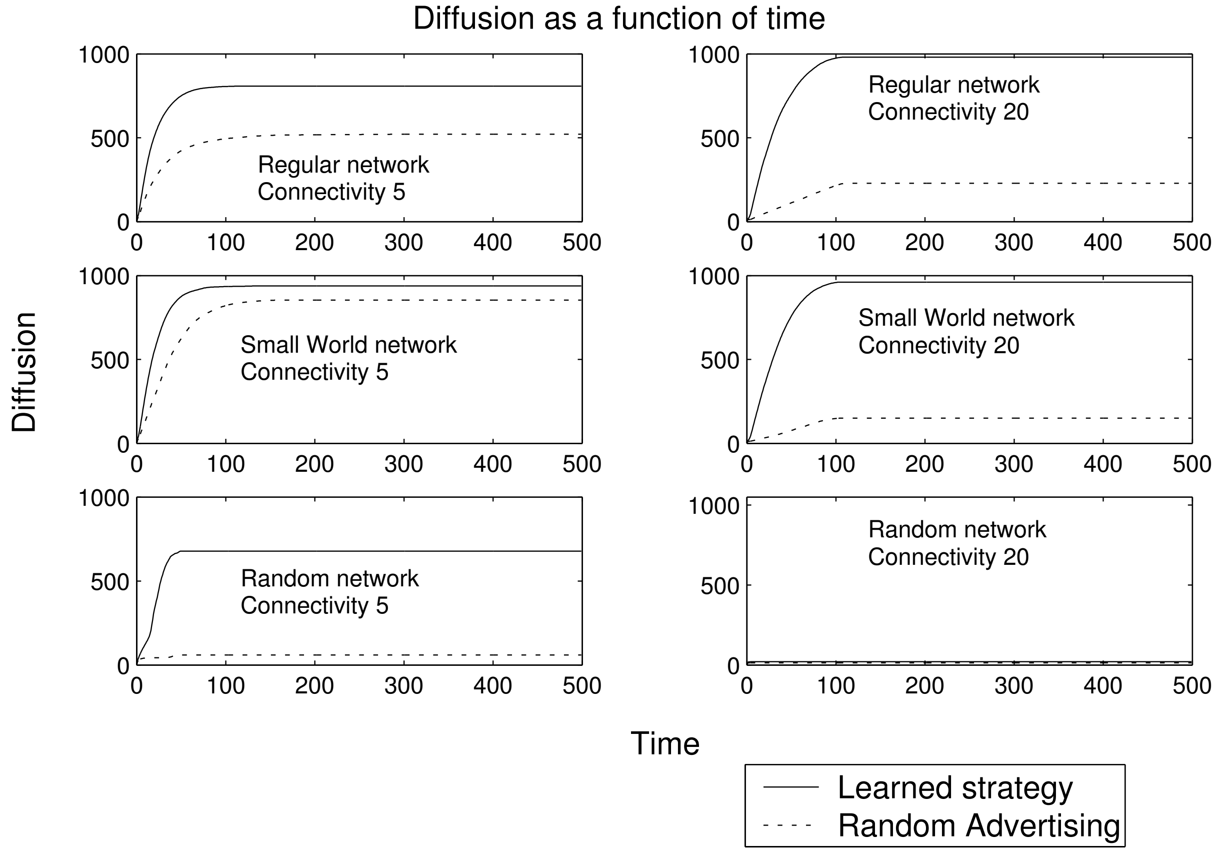

Figure 6.6: Directed versus Random advertising when consumers are heterogeneous: Diffusion as a function of time for different types of networks.

the firms target consumers with a high exposure threshold (that is, the most conservative consumers). Second they target consumers with a high number of neighbors, and third they target isolated consumers (these consumers can otherwise never be reached by the cascade). Furthermore, we notice that in the case of the Small World network, both nodes that have a "rewired" link are targeted, ensuring short pathlength for the diffusion. In the case of random networks, the disconnected nodes (or components) have to be targeted individually.

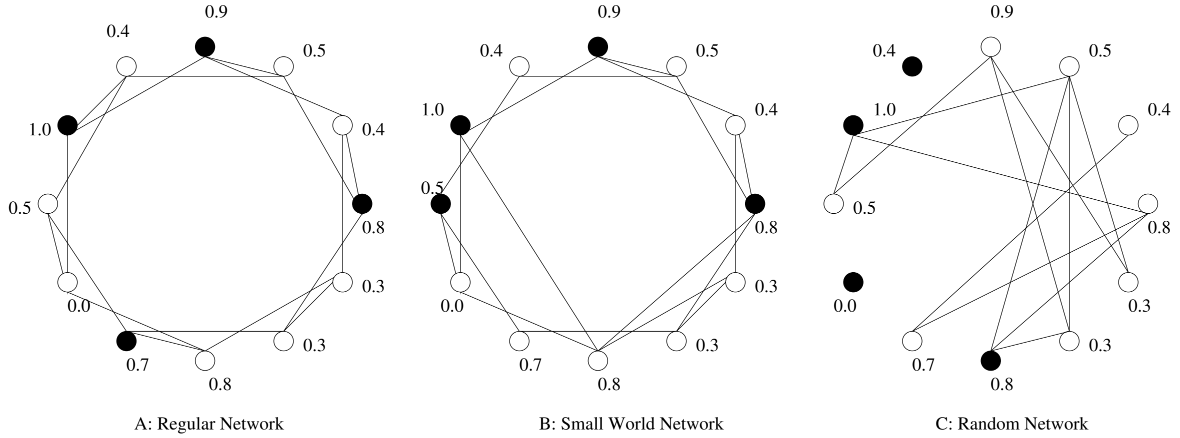

Figure 6.7: Learned directed advertising strategies. Black nodes are targeted by the learned strategy. Numbers at the nodes state the exposure thresholds of the individual consumers.

6.5.4 Introducing negative externalities

This section describes the network dynamics when negative as well as positive externalities are present in the model. We consider two scenarios. In the first scenario we assume that the attractiveness of the product depends on the number of other consumers that have adopted it. A consumer decides to adopt the product if this number exceeds its exposure threshold, but discontinues using the product if its over-exposure threshold is exceeded. However, once the number of other users decreases, the user may decide to use the product again. In the second scenario, a consumer never returns to the product once he has abandoned it. This may model a situation where the consumers abandon a fashion or fad and move on to the next hip thing.

Scenario 1: Figure 6.8 shows results for heterogeneous agents when negative externalities are also present. In these experiments, the exposure threshold et is drawn from uniform[0,0.5]. The exposure threshold and the over-exposure threshold et and oet are correlated, namely $oet = et + 0.5$. This reflects the fact that the innovators may also be the first consumers who abandon the product if the next innovation reaches the market or if they do no longer consider the item fashionable if too many consumers are using it. Again, we have used a network of 1000 consumers and look at the size of the diffusion after 100 periods (the size of the diffusion remained constant if we consider 1000 timesteps). First, we note that the learned strategy performs much better than the random strategy in regular networks. This can be explained by looking at the path of the diffusion. In networks with high clustering (such as regular networks and small world networks with a low rewiring constant) the diffusion progresses from neighbor to neighbor. But since in such networks most neighbors of one consumer are also neighbors of each other, the *over-exposure* threshold is reached sooner than in networks with low clustering.

