This copy has been supplied by the Library of the University of Otago on the understanding that the following conditions will be observed:

1. To comply with s56 of the Copyright Act 1994 [NZ], this thesis copy must only be used for the purposes of research or private study.

2. The author's permission must be obtained before any material in the thesis is reproduced, unless such reproduction falls within the fair dealing guidelines of the Copyright Act 1994. Due acknowledgement must be made to the author in any citation.

3. No further copies may be made without the permission of the Librarian of the University of Otago.
An Introduction to Agent-based Modelling$^1$

Peter Green

July 2007

$^1$A dissertation submitted in partial fulfillment of the requirements for the degree of Master of Business at the University of Otago, Dunedin, New Zealand
Acknowledgements

Robert for incredible patience.
Pisey for company at the library and during coffee breaks.
Jenny for proof-reading (any remaining errors, of course, are mine).
LIST OF FIGURES

0.1 Schelling’s model of residential segregation. .................. 3
1.1 Canonical payoffs for the Prisoner’s Dilemma game. .......... 7
1.2 Some strategies and best responses. ............................ 9
1.3 Subgame perfect Nash equilibrium outcomes. .................. 11
1.4 A table representing the strategy TIT-FOR-TAT. ............... 16
1.5 Finite State Machine representing the strategy TIT-FOR-TAT. . 17
1.6 Pseudo-equilibria in the Eight-turn Prisoner’s Dilemma. ........ 20
3.1 An example of cross-over. ...................................... 35
3.2 An observation is mapped to a response by a classifier. ....... 36
Introduction

Start with a chessboard. Take some black and white checkers and place them at random on the board, leaving one square empty. Now go through the squares one-by-one. Move the checker in that square to the empty square if two conditions are met: One, there are more other-coloured checkers adjacent to this square than there are same-coloured; Two, the empty square has a higher ratio of same-coloured to other-coloured neighbours than this square. The evolution of the board should resemble figure 0.1.

Figure 0.1: Schelling’s model of residential segregation.

This is Thomas Schelling’s model of residential segregation (Schelling 1978). By modelling each agent as a checker on a chessboard, each agent is given a unique location, and so has a unique view of the world. The agents’ behaviour is determined by interaction with the other agents — their decisions are based
on the colour of the neighbours of each location under consideration. The surprising global result of the model is an emergent property of the model's local specification. Despite only having moderate preferences (no one wants to be a minority in their neighbourhood), the agent's behaviour leads to an extreme outcome with (nearly) maximal segregation.

These properties — autonomous interacting agents, leading to emergent properties upon aggregation — are characteristic of Agent-based Modelling (Tesfatsion 2002). Agent-based models give us a way to model the aggregation of heterogeneous agents, a feat that is nearly impossible in a deductive framework.

Because these models cannot be solved exactly, they will often be explored using computer simulations. Computers are an important tool in this field, but they are not central to the methodology. A simple model like Schelling's can be investigated using a few toys from the games cupboard.

The ability to program simulated models is a lower barrier to entry than the ability to build tractable analytical models. As a result the field is tending towards breadth rather than depth. Creating a new model ex nihilo is more straightforward and more rewarding than adapting an existing model (Axelrod 2003). This creative “anarchy” makes it difficult to compare and to replicate results (Leombruni et al. 2006). The methodology itself is promising though, and a more disciplined approach could make significant contributions to economics.

This review presents the motivation behind agent-based modelling, and its epistemological justification.

Chapter 1 describes the application of agent-based modelling to the iterated prisoner’s dilemma. This game proved intractable to conventional game-theoretic analysis. It has an infinite number of Nash equilibria, yet none of those equilibria are evolutionarily stable. Robert Axelrod’s solution to the problem using agent-based simulations satisfied most people. For those who wanted a more rigorous answer, it supplied clues to where that answer might lie. The prisoner’s dilemma provides motivation for the use of agent-based modelling, an introduction to some of its techniques, and warnings of some of its shortcomings.

The philosophy behind agent-based modelling is described in Chapter 2. Economics can be done by building theoretical models, or by doing empirical
research. Agent based modelling offers a "third way" of doing economics. Agent-based modelling can be used as a substitute for traditional analysis by providing generative explanations of economic phenomena. And it can be used to complement traditional analysis by testing the sensitivity and robustness of mathematical results. There are serious difficulties that prevent agent-based modelling from offering a true alternative to orthodox economics. These problems, and some of the solutions suggested for them, are explored.

Chapter 3 gives an overview of the sorts of agents that can be used as the basis of an agent-based model. The simplest agents make random choices, which result in surprisingly ordered emergent behaviour. More complicated agents implement some kind of learning model.
In this chapter we look at the iterated prisoner's dilemma, a model used to study cooperation between self-interested individuals. Standard analytical tools using rational agents only provided limited insight into this model. The game has too many Nash equilibria but no evolutionarily stable strategies. Computer simulation made more progress in understanding the game. This provides a demonstration of the usefulness of agent-based modelling. More recent analytical results, building on the simulation work, show some of the limits of the simulation approach.

1.1 Prisoner's Dilemma

The *Prisoner's Dilemma* was invented in 1950 by Merril Flood and Melvin Dresher, to demonstrate a difficult case in Von Neumann’s theory of games. The dilemma was named by Albert Tucker, who recast the game as a story about two prisoners (Poundstone 1993, pp. 117-9).

The game has two players who each may choose from two strategies: cooperation (C) or defection (D). Mutual cooperation earns the *Reward* payoff, \( R \), and mutual defection earns the *Punishment* payoff, \( P \). If one player defects while the other cooperates, the players get the *Temptation* (\( T \)) and
Figure 1.1: Canonical payoffs for the Prisoner’s Dilemma game.

<table>
<thead>
<tr>
<th></th>
<th>Player 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Player 1</td>
<td>C</td>
</tr>
<tr>
<td>C</td>
<td>3,3</td>
</tr>
<tr>
<td>D</td>
<td>5,0</td>
</tr>
</tbody>
</table>

Sucker’s (S) payoffs respectively. The Prisoner’s Dilemma requires that $T > R > P > S$; “winning” and “losing” are the extreme outcomes, and mutual cooperation beats mutual defection. The restriction $2R > T + S$ is usually applied; mutual cooperation gives the highest combined score. The canonical (Axelrod 1980a) values for the payoffs are given in figure 1.1.

No matter what move the other player makes, the optimal reply is defection (Webb 2007, p. 63). Therefore $(D,D)$ is a Nash Equilibrium — in this state neither player can unilaterally improve their position by changing their move. But both players would be better off at $(C,C)$, that is, a move from $(D,D)$ is a Pareto-improvement, and so $(D,D)$ is not Pareto-optimal. At the Pareto-optimal states, the cooperating player has incentive to defect (at the other player’s expense), so these states are unstable.

1.1.1 Iterated Prisoner’s Dilemma

The prisoner’s dilemma acquires greater strategic depth when it is turned into a repeated game. In this form it is known as the repeated prisoner’s dilemma, or the iterated prisoner’s dilemma. In the iterated prisoner’s dilemma it is not always rational to defect, because cooperating on this turn can increase the odds of your opponent cooperating in future turns.

The simplest way to iterate the prisoner’s dilemma is to repeat the game a fixed, finite number of times. When the game has a known length, strategies can take advantage of the end stages of the game to defect when there is less time available for retaliation. For this reason the game is often repeated indefinitely, with a fixed (small) probability, $w$, of ending the game after any turn.

The game can also be played (in theory) for an infinite number of turns.
Scoring in the finitely repeated game is simply a matter of summing or averaging payoffs; this is obviously not possible in the infinite game. The usual way to score the infinite game is to discount the payoffs from future moves. If the payoff from turn $t$ is $p_t$, then the total score is $p_1 + r \cdot p_2 + r^2 \cdot p_3 + \ldots$.

If the discount rate $r$ is set to $1 - w$, where $w$ is the probability of ending the game, then the expected payoff for the indefinitely repeated game is equal to the payoff from the infinitely repeated game with discounted payoff; the two versions of the game are essentially the same.