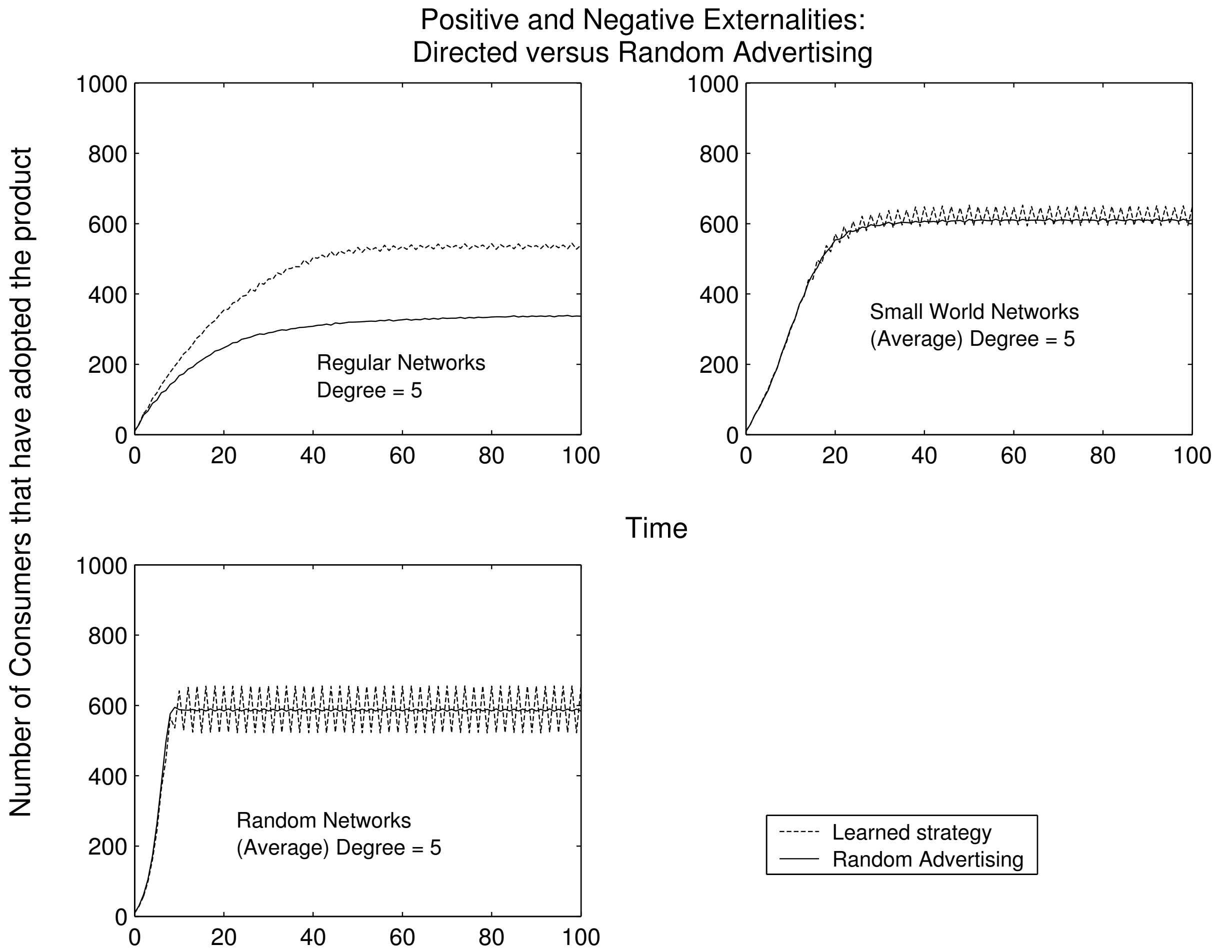

Figure 6.8: Diffusion of the product when negative as well as positive externalities are present. Averages over 20 runs.

This explains why the average size of the diffusion is lower in regular networks than for less clustered networks. Table 6.2 gives the average size of the diffusion for the different types of networks. Figure 6.9 shows a typical

Network Topology	**Average size of the diffusion** *Learned/Random Strategy*
Regular Networks	535/337
Small World Networks	623/610
Random Networks	589/586

Table 6.2: Average number of consumers that have adopted the innovation, when negative externalities are present, averaged over 20 runs and taken over timestep 100–1000.

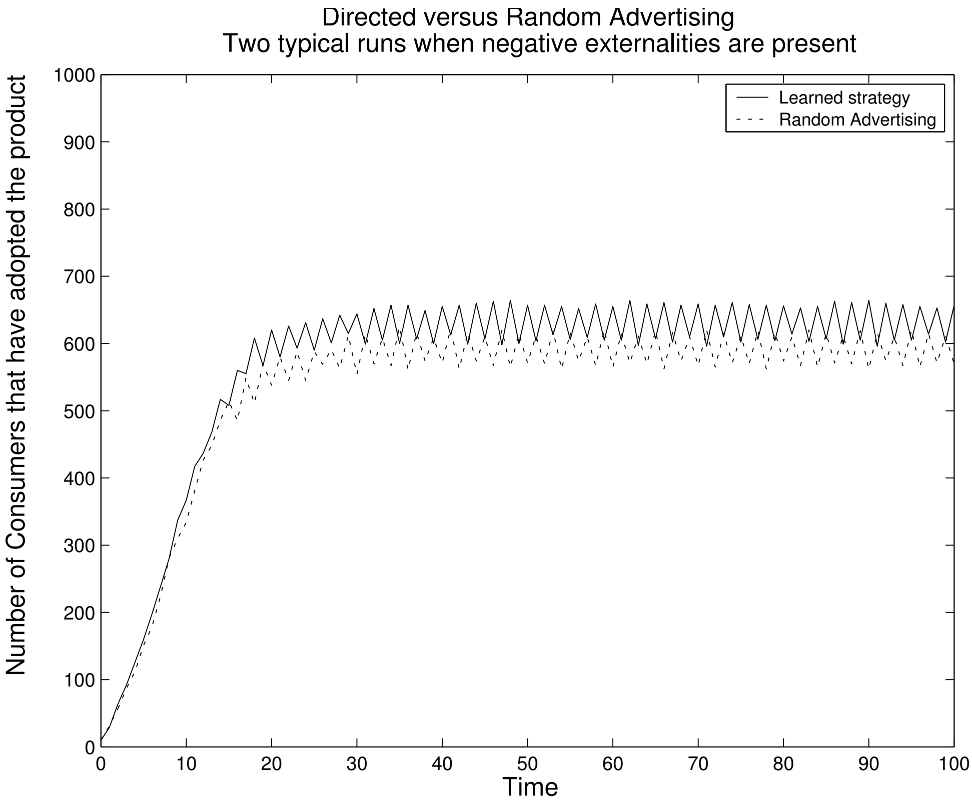

Figure 6.9: Diffusion of the product in a small world network when negative as well as positive externalities are present. Two typical runs.

run for a Small-World Network. Note that the average size of the diffusion is a little higher for the learned strategy. The oscillating behaviour is caused by a small group of consumers that continually switches between using and not using the product. If we look at the second scenario we see that this behaviour disappears.

Scenario 2: Under this scenario, the negative externality has a permanent effect and a consumer who decides to discontinue using the product will never resume using it. Figure 6.10 gives results for different types of networks under these conditions. First, we notice that the fluctuating behaviour disappears. Furthermore, we can observe that the total size of the diffusion is smaller than in the first scenario. This is caused by the fact that the diffusion dies off because of the negative externality, before all potential consumers have been reached. This effect is stronger for the more regular networks than for the random network. Moreover, the effect of learning is also much clearer for the regular and the small world network. This is caused by the fact that the diffusion goes very fast in random networks.

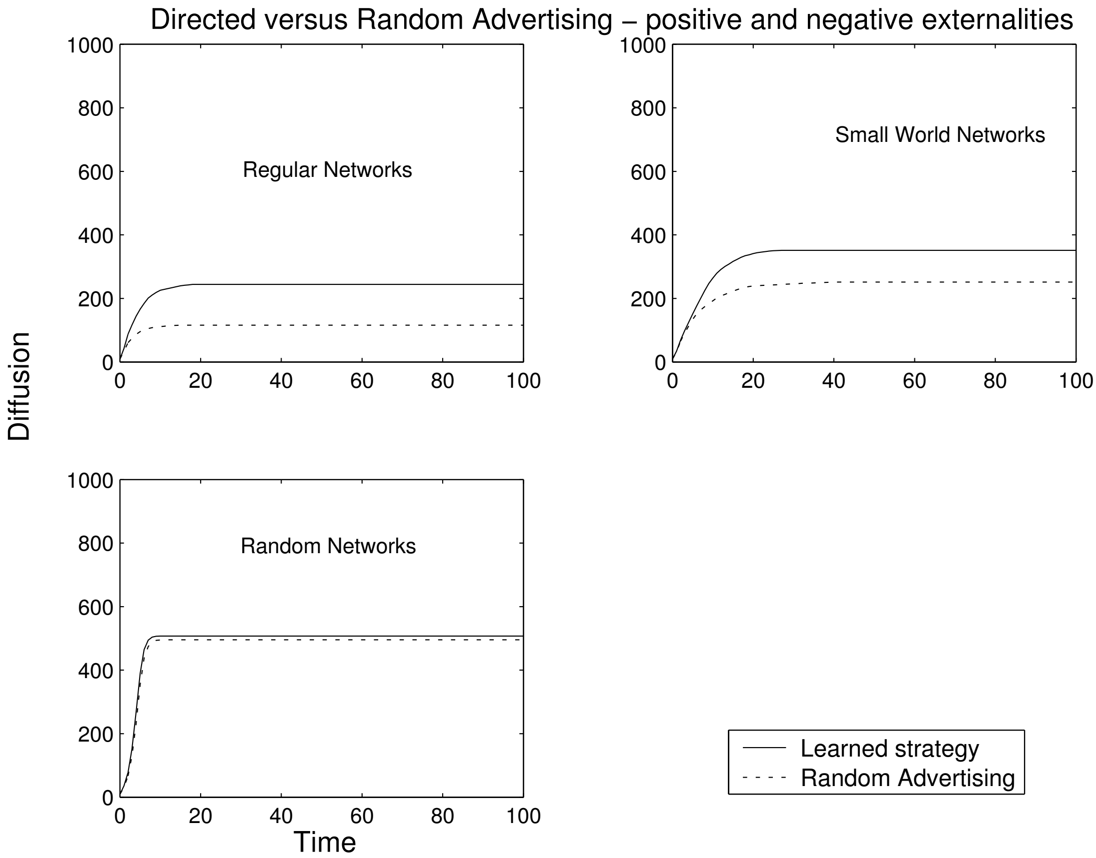

Figure 6.10: Diffusion patterns when negative externalities are present: scenario 2.