An alternative method of scoring the infinite game uses the limit of the mean of the payoffs\footnote{See, for example Abreu & Rubinstein (1988) or Binmore & Samuelson (1992).}, that is $s = \lim_{t \to \infty} \frac{1}{t} \sum p_t$. The limit of the mean is undefined for some sequences of payoffs, although some limited classes\footnote{Pure strategies with finite memories must eventually settle into a repeating sequence of payoffs, guaranteeing the mean payoff will converge.} of strategies do guarantee convergence. Any moves prior to settling into a final pattern become irrelevant using this method of scoring. Limit of the mean will give the same score to the strategies GRIM and ALL-D\footnote{See figure 1.2.} if they play each other, since after the initial Temptation/Sucker payoffs both strategies settle into mutual defection. Strategies can also perform costless “handshaking” to recognise copies of themselves in the early turns of the game (Binmore & Samuelson 1992) and any “thrown” turns early on will be dominated by the later turns in the game. Because of these two limitations, the usual discounted scoring method will be assumed in the rest of this chapter.

In the one-turn prisoner's dilemma, defection is the rational choice no matter what strategy your opponent is using. In the repeated game, optimal play depends on your opponent's strategy. Figure 1.2 gives some common strategies and the best responses to those strategies. The best responses to a strategy will form a class of strategies; there is a sequence of moves that will maximise your score, and any strategy that plays these moves against this strategy will be a best response.
Figure 1.2: Some strategies and best responses.

<table>
<thead>
<tr>
<th>Strategy</th>
<th>Description</th>
<th>Best Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIT-FOR-TAT</td>
<td>Cooperate on the first turn; then always play the opponent’s previous move.</td>
<td>Cooperate.</td>
</tr>
<tr>
<td>GRIM</td>
<td>Cooperate until the opponent defects; then defect forever.</td>
<td>Cooperate.</td>
</tr>
<tr>
<td>ALL-C</td>
<td>Always cooperate.</td>
<td>Defect.</td>
</tr>
<tr>
<td>ALL-D</td>
<td>Always defect.</td>
<td>Defect.</td>
</tr>
<tr>
<td>PAVLOV</td>
<td>Cooperate on the first turn; repeat your last move if you got R or T; play the opposite move if you got P or S.</td>
<td>Defect.</td>
</tr>
<tr>
<td>TF2T</td>
<td>Tit for two tats. Defects if and only if its opponent’s last two moves were defections.</td>
<td>Alternate D/C.</td>
</tr>
</tbody>
</table>

1.2 Analytical Solutions

1.2.1 Finitely Repeated Prisoner’s Dilemma

If the Prisoner’s Dilemma is repeated a finite number of times, it has a unique Nash equilibrium: always defect. This result can be proved by backwards induction. The last turn is equivalent to a one-turn prisoner’s dilemma — there are no future turns in which we might hope for reciprocated cooperation. The only rational way to play the last move is to defect. But since the outcome of the last turn is known, the second-to-last turn is also equivalent to the one-turn game. So both players should defect on the second-to-last turn too. This line of reasoning eventually carries back to the first move, and so a Nash-rational player will defect on every turn.

Unfortunately this clear game theoretic result has neither positive nor normative value. People do not play this strategy (Oskamp 1971, Andreoni & Miller 1993), even when they are aware of it (Axelrod 1998). Nor is it sen-
sible for people to play the always defect strategy. While the strategy is optimal against a Nash-rational opponent, it performs poorly against plausible strategies (Axelrod 1980a).

1.2.2 Subgame Perfect Nash Equilibria

A pair of strategies $i, j$ form a Nash Equilibrium if they are best responses to each other, i.e., where $V(i, j)$ is the payoff for playing strategy $i$ against strategy $j$,

$$\forall k, \ V(i, j) \geq V(k, j) \quad [1]$$
$$V(j, i) \geq V(k, i) \quad [2].$$

If [1] did not hold, we should expect $k$ to be played instead of $i$. Or, if [2] did not hold, $k$ instead of $j$. A single strategy is called a Nash Equilibrium if it forms a Nash Equilibrium when paired with itself.

Strategies should also be best responses for each subgame. If there is a better response $j$ than some strategy $i$ for some subgame$^4$, then we would expect a player to switch from $i$ to $j$ at that subgame. Equilibria where a pair of strategies are best responses to each other at every subgame$^5$ are called subgame perfect Nash equilibria.

The familiar ultimatum game between two rational players illustrates the need for a subgame perfect equilibrium. Player One must propose a way to split $100 between the two players. Player Two may accept the proposal, or reject it. If Player One’s proposal is rejected, both players get nothing. The optimal strategy for Player Two is to reject any proposal that gives them less than $99. In this case, Player One has a choice between $1 and nothing, and will choose the $1. But Player Two cannot credibly commit to this strategy. If Player One offers Player Two $1, Player Two has a choice between $1 and nothing, and will choose the $1. While every strategy pair (Offer $x$, Reject < $x$) is a Nash equilibrium, only (Offer $1$, Accept ≥ $1$) is subgame perfect.