6.6 Conclusions

This chapter investigated the spread of information on a social network. The network consists of agents that are exposed to the introduction of a new product. Consumers decide whether or not to buy the product based on their own preferences and the decisions of their neighbors in the social network. We have used and extended concepts from the literature on epidemics and herd behavior to study this problem. The central question of this chapter is whether firms can learn about the network structure and consumer characteristics when only limited information is available, and use this information to evolve a successful directed advertising strategy. In order to do so, we have extended existing models to allow for heterogeneous agents and positive as well as negative externalities. The firm can learn a directed advertising strategy that takes into account both the topology of the social consumer network and the characteristics of the consumer. Such directed advertising strategies outperform random advertising.

Chapter 7

Concluding remarks

Evolutionary agent-based economics differs from traditional economics in two important aspects. Agents are boundedly rational instead of perfectly rational and interaction is not global and not anonymous. These two factors form the basis of this thesis and in the previous chapters we have studied the effects of incorporating bounded rationality as well as different forms of interaction on economic model outcomes.

We have shown that different types of interaction between agents can have a great impact on the outcomes of the economic game or model under investigation. We have modeled several local interaction mechanisms and structures that are observed in the real world. Using these forms of interaction, outcomes that are different from economic theory but that do occur in the real world were observed. Furthermore, our simulations have shown that the level of rationality of the agents also has a great influence on the outcomes of the model under investigation. Boundedly rational agents often achieve outcomes that differ from the outcomes predicted by economic theory but that are observed in the real world (i.e., collusion or cooperation). In this way agent-based simulations can help us to close the gap between theory and the real world.

Evolutionary agent-based economics allows us to study the emerging outcomes from micro-level behavior in large systems of interacting agents. Furthermore, this technique makes it possible to relax some of the restrictive assumptions made in economic theory. Agent-based simulations are thus a useful and necessary addition to game theoretic models in the study of systems of interacting intelligent agents. Evolutionary game theoretic models study such systems through mathematical analysis and have given us many useful insights. However, in order for these models to remain analytically tractable, agent behavior is oversimplified and assumptions are often far from realistic. Advances in computing, in both hardware and algorithms, enable us to study economic models and games through simulations. Such simulations are particularly suitable to study how agents learn from past experience and from each other, and how such learning processes lead to change and dynamic emergent behavior at the aggregate level. Furthermore, it allows us to further investigate systems where path-dependency plays an important role, that is systems where small differences in the initial conditions or the interaction patterns between agents can lead to significant changes in outcomes. Using evolutionary algorithms to model agent learning behavior makes it possible to study heterogeneous agents with a wide variety of (innovative) learned strategies. We thus use evolutionary algorithms to evolve good agent strategies based on the Darwinian principle of 'survival of the fittest'. We have shown that the evolutionary agent-based economics approach allows us to both study extensions of existing models and gain insight in the dynamics of newly arising market structures.

Finally, we would like to emphasize that it in order to draw valid conclusions from evolutionary agent-based economics models, the modeler should take several modeling and methodological issues into account. First, simulation results must always be compared to a benchmark model in order to be able to interpret these results. In this thesis the benchmark model used is the existing economic theory. Simulation models can be validated by comparing outcomes in standard cases with theoretical outcomes, and known outcomes form a benchmark for the outcomes of new simulations with less restrictive assumptions. The second important issue is how the parameters of the computational intelligence technique relate to the learning behavior of the agents. Modelers have to distinguish technical parameters from learning parameters. Furthermore, simulation results should be robust to small changes in the technical parameter settings. If these guidelines are followed, agent-based evolutionary economics makes it possible to study a wide variety of previously unstudied models and phenomena.

Bibliography

Albin, P.S. (1998). *Barriers and bounds to rationality: essays on economic complexity and dynamics in interactive systems*. Princeton University Press, Princeton (NJ).

Alchian, A.A. (1950). Uncertainty, evolution, and economic theory. *Journal of Political Economy* **58**, 211–221.

Alkemade, F. and C. Castaldi (2004). Diffusion of information on a social network. Technical report. CWI.

Alkemade, F. and J.A. La Poutré (2002). Boundedly rational agents achieving collusive outcomes in the cournot game. In: *Proceedings of the Sixth Joint Conference on Information Sciences* (P.P. Wang, Ed.). Association for Intelligent Machinery. pp. 1143–1146.

Alkemade, F. and J.A. La Poutré (2003). Heterogeneous, boundedly rational agents in the cournot duopoly. In: *Heterogenous Agents, Interactions and Economic Performance* (R. Cowan and N. Jonard, Eds.). Vol. 521 of *Lecture Notes in Economics and Mathematical Systems*. Springer. pp. 3–17.

Alkemade, F., D.D.B. van Bragt and J.A. La Poutré (2000). Stabilization of tag-mediated interaction by sexual reproduction in an evolutionary agent system. In: *Proceedings of the Fifth Joint Conference on Information Sciences* (P.P. Wang, Ed.). Vol. 2. Association for Intelligent Machinery. pp. 945–949.

Alkemade, F., D.D.B. van Bragt and J.A. La Poutré (In press). Stabilization of tag-mediated interaction by sexual reproduction in an evolutionary agent system. *Journal of Information Sciences*.

Alkemade, F., H.M. Amman and J.A. La Poutré (2004). The separation of economic versus ea parameters in ea-learning. Technical report. CWI.

Alkemade, F., J.A. La Poutré and H.M. Amman (2003*a*). An agent-based evolutionary trade network simulation. In: *Innovations in Financial and Economic Networks* (A. Nagurney, Ed.). Edward Elgar, Northampton (MA). pp. 237–252.

Alkemade, F., J.A. La Poutré and H.M. Amman (2003*b*). Intermediaries in an electronic trade network. In: *Proceedings of the BNAIC 2003* (T. Heskes, P. Lucas, L. Vuurpijl and W. Wiegerinck, Eds.). pp. 381–382.

Alkemade, F., J.A. La Poutré and H.M. Amman (2003*c*). Intermediaries in an electronic trade network. In: *Proceedings of the 4th ACM Conference on Electronic Commerce (EC-03)*. ACM Press. pp. 200–201.

Arifovic, J. (1994). Genetic algorithm learning and the cobweb model.. *Journal of Economic Dynamics and Control* **18**, 3–28.

Arifovic, J. (1996). The behavior of exchange rate in in the genetic algorithm and experimental economies. *Journal of Political Economy* **104**(3), 510–541.

Arifovic, J. (1998). Stability of equilibria under genetic algorithm adaptation. *Macroeconomic Dynamics* **2**(1), 1–21.

Arifovic, J. (forthcoming). Statistical properties of genetic algorithm learning in macroeconomic models. *Journal of Economic Dynamics and Control.*

Arifovic, J. and C. Eaton (1995). Coordination via genetic learning. *Computational Economics* **8**, 181–203.

Arthur, W.B. (1993). On designing economic agents that behave like human agents. *Journal of Evolutionary Economics* **3**, 1–22.

Arthur, W.B. (1994). *Increasing returns and path dependence in the economy.* University of Michigan Press, Ann Arbor (MI).

Axelrod, R. (1984). *The Evolution of Co-operation.* Basic Books, New York.

Axelrod, R. (1987). The evolution of strategies in the iterated prisoner's dilemma. In: *Genetic Algorithms and Simulated Annealing.* Pitman, London. pp. 32–41.

Axtell, R., J.M. Epstein and H.P. Young (2000). The emergence of classes in a multi-agent bargaining model. *Center on Social and Economic Dynamics Working Paper No. 9.*

Bäck, T., D. Fogel and (Eds.) Z. Michalewicz (1997*a*). *Handbook of evolutionary computation.* Oxford University Press, Oxford.

Bäck, T., U. Hammel and H.P. Schwefel (1997*b*). Evolutionary computation: Comments on the history and current state. *IEEE Transactions on Evolutionary Computation* **1**(1), 3–17.

Bala, V. and S. Goyal (2000). A non-cooperative model of network formation. *Econometrica* **68**(5), 1181–1231.

Banerjee, A.V. (1992). A simple model of herd behaviour. *Quarterly Journal of Economics* **107**(3), 797–818.

Barabási, A.L. and R. Albert (1999). Emergence of scaling in random networks. *Science* **286**, 509–512.

Beaufills, B., J. Delahaye and P. Mathieu (1996). Our meeting with Gradual: A good strategy for the iterated prisoner's dilemma. In: *Artificial Life, Proc. 5th International Workshop on the Synthesis and Simulation of Living Systems* (C. Langton, Ed.). Vol. 5. MIT Press. pp. 202–209.

Bernheim, D. (1994). A theory of conformity. *Journal of Political Economy* **105**, 841–877.

Biethahn, J. and V. Nissen (Eds.) (1995). *Evolutionary algorithms in management applications*. Springer Verlag, Berlin.

Bikhchandani, S., Hirshleifer and I. Welch (1992). A theory of fads, fashion, custom, and cultural change as informational cascades. *Journal of Political Economy* **100**(5), 992–1026.

Blume, L.E. and D. Easley (1993). Economic natural selection. *Economic letters* **42**, 281–289.

Boyd, R. and P.J. Richerson (1991). Culture and cooperation. In: *Cooperation and prosocial behavior* (R.A. Hinde and J. Groebel, Eds.). Cambridge University press, Cambridge (MA). pp. 27–48.

Bradshaw, J.M. (Ed.) (1997). *Software Agents*. AAAI/MIT Press, Cambridge (MA).

Bragt van, D.D.B., C.H.M. van Kemenade and J.A. La Poutré (2001). The influence of evolutionary selection schemes on the iterated prisoner's dilemma. *Computational Economics* **17**(2), 253–263.

Brock, W.A. and S.N. Durlauf (1997). Discrete choice with social interaction. *Review of Economic Studies* **68**, 235–260.

Bullard, J. (1999). Using genetic algorithms to model the evolution of heterogeneous beliefs. *Computational Economics* **13**(1), 41–60.

Callaway, D.S., M.E.J. Newman, S.H. Strogatz and D.J. Watts (2000). Network robustness and fragility: percolation on random graphs. *Phys. Rev. Lett.* **85**, 5468–5471.

Chattoe, E. (1998). Just how (un)realistic are evolutionary algorithms as representations of social processes?. *Journal of Artificial Societies and Social Simulation.* http://www.soc.surrey.ac.uk/JASSS/1/3/2.html.

Chen, S.H. and C.H. Yeh (1997). Towards a computable approach to the efficient market hypothesis: An application of genetic programming. *Journal of Economic Dynamics and Control* **21**(6), 1043–1064.

Chen, S.H. and C.H. Yeh (2001). Evolving traders and the business school with genetic programming: A new architecture of the agent-based artificial stock market. *Journal of Economic Dynamics and Control* **25**(3-4), 363–393.

Chiarella, C., M. Gallegati, R. Leombruni and A. Palestrini (2003). Asset price dynamics among heterogeneous interacting agents. *Computational Economics* **22**(2), 213–223.

Choi, S.Y., D.O. Stahl and A.B. Whinston (1997). *The economics of electronic commerce*. Macmillan technical publishing, Indianapolis (IN).

Cournot, A. (1897). *Researches into the Mathematical Principles of the Theory of Wealth*. Macmillan, New York (NY).

Cowan, R. and N. Jonard (2004). Network structure and the diffusion of knowledge. *Journal of Economic Dynamics and Control* **28**, 1557–1575.

Cowan, R., W. Cowan and P. Swann (1997). A model of demand with interactions among consumers. *International Journal of Industrial Organization* **15**, 711–732.

David, P.A. (1985). Clio and the economics of qwerty. *American Economic Review* **75**, 332–337.

Dawes, R.M. (1980). Social dilemmas. *Annual Review of Psychology* **31**, 169–193.

Dawid, H. (1999). *Adaptive Learning by Genetic Algorithms: Analytical Results and Applications to Economic Models*. 2nd, revised and enlarged ed.. Springer Verlag, Berlin.

DiMicco, J.M., A. Greenwald and P. Maes (2001). Dynamic pricing strategies under a finite time horizon. In: *Proceedings of the 3rd ACM Conference on Electronic Commerce*. ACM Press, New York (NY). pp. 95–104.

Edmonds, B. (2001). Towards a descriptive model of agent strategy search. *Computational Economics* **18**, 111–133.

Eisenhardt, K.M. (1989). Agency theory: an assessment and review. *Academy of Management Review* **14**(1), 57–74.

Ellison, G. (1993). Learning, social interaction and coordination. *Econometrica* **61**, 1047–1071.

Faratin, P., C. Sierra and N. R. Jennings (2003). Using similarity criteria to make issue trade-offs. *Journal of Artificial Intelligence* **142**(2), 205–237.

Farrell, J. and G. Saloner (1986). Installed base and compatibility: Innovation, product preannouncements, and predation. *The American Economic Review* **76**(5), 940–955.

Fatima, S., M. Wooldridge and N.R. Jennings (2003). Comparing equilibria for game theoretic and evolutionary bargaining models. In: *Proceedings of the 5th. International Workshop in Agent-Mediated E-Commerce*. pp. 70–73.

Fudenberg, D. and D. Levine (1998*a*). *The Theory of Learning in Games*. MIT Press, Cambridge (MA).

Fudenberg, D. and D.K. Levine (1998*b*). Learning in games. *European economic review* **42**, 631–639.

Gilbert, D. and P. Janca (1997). IBM intelligent agents.

Glance, N.S. and B.A. Huberman (1994). The dynamics of social dilemmas. *Scientific American* pp. 76–81.

Greenwald, A.M. and J.O. Kephart (1999). Shopbots and pricebots. In: *Proceedings of the 16th Joint Conference on Artificial Intelligence (IJCAI-99)*. pp. 506–511.

Grefenstette, J.J. (1986). Optimization of control parameters for genetic algorithms. *IEEE Trans. on Systems, Man, and Cybernetics* **16**(1), 122–128.

Hardin, G. (1968). The tragedy of the commons. *Science* **162**, 1243–1248.

Hippel von, E. (1988). *The sources of innovation*. Oxford University Press, New York (NY).

Hodgson, G.M. (2002). Darwinism in economics: from analogy to ontology. *Journal of Evolutionary Economics* **12**, 259–281.

Holland, J.H. (1975). *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor (MI).

Holland, J.H. (1993). The effect of labels (tags) on social interactions. *Santa Fe institute Working Paper 93-10-064*.

Holland, J.H. (1995). *Hidden order: How adaptation builds complexity*. Addison-Wesley, Reading (MA).

Hu, J. and M.P. Wellman (2003). Nash Q-learning for general-sum stochastic games. *Journal of Machine Learning Research* **4**, 1039–1069.

Jong, K.A. De (1975). An analysis of the behavior of a class of genetic adaptive systems. PhD thesis. University of Michigan.

Kalai, E. and E. Lehrer (1993). Rational learning leads to nash equilibrium. *Econometrica* **61**(5), 1019–1045.

Kandori, M., G.J. Mailath and R. Rob (1993). Learning, mutation, and long run equilibria in games. *Econometrica* **61**(1), 29–56.

Katz, M.L. and C. Shapiro (1985). Network externalities, competition, and compatibility. *American Economic Review* **75**, 424–440.

Klos, T.B. (1999). Decentralized interaction and co-adaptation in the repeated prisoner's dilemma. *Computational and Mathematical Organization Theory* **5**(2), 147–165.

Knudsen, T. (2002). Economic selection theory. *Journal of Evolutionary Economics* **12**, 443–470.

Kranton, R.E. and D.F. Minehart (2001). A theory of buyer-seller networks. *American Economic Review* **91**(3), 485–508.

Maes, P. (1994). Agents that reduce work and information overload. *Communications of the ACM* **37**(7), 30–40.

Mailath, G.J. (1992). Introduction: Symposium on evolutionary game theory.. *Journal of economic theory* **57**, 259–277.

Marks, R.E. (1992). Breeding hybrid strategies: Optimal behaviour for oligopolists. *Journal of Evolutionary Economics* **2**, 17–38.