$^4$A game starting at turn $t$, with a specific history up until turn $t$.

$^5$Including subgames that will not actually occur.
GRIM⁶ is a subgame perfect strategy for the indefinitely repeated prisoner's dilemma. So there is a subgame perfect Nash equilibrium that produces the reward payoff for both players. In fact, by the Folk Theorem, any pair of payoffs that is

a) bounded by the one-turn game payoff pairs, and

b) gives each player at least the one-turn game Nash equilibrium payoff

is the result of some subgame perfect Nash equilibrium (Webb 2007, pp. 130-1). Figure 1.3 shows this graphically. The larger area shows the outcomes that are possible for some sequence of moves and discount rate⁷. The smaller area shows outcomes that are the result of some subgame perfect Nash equilibrium and some discount rate.

⁶If defined as “cooperate unless either player has defected”. The usual definition, “cooperate unless the other player has defected”, is not subgame perfect. If these two strategies are equivalent in some sense, then subgame perfect is not well-defined in that sense.

⁷The payoffs have been scaled for simplicity: e.g. \( R \) should actually be \( \frac{R}{1-p} \), the total payoff for getting \( R \) every turn.
This leads to the question of which equilibrium will be chosen. We can reduce the number of equilibria under consideration by considering their stability.

1.2.3 Evolutionarily Stable Strategies

A strategy is an evolutionarily stable strategy if a population playing the strategy cannot be invaded by "mutants" playing another strategy (Maynard Smith 1998, p. 14). Specifically, a strategy $x$ is an evolutionarily stable strategy if

$$\forall \text{ strategies } y \neq x, \text{ either } V(x, x) > V(y, x)$$

or

$$V(x, x) = V(y, x) \text{ and } V(x, y) > V(y, y)$$

Unfortunately, there are no evolutionarily stable strategies for the iterated prisoner’s dilemma (Boyd & Lorberbaum 1987, Lorberbaum 1994).

1.3 Solution by Simulation

There is a large body of research investigating how the iterated prisoner’s dilemma is played (Oskamp 1971). However, this research focused on inexperienced players, and provided little insight on how to play the game well. Robert Axelrod organised two computer tournaments (Axelrod 1980a, Axelrod 1980b), inviting game theorists from various disciplines — e.g. economics, political science, and psychology — to submit strategies in the form of computer programs.

1.3.1 Axelrod’s Tournaments

In the tournament each program was played against every other program, in a round-robin. Each program also played against the strategy RANDOM, which had a 50/50 probability of cooperating/defecting each turn. In the first tournament the game was the 200 turn finitely repeated prisoner’s dilemma. In the second the strategies played several indefinitely repeated prisoner’s
dilemmas, where the probability the game would end after any given turn was $w = 0.00346$, giving an expected median game length of 200 moves. Final scores were determined by summing the programs' scores in each match; the programs were then ranked by their final scores.

1.3.2 Tournament Results

The strategy TIT-FOR-TAT won both tournaments, but not by a large margin. Axelrod was able to identify several properties that characterised successful strategies (Axelrod & Dion 1988). There is no "best" strategy for the iterated prisoner's dilemma, since the optimal strategy depends on the mix of other strategies. But strategies with these properties should perform well in most settings.

Nice

A nice strategy is a strategy which is never the first to defect. Two nice strategies will therefore score an uninterrupted string of Reward payoffs. This is usually the best result a player can expect since a strong strategy is unlikely to allow its opponent a large number of Temptation payoffs.

Provocable

A strategy should punish its opponent's defections by retaliating. A strategy that is not provoked by defections can be exploited by an opponent taking Temptation payoffs, whereas punishing a defection may encourage an opponent to cooperate.

---

8The number of turns was actually predetermined so that every pair of programs played a series of games with the same lengths. Since the entrants were not told these lengths, they were still technically playing the indefinitely repeated prisoner's dilemma.

9It was submitted to both tournaments by Anatol Rapoport.
Forgiving

A non-forgiving strategy such as GRIM can get trapped in a long or permanent series of mutual defections, earning Punishment payoffs. A forgiving strategy will respond to its opponent’s attempts to restart mutual cooperation (e.g. TIT-FOR-TAT), or even make such attempts itself (e.g. PAVLOV), making a return to Reward payoffs possible.