Marks, R.E. (1998). Evolved behaviour and perception in oligopolies. *Journal of Economic Dynamics and Control* **22**, 1209–1233.

Mas-Colell, A., M.D. Whinston and J.R. Green (1995). *Microeconomic theory.* Oxford University Press, Oxford.

Miller, J.H. (1986). A genetic model of adaptive economic behavior.

Mitchell, M. (1996). *An introduction to genetic algorithms.* MIT Press, Cambridge (MA).

Moore, C. and M.E.J. Newman (2000). Epidemics and percolation in a small-world networks. *Phys. Rev. E* **61**, 5678–5682.

NachBar, J.H. (1997). Prediction, optimization, and learning in repeated games. *Econometrica* **65**(2), 275–309.

Nelson, R.R. and S.G. Winter (1982). *An evolutionary theory of economic change.* Harvard Univeristy Press, Cambridge (MA).

Newell, A. and H.A. Simon (1972). *Human Problem Solving.* Prentice Hall, Englewood Cliffs (NJ).

Newman, M.E.J. (2000). The spread of epidemic disease on networks. *Phys. Rev. E* **66**, ?

Newman, M.E.J. and D.J. Watts (1999). Scaling and percolation in the small-world network model. *Phys. Rev. E* **60**, 7332–7342.

Nowak, M.A. and K. Sigmund (1998). Evolution of indirect reciprocity by image scoring. *Nature* **393**(11), 573–577.

Price, T.C. (1997). Using co-evolutionary programming to simulate strategic behaviour in markets. *Journal of Evolutionary Economics* **7**, 219–254.

Riechmann, T. (2001*a*). Genetic algorithm learning and evolutionary games. *Journal of Economic Dynamics and Control* **25**(6-7), 1019–1037.

Riechmann, T. (2001*b*). *Learning in Economics*. Physica-Verlag, Heidelberg.

Riolo, R.L. (1997*a*). The effects and evolution of tag-mediated selection of partners in populations playing the iterated prisoner's dilemma. In: *Proceedings of the 7th International Conference on Genetic Algorithms*. pp. 378–385.

Riolo, R.L. (1997*b*). The effects of tag-mediated selection of partners in evolving populations playing the iterated prisoner's dilemma. *Santa Fe institute Working Paper 97-02-016*.

Sargent, T.J. (1993). *Bounded Rationality in Macroeconomics*. Oxford University Press, New York (NY).

Schelling, T.C. (1978). *Micromotives and macrobehavior*. W.W. Norton and Company Inc., New York (NY).

Scott, J. (2000). *Social Network Analysis*. 2nd ed.. Sage Publications, London.

Shapiro, C. and H.R. Varian (1999). *Information rules: a strategic guide to the network economy*. Harvard Business School Press, Boston (MA).

Simon, H.A. (1984). *Models of Bounded Rationality*. The MIT Press, Cambridge (MA).

Spulber, D.F. (1999). *Market microstructure: intermediaries and the theory of the firm*. Cambridge University Press, Cambridge (MA).

Stanley, E.A., D. Ashlock and L. Tesfatsion (1994). Iterated prisoner's dilemma with choice and refusal of partners. In: *Artificial Life III: SFI Studies in the Sciences of Complexity* (C.G. Langton, Ed.). Vol. XVIII. Addision-Wesley, Reading (MA). pp. 131–175.

Strogatz, S.H. (2001). Exploring complex networks. *Nature* **410**, 268–276.

Tesfatsion, L. (2001). Introduction to the special issue on agent-based computational economics. *Journal of Economic Dynamics and Control* **25**(3/4), 281–293.

Tsang, E.P.K. and J. Li (2000). Combining ordinal financial predictions with genetic programming. In: *Proceedings of the second international conference on intelligent data engineering and automated learning (IDEAL2000)* (K.S. leung, L.W. Chan and H. Meng, Eds.). Vol. 1983 of *Lecture Notes in Computer Science*. Springer. pp. 532–537.

Turban, E. and J.E. Aronson (1998). *Decision support systems and intelligent systems*. Prentice-Hall, Upper Saddle River (NJ).

Tuson, P. and P. Ross (1998). Adapting operator settings in genetic algorithms. *Evolutionary computation* **6**(2), 161–184.

Valente, T.W. and R.L. Davis (1999). Accelerating the diffusion of innovations using opinion leaders. *Annals of the American Academy of Political and Social Sciences* **566**, 55–67.

Varian, H.R. (1992). *Microeconomic analysis*. W.W. Norton, New York (NY).

Vega-Redondo, F. (1997). The evolution of walrasian behavior. *Econometrica* **65**, 375–384.

Vriend, N.J. (1995). Self-organization of markets: An example of a computational approach. *Computational Economics* **8**, 205–231.

Vriend, N.J. (2000). An illustration of the essential difference between individual learning and social learning and its consequences for computational analyses. *Journal of Economic Dynamics and Control* **24**, 1–19.

Walsh, W.E. and M.P. Wellman (2003). Decentralized supply chain formation: A market protocol and competitive equilibrium analysis. *Journal of Artificial Intelligence Research* **19**, 513–567.

Wasserman, S. and K. Faust (1994). *Social Network Analysis*. Cambridge University Press, Cambridge (MA).

Watts, D.J. (2002). A simple model of global cascades on random networks. *Proceedings of the National Academy of Sciences (PNAS)* **99**(9), 5766–5771.

Watts, D.J. and S.H. Strogatz (1998). Collective dynamics of 'small-world' networks. *Nature* **393**, 440–442.

Weibull, J. (2002). *Evolutionary Game Theory*. MIT Press, Cambridge (MA).

Weiss, G. (Ed.) (1999). *Multiagent systems: a modern approach to distributed artificial intelligence*. MIT Press, Cambridge (MA).

Wooldridge, M. and N.R. Jennings (1995). Intelligent agents: Theory and practice. *The knowledge engineering review* **10**(2), 115–152.

Yao, X. and P.J. Darwen (1994). An experimental study of N-person iterated prisoner's dilemma games. *Informatica* **18**, 435–450.

Young, H.P. (1993). The evolution of conventions. *Econometrica* **61**(1), 57–84.

Young, H.P. (1998). *Individual strategy and social structure: an evolutionary theory.* Princeton University Press, Princeton, (NJ).

Young, H.P. (2003). The diffusion of innovations in social networks. In: *The Economy as an Evolving Complex System III* (L.E. Blume and S.N. Durlauf, Eds.). Oxford University Press, Oxford.

Summary

Evolutionary agent-based economics differs from traditional economics in two important aspects. First, agents are boundedly rational instead of perfectly rational and second, the interaction between agents is not global and not anonymous. These two factors form the basis of this thesis and we aim to include these aspects in our models. First by investigating the behavior of boundedly rational agents and the effects of interacting structure in existing economic models, and then by addressing new models concerning internet markets and information goods. Chapter 1 contains the introduction of the thesis and presents the most important concepts and methods used in the remainder of the thesis.

Chapter 2 is concerned with the proper design and implementation of evolutionary economic simulations. Agent-based computational economics (ACE) combines elements from economics and computer science. In this chapter, we focus on the relation between the evolutionary techniques that are used and the economic problem that is modeled. Current economic simulations often derive parameter settings for the genetic algorithm directly from the values of the economic model parameters. In this chapter we show that this practice may hinder the performance of the GA and thereby hinder agent learning. More specifically, we show that economic model parameters and evolutionary algorithm parameters should be treated separately by comparing two widely used approaches to population learning with respect to their convergence properties and robustness.

Chapter 3 is concerned with the Cournot model. In this chapter we present an individual learning model which allows us to compare different types of agents with different levels of rationality that are interacting in the market. A Cournot duopoly market modeled as a co-evolving system of autonomous interacting agents is investigated. We present results for

different types of boundedly rational agents. Agent types differ in both the complexity of their strategies and the information they have available to make their decision. Some types of agents use very simple strategies to make a production decision, while other types use a quite sophisticated decision rule. All agents types are tested in a round robin tournament. We consider the evolutionary stability of the evolving populations, especially with respect to the different equilibria of the Cournot game. Furthermore, we investigate the performance of the different agent types under changing market conditions.

In Chapter 4 we return to a population learning model but the focus of this chapter is on the structure of the interaction and the influence of the recombination operator on the outcomes of the simulation. The evolution of cooperation in a system of agents playing the iterated prisoner's dilemma (IPD) is investigated. We present results for the standard two-person IPD as well as the more general N-person IPD (NIPD) game. In our computational model, agents have visible tags and choose whether to interact or not based upon these tags. We consider the evolutionary stability of the evolving populations. We extend previous work by introducing sexual reproduction (recombination) of strategies and by analyzing its influence on the evolving populations. We observe the occasional formation of very stable cooperative societies, as opposed to previous results without sexual reproduction. These cooperative societies are able to resist invasions of "mimics" (defecting agents with the tag of a cooperating agent).

Chapter 5 builds on the techniques presented in the previous chapters to address the question whether intermediaries will still exist and be able to make a profit if consumers can make direct connections with producers, as is often the case in electronic commerce. We have performed agent-based simulations to study the performance of intermediaries under different market conditions. We modeled an electronic trade network where an information good is traded over the network. Each trade period, cost-minimizing consumers have to decide which links to form to sellers (i.e., producers and intermediaries), while the good can only be purchased if a link between the buyer and the seller exists. Links thus represent trading possibilities, and flows of relevant information (i.e., in our case, price quotes between potential buyers and sellers) and the consumers have to make a strategic decision about which (costly) links to form. We have used an evolutionary algorithm to model the search and learning behavior of the buyers. Our main finding is that if market dynamics are sufficiently complex, intermediaries that have better knowledge about the market than the average consumer are initially able to increase their market share and make a profit.

Chapter 6 investigates the spread of information on a social network. The network consists of agents that are exposed to the introduction of a new product. Consumers decide whether or not to buy the product based on their own preferences and the decisions of their neighbors in the social network. We have used and extended concepts from the literature on epidemics and herd behavior to study this problem. The central question of this chapter is whether firms can learn about the network structure and consumer characteristics when only limited information is available, and use this information to evolve a successful directed advertising strategy. In order to do so, we have extended existing models to allow for heterogeneous agents and positive as well as negative externalities. The firm can learn a directed advertising strategy that takes into account both the topology of the social consumer network and the characteristics of the consumer. Such directed advertising strategies outperform random advertising. Concluding remarks are given in Chapter 7.

Samenvatting

Evolutionary agent-based economics verschilt van de klassieke economische aanpak in twee belangrijke opzichten. Ten eerste zijn de agenten (of actoren) begrensd wat betreft hun rationaliteit, in plaats van volledig rationeel zoals vaak wordt aangenomen in de economische literatuur. Ten tweede is de interactie tussen de agenten niet globaal maar lokaal gespecificeerd, ook dit in tegenstelling tot wat we in de economische theorie vaak tegenkomen. Deze twee principes, begrensde rationaliteit en lokale interactie, vormen de basis van dit proefschrift. In de afzonderlijke hoofdstukken integreren we deze principes stap voor stap in economische modellen en bestuderen we de gevolgen hiervan. Allereerst bekijken we het gedrag van agenten met beperkte rationaliteit in bekende economische modellen. Dit geeft ons de mogelijkheid de uitkomsten goed te vergelijken met theoretische uitkomsten. Vervolgens richten we ons op modellen van internetmarkten en markten waar informatiegoederen verhandeld worden. Dit type markten is vrij recent en simulatietechnieken helpen om inzicht te krijgen in de dynamiek van deze markten. Hieronder geven we een meer gedetailleerde beschrijving van de verschillende hoofdstukken.

Hoofdstuk 1 bevat de introductie tot het proefschrift en behandelt de belangrijkste begrippen.

Hoofdstuk 2 behandelt verschillende aspecten van het modelleren en implementeren van evolutionaire economische simulaties. Agent-based computational economics is een interdisciplinair vakgebied op het gebied tussen de economie, psychologie en de informatica. In dit hoofdstuk houden we ons bezig met het onderscheid tussen de gebruikte evolutionaire techniek en het economische probleem dat onderzocht wordt. Veel onderzoekers nemen de keuzen voor de waarden van de parameters van het genetisch algoritme direct over uit het economische model. In dit hoofdstuk tonen we aan dat deze handelswijze de werking van het genetisch algoritme en daardoor het leren van de agenten negatief kan beïnvloeden. We doen dit door twee veelgebruikte ontwerpmethoden te vergelijken. We pleiten ervoor dat de modelbouwer de economische parameters en de parameters van het genetisch algoritme afzonderlijk behandelt. De richtlijnen voor het modelleren van evolutionaire simulaties die in dit hoofdstuk gegeven worden, functioneren als de basis voor de modellen in volgende hoofdstukken.

In hoofdstuk 3 bestuderen we het Cournot model. Hier gebruiken we een model voor individueel leren dat ons in staat stelt verschillende typen beperkt rationele agenten te vergelijken. We bekijken een Cournot duopolie markt waar de verschillende typen agenten elkaar tegenkomen. In dit type markt wordt de beste strategie mede bepaald door de beslissingen van de andere agent. Agenten verschillen zowel wat betreft de complexiteit van hun strategieën als wat betreft de beschikbare informatie op basis waarvan een beslissing moet worden gemaakt. Sommige typen agenten gebruiken hele eenvoudige strategieën om tot een produktiebeslissing te komen terwijl andere agenten meer complexe en verfijnde beslisregels gebruiken. We testen het succes van de verschillende typen door middel van marktsimulaties. De nadruk ligt hierbij op de stabiliteit van de evoluerende populaties rondom de mogelijke marktevenwichten. Ten tweede bestuderen we het gedrag van de agenten wanneer de marktcondities veranderen en dus een voortdurende aanpassing van de strategie vereist is.

In hoofdstuk 4 bekijken we de effecten van selectieve interactie. Bij selectieve interactie is de interactie tussen agenten niet globaal en anoniem zoals we in de theorie zien maar hebben agenten een voorkeur voor interactie met bepaalde andere agenten. We bestuderen dit door middel van het bekende herhaalde prisoner's dilemma spel, een veelgebruikt model voor de evolutie van samenwerking in situaties waarin het op de korte termijn juist lonend is om niet coöperatief te handelen. Naast een strategie voor het spel beschikken alle agenten ook over een zogenaamde tag, een label. We bestuderen of de agenten kunnen leren om bepaald gedrag met bepaalde tags te associëren om zo de coöperatie te bevorderen. De strategieën evolueren met behulp van een genetisch algoritme. We breiden eerder werk op dit gebied uit door niet alleen naar genetische algoritmen met selectie en mutatie, maar ook naar genetische algoritmen met reproductie (crossover) te kijken. De resultaten laten zien dat in tegenstelling tot bij asexuele genetische algoritmen, er stabiele coöperatieve (sub)populaties onstaan wanneer recombinatie wordt gebruikt. Deze stabiele populaties zijn in staat coöperatie in stand te houden en invasies van mimics (agenten met een coöperatieve tag maar niet-coöperatief gedrag) te weerstaan. Deze mimics zorgden er in eerdere experimenten voor dat de coöperatie steeds opnieuw teniet werd gedaan.

Hoofdstuk 5 richt zich op de vraag of intermediairs (tussenpersonen) kunnen blijven bestaan en winst kunnen maken in een markt waar de consumenten ook direct zaken kunnen doen met de producenten zoals vaak het geval is bij electronic commerce (handel over het internet). We bestuderen de prestaties van de tussenpersonen onder verschillende marktcondities door middel van simulaties met agenten. We bestuderen de handel in een informatiegoed over een electronisch (handels) netwerk. De kopers (consumenten, tussenpersonen) moeten beslissen met welke aanbieders (producenten dan wel tussenpersonen) ze een verbinding willen onderhouden; wat kosten met zich mee brengt. Er vindt alleen handel plaats als er een verbinding bestaat tussen een aanbieder en een koper. Verbindingen zijn dus handelsmogelijkheden, oftewel een uitwisseling van relevante informatie en kopers moeten een strategische keuze maken welke links te vormen. Ook hier gebruiken we een genetisch algoritme om het leergedrag van de agenten te simuleren. De belangrijkste conclusie uit dit hoofdstuk is dat tussenpersonen met expert-kennis over de markt in staat zijn om een substantieel marktaandeel te verwerven in snel-veranderende markten en winst te maken.

In hoofdstuk 6 onderzoeken we hoe informatie zich verspreidt over een sociaal netwerk. De agenten in het sociale netwerk krijgen te maken met de introductie van een nieuw product op de markt. Bij de beslissing om het nieuwe product al dan niet te kopen spelen niet alleen de preferenties van de agent zelf een rol, maar ook de beslissingen van zijn buren in het netwerk. Het gaat hier dus om goederen waarbij de waarde van het goed mede wordt bepaald door de mening van andere agenten, zogenaamede mode of hype goederen. We gebruiken concepten uit de literatuur over het verspreiden van epidemieën en over kuddegedrag bij consumenten als uitgangspunten voor ons model. De belangrijkste vraag hier is of een verkoper die slechts over geaggregeerde verkoopdata beschikt kan leren om gebruik te maken van de karaketeristieken van het sociale netwerk om tot een hogere marktpenetratie te komen. Deze verkopers leren dan een gerichte advertentiestrategie die slechts die consumenten benadert die een snelle verspreiding van het product garanderen. We bestuderen dit probleem door middel van simulaties met agenten en de producent leert marketingstrategieën door middel van een genetisch algoritme. Onze experimenten laten zien dat producenten inderdaad strategieën kunnen leren die op een efficiënte wijze gebruik maken van de structuur van het netwerk en de karakteristieken van de individuele consumenten. Zulke strategieën leiden tot een voordeel voor de producenten.

Curriculum Vitae

Floortje Alkemade werd geboren op 10 november 1975 te Hengelo, Overijssel. Van 1988 tot 1994 doorliep zij het Gymnasium op het Huygens Lyceum in Voorburg. Van 1994 tot 1999 studeerde zij Kunstmatige Intelligentie aan de Vrije Universiteit te Amsterdam. Haar afstudeerproject deed zij op het CWI, bij de groep van professor Han La Poutré. Van 1999 tot 2004 was zij als onderzoeker in opleiding (OIO) aangesteld in deze groep.

Titles in the IPA Dissertation Series

J.O. Blanco. *The State Operator in Process Algebra.* Faculty of Mathematics and Computing Science, TUE. 1996-01

A.M. Geerling. *Transformational Development of Data-Parallel Algorithms.* Faculty of Mathematics and Computer Science, KUN. 1996-02

P.M. Achten. *Interactive Functional Programs: Models, Methods, and Implementation.* Faculty of Mathematics and Computer Science, KUN. 1996-03

M.G.A. Verhoeven. *Parallel Local Search.* Faculty of Mathematics and Computing Science, TUE. 1996-04

M.H.G.K. Kesseler. *The Implementation of Functional Languages on Parallel Machines with Distrib. Memory.* Faculty of Mathematics and Computer Science, KUN. 1996-05

D. Alstein. *Distributed Algorithms for Hard Real-Time Systems.* Faculty of Mathematics and Computing Science, TUE. 1996-06

J.H. Hoepman. *Communication, Synchronization, and Fault-Tolerance.* Faculty of Mathematics and Computer Science, UvA. 1996-07

H. Doornbos. *Reductivity Arguments and Program Construction.* Faculty of Mathematics and Computing Science, TUE. 1996-08

D. Turi. *Functorial Operational Semantics and its Denotational Dual.* Faculty of Mathematics and Computer Science, VUA. 1996-09

A.M.G. Peeters. *Single-Rail Handshake Circuits.* Faculty of Mathematics and Computing Science, TUE. 1996-10

N.W.A. Arends. *A Systems Engineering Specification Formalism.* Faculty of Mechanical Engineering, TUE. 1996-11

P. Severi de Santiago. *Normalisation in Lambda Calculus and its Relation to Type Inference.* Faculty of Mathematics and Computing Science, TUE. 1996-12

D.R. Dams. *Abstract Interpretation and Partition Refinement for Model Checking.* Faculty of Mathematics and Computing Science, TUE. 1996-13

M.M. Bonsangue. *Topological Dualities in Semantics.* Faculty of Mathematics and Computer Science, VUA. 1996-14

B.L.E. de Fluiter. *Algorithms for Graphs of Small Treewidth.* Faculty of Mathematics and Computer Science, UU. 1997-01

W.T.M. Kars. *Process-algebraic Transformations in Context.* Faculty of Computer Science, UT. 1997-02

P.F. Hoogendijk. *A Generic Theory of Data Types.* Faculty of Mathematics and Computing Science, TUE. 1997-03

T.D.L. Laan. *The Evolution of Type Theory in Logic and Mathematics.* Faculty of Mathematics and Computing Science, TUE. 1997-04

C.J. Bloo. *Preservation of Termination for Explicit Substitution.* Faculty of Mathematics and Computing Science, TUE. 1997-05

J.J. Vereijken. *Discrete-Time Process Algebra.* Faculty of Mathematics and Computing Science, TUE. 1997-06

F.A.M. van den Beuken. *A Functional Approach to Syntax and Typing.* Faculty of Mathematics and Informatics, KUN. 1997-07

A.W. Heerink. *Ins and Outs in Refusal Testing.* Faculty of Computer Science, UT. 1998-01

G. Naumoski and W. Alberts. *A Discrete-Event Simulator for Systems Engineering.* Faculty of Mechanical Engineering, TUE. 1998-02

J. Verriet. *Scheduling with Communication for Multiprocessor Computation.* Faculty of Mathematics and Computer Science, UU. 1998-03

J.S.H. van Gageldonk. *An Asynchronous Low-Power 80C51 Microcontroller.* Faculty of Mathematics and Computing Science, TUE. 1998-04

A.A. Basten. *In Terms of Nets: System Design with Petri Nets and Process Algebra.* Faculty of Mathematics and Computing Science, TUE. 1998-05

E. Voermans. *Inductive Datatypes with Laws and Subtyping – A Relational Model.* Faculty of Mathematics and Computing Science, TUE. 1999-01

H. ter Doest. *Towards Probabilistic Unification-based Parsing.* Faculty of Computer Science, UT. 1999-02

J.P.L. Segers. *Algorithms for the Simulation of Surface Processes.* Faculty of Mathematics and Computing Science, TUE. 1999-03

C.H.M. van Kemenade. *Recombinative Evolutionary Search.* Faculty of Mathematics and Natural Sciences, UL. 1999-04

E.I. Barakova. *Learning Reliability: a Study on Indecisiveness in Sample Selection.* Faculty of Mathematics and Natural Sciences, RUG. 1999-05

M.P. Bodlaender. *Scheduler Optimization in Real-Time Distributed Databases.* Faculty of Mathematics and Computing Science, TUE. 1999-06

M.A. Reniers. *Message Sequence Chart: Syntax and Semantics.* Faculty of Mathematics and Computing Science, TUE. 1999-07

J.P. Warners. *Nonlinear approaches to satisfiability problems.* Faculty of Mathematics and Computing Science, TUE. 1999-08

J.M.T. Romijn. *Analysing Industrial Protocols with Formal Methods.* Faculty of Computer Science, UT. 1999-09

P.R. D'Argenio. *Algebras and Automata for Timed and Stochastic Systems.* Faculty of Computer Science, UT. 1999-10

G. Fábián. *A Language and Simulator for Hybrid Systems.* Faculty of Mechanical Engineering, TUE. 1999-11

J. Zwanenburg. *Object-Oriented Concepts and Proof Rules.* Faculty of Mathematics and Computing Science, TUE. 1999-12

R.S. Venema. *Aspects of an Integrated Neural Prediction System.* Faculty of Mathematics and Natural Sciences, RUG. 1999-13

J. Saraiva. *A Purely Functional Implementation of Attribute Grammars.* Faculty of Mathematics and Computer Science, UU. 1999-14

R. Schiefer. *Viper, A Visualisation Tool for Parallel Program Construction.* Faculty of Mathematics and Computing Science, TUE. 1999-15

K.M.M. de Leeuw. *Cryptology and Statecraft in the Dutch Republic.* Faculty of Mathematics and Computer Science, UvA. 2000-01

T.E.J. Vos. *UNITY in Diversity. A stratified approach to the verification of distributed algorithms.* Faculty of Mathematics and Computer Science, UU. 2000-02

W. Mallon. *Theories and Tools for the Design of Delay-Insensitive Communicating Processes.* Faculty of Mathematics and Natural Sciences, RUG. 2000-03

W.O.D. Griffioen. *Studies in Computer Aided Verification of Protocols.* Faculty of Science, KUN. 2000-04

P.H.F.M. Verhoeven. *The Design of the MathSpad Editor.* Faculty of Mathematics and Computing Science, TUE. 2000-05

J. Fey. *Design of a Fruit Juice Blending and Packaging Plant.* Faculty of Mechanical Engineering, TUE. 2000-06

M. Franssen. *Cocktail: A Tool for Deriving Correct Programs.* Faculty of Mathematics and Computing Science, TUE. 2000-07

P.A. Olivier. *A Framework for Debugging Heterogeneous Applications.* Faculty of Natural Sciences, Mathematics and Computer Science, UvA. 2000-08

E. Saaman. *Another Formal Specification Language.* Faculty of Mathematics and Natural Sciences, RUG. 2000-10

M. Jelasity. *The Shape of Evolutionary Search Discovering and Representing Search Space Structure.* Faculty of Mathematics and Natural Sciences, UL. 2001-01

R. Ahn. *Agents, Objects and Events a computational approach to knowledge, observation and communication.* Faculty of Mathematics and Computing Science, TU/e. 2001-02

M. Huisman. *Reasoning about Java programs in higher order logic using PVS and Isabelle.* Faculty of Science, KUN. 2001-03

I.M.M.J. Reymen. *Improving Design Processes through Structured Reflection.* Faculty of Mathematics and Computing Science, TU/e. 2001-04

S.C.C. Blom. *Term Graph Rewriting: syntax and semantics.* Faculty of Sciences, Division of Mathematics and Computer Science, VUA. 2001-05

R. van Liere. *Studies in Interactive Visualization.* Faculty of Natural Sciences, Mathematics and Computer Science, UvA. 2001-06

A.G. Engels. *Languages for Analysis and Testing of Event Sequences.* Faculty of Mathematics and Computing Science, TU/e. 2001-07

J. Hage. *Structural Aspects of Switching Classes.* Faculty of Mathematics and Natural Sciences, UL. 2001-08

M.H. Lamers. *Neural Networks for Analysis of Data in Environmental Epidemiology: A Case-study into Acute Effects of Air Pollution Episodes.* Faculty of Mathematics and Natural Sciences, UL. 2001-09

T.C. Ruys. *Towards Effective Model Checking.* Faculty of Computer Science, UT. 2001-10

D. Chkliaev. *Mechanical verification of concurrency control and recovery protocols.* Faculty of Mathematics and Computing Science, TU/e. 2001-11

M.D. Oostdijk. *Generation and presentation of formal mathematical documents.* Faculty of Mathematics and Computing Science, TU/e. 2001-12

A.T. Hofkamp. *Reactive machine control: A simulation approach using χ.* Faculty of Mechanical Engineering, TU/e. 2001-13

D. Bošnački. *Enhancing state space reduction techniques for model checking.* Faculty of Mathematics and Computing Science, TU/e. 2001-14

M.C. van Wezel. *Neural Networks for Intelligent Data Analysis: theoretical and experimental aspects.* Faculty of Mathematics and Natural Sciences, UL. 2002-01

V. Bos and J.J.T. Kleijn. *Formal Specification and Analysis of Industrial Systems.* Faculty of Mathematics and Computer Science and Faculty of Mechanical Engineering, TU/e. 2002-02

T. Kuipers. *Techniques for Understanding Legacy Software Systems.* Faculty of Natural Sciences, Mathematics and Computer Science, UvA. 2002-03

S.P. Luttik. *Choice Quantification in Process Algebra.* Faculty of Natural Sciences, Mathematics, and Computer Science, UvA. 2002-04

R.J. Willemen. *School Timetable Construction: Algorithms and Complexity.* Faculty of Mathematics and Computer Science, TU/e. 2002-05

M.I.A. Stoelinga. *Alea Jacta Est: Verification of Probabilistic, Real-time and Parametric Systems.* Faculty of Science, Mathematics and Computer Science, KUN. 2002-06

N. van Vugt. *Models of Molecular Computing.* Faculty of Mathematics and Natural Sciences, UL. 2002-07

A. Fehnker. *Citius, Vilius, Melius: Guiding and Cost-Optimality in Model Checking of Timed and Hybrid Systems.* Faculty of Science, Mathematics and Computer Science, KUN. 2002-08

R. van Stee. *On-line Scheduling and Bin Packing.* Faculty of Mathematics and Natural Sciences, UL. 2002-09

D. Tauritz. *Adaptive Information Filtering: Concepts and Algorithms.* Faculty of Mathematics and Natural Sciences, UL. 2002-10

M.B. van der Zwaag. *Models and Logics for Process Algebra.* Faculty of Natural Sciences, Mathematics, and Computer Science, UvA. 2002-11

J.I. den Hartog. *Probabilistic Extensions of Semantical Models.* Faculty of Sciences, Division of Mathematics and Computer Science, VUA. 2002-12

L. Moonen. *Exploring Software Systems.* Faculty of Natural Sciences, Mathematics, and Computer Science, UvA. 2002-13

J.I. van Hemert. *Applying Evolutionary Computation to Constraint Satisfaction and Data Mining.* Faculty of Mathematics and Natural Sciences, UL. 2002-14

S. Andova. *Probabilistic Process Algebra.* Faculty of Mathematics and Computer Science, TU/e. 2002-15

Y.S. Usenko. *Linearization in μCRL.* Faculty of Mathematics and Computer Science, TU/e. 2002-16

J.J.D. Aerts. *Random Redundant Storage for Video on Demand.* Faculty of Mathematics and Computer Science, TU/e. 2003-01

M. de Jonge. *To Reuse or To Be Reused: Techniques for component composition and construction.* Faculty of Natural Sciences, Mathematics, and Computer Science, UvA. 2003-02

J.M.W. Visser. *Generic Traversal over Typed Source Code Representations.* Faculty of Natural

Sciences, Mathematics, and Computer Science, UvA. 2003-03

S.M. Bohte. *Spiking Neural Networks*. Faculty of Mathematics and Natural Sciences, UL. 2003-04

T.A.C. Willemse. *Semantics and Verification in Process Algebras with Data and Timing*. Faculty of Mathematics and Computer Science, TU/e. 2003-05

S.V. Nedea. *Analysis and Simulations of Catalytic Reactions*. Faculty of Mathematics and Computer Science, TU/e. 2003-06

M.E.M. Lijding. *Real-time Scheduling of Tertiary Storage*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2003-07

H.P. Benz. *Casual Multimedia Process Annotation – CoMPAs*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2003-08

D. Distefano. *On Modelchecking the Dynamics of Object-based Software: a Foundational Approach*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2003-09

M.H. ter Beek. *Team Automata – A Formal Approach to the Modeling of Collaboration Between System Components*. Faculty of Mathematics and Natural Sciences, UL. 2003-10

D.J.P. Leijen. *The λ Abroad – A Functional Approach to Software Components*. Faculty of Mathematics and Computer Science, UU. 2003-11

W.P.A.J. Michiels. *Performance Ratios for the Differencing Method*. Faculty of Mathematics and Computer Science, TU/e. 2004-01

G.I. Jojgov. *Incomplete Proofs and Terms and Their Use in Interactive Theorem Proving*. Faculty of Mathematics and Computer Science, TU/e. 2004-02

P. Frisco. *Theory of Molecular Computing – Splicing and Membrane systems*. Faculty of Mathematics and Natural Sciences, UL. 2004-03

S. Maneth. *Models of Tree Translation*. Faculty of Mathematics and Natural Sciences, UL. 2004-04

Y. Qian. *Data Synchronization and Browsing for Home Environments*. Faculty of Mathematics and Computer Science and Faculty of Industrial Design, TU/e. 2004-05

F. Bartels. *On Generalised Coinduction and Probabilistic Specification Formats*. Faculty of Sciences, Division of Mathematics and Computer Science, VUA. 2004-06

L. Cruz-Filipe. *Constructive Real Analysis: a Type-Theoretical Formalization and Applications*. Faculty of Science, Mathematics and Computer Science, KUN. 2004-07

E.H. Gerding. *Autonomous Agents in Bargaining Games: An Evolutionary Investigation of Fundamentals, Strategies, and Business Applications*. Faculty of Technology Management, TU/e. 2004-08

N. Goga. *Control and Selection Techniques for the Automated Testing of Reactive Systems*. Faculty of Mathematics and Computer Science, TU/e. 2004-09

M. Niqui. *Formalising Exact Arithmetic: Representations, Algorithms and Proofs*. Faculty of Science, Mathematics and Computer Science, RU. 2004-10

A. Löh. *Exploring Generic Haskell*. Faculty of Mathematics and Computer Science, UU. 2004-11

I.C.M. Flinsenberg. *Route Planning Algorithms for Car Navigation*. Faculty of Mathematics and Computer Science, TU/e. 2004-12

R.J. Bril. *Real-time Scheduling for Media Processing Using Conditionally Guaranteed Budgets*. Faculty of Mathematics and Computer Science, TU/e. 2004-13

J. Pang. *Formal Verification of Distributed Systems*. Faculty of Sciences, Division of Mathematics and Computer Science, VUA. 2004-14

F. Alkemade. *Evolutionary Agent-Based Economics*. Faculty of Technology Management, TU/e. 2004-15