1.3.3 Evolutionary Approach

Anticipating the results of future tournaments

Very forgiving strategies, such as TF2T, could have won the first tournament if they were entered. Some participants followed the lessons of the first tournament and entered nice and forgiving strategies in the second. Other participants anticipated this, and tried to write programs that would exploit the more forgiving strategies. While the exploiting strategies were not successful in the second tournament, they did manage to take the more forgiving strategies down with them: TIT-FOR-TWO-TATS would have done poorly in the second tournament. This suggested a kind of evolution in the strategies entered; programmers would examine the programs and results from the previous tournament, and adjust their programs for the next.

Ecological and Evolutionary Simulation

Rather than hold more tournaments, Axelrod attempted to simulate the process (Axelrod 1997). Since a simulation could not produce new strategy programs, the 63 strategies from the second tournament were used. The frequencies of each program in the population of each successive simulated tournament were allowed to vary: the more successful a program was, the more copies were entered into the next round. Axelrod called this the ecological approach. The major drawback of the ecological approach was its inability to simulate small adjustments to the programs, let alone to introduce entirely new programs.

To simulate these sorts of changes, Axelrod used a genetic algorithm (Holland
1992), at some cost to the complexity of the strategies represented. Axelrod called this an evolutionary approach. A genetic algorithm requires strategies to be encoded into a genome. Axelrod used strings of Cs and Ds, representing all the possible strategies with a memory of at most three turns. New strategies were created by mutating a string — changing a C to a D or visa versa — or by crossing over two strings — adding the end of the second string to the beginning of the first.

At this point Axelrod was engaged in agent-based modelling. Each simulated player was an agent, with its own independent strategy. These player-agents interacted with each other by competing in the iterated prisoner's dilemma. Cooperative behaviour emerged in most runs of the simulation, with agents adopting strategies similar to TIT-FOR-TAT.

1.4 Agent-based Models

1.4.1 Agent Representation

Moving from programs submitted by actual people, to strategies generated by agents, leads to the question of how to represent agents in the model. Using rational agents is not the answer, since the nature of “rational” play in the iterated prisoner’s dilemma is unknown.

The representation should cover a complete class (Beaufils et al. 1998) of strategies, to prevent biasing the outcome with an arbitrary sample (Linster 1992). There should also be a path from the initial distribution of strategies to any equilibrium (Nachbar 1992).

Agents with Finite Memory

Axelrod’s evolutionary simulation (Axelrod 1997) allowed for every possible strategy with three-turn memory. With two players and two choices, there are four possible outcomes for any one turn, corresponding to the Reward, Sucker, Temptation, and Punishment payoffs. For a history of three turns, there are 64 different outcomes. A string of 64 genes, each a C or a D, specifies a strategy’s response to every possible three-turn history.
Figure 1.4 shows the TIT-FOR-TAT strategy represented as a strategy with a single turn memory, with a response coded for each possible result from the previous turn. An additional gene is required to specify behaviour for the first turn. Axelrod used an additional six genes to specify a fictional history for the agents’ initial memories. The agents used this fictitious history to play the first three moves.

If a strategy remembers only its opponent’s past moves, the representation is more compact (Lomborg 1996). But some interesting strategies like GRIM can no longer be represented\(^\text{10}\). Exploitation becomes less likely, because a player cannot remember if it is cooperating with or exploiting its opponent.

**Finite State Machines**

A *finite state machine* represents a strategy as a collection of states. Each state has a choice, C or D, associated with it, and a transition to another state (possibly itself) for each choice, C or D, an opponent might make. One state is designated as the machine’s initial state. A finite state machine plays the choice associated with the current state, then moves to a new state based on its opponent’s move. Figure 1.5 shows a two state machine that plays the TIT-FOR-TAT strategy. The two circles are the two states. When the machine is in the left-hand state it plays C, in the right-hand state D. The bold outlined left-hand circle is the initial state. The arrows are the transitions; each arrow is labelled with the opponent’s move that triggers that transition.

Finite state machines allow increased complexity at less cost than the simple finite memory representation\(^\text{11}\), while still providing a complete class of strategies.

\(^{10}\) GRIM must either remember both players’ last moves, or remember an opponents defection for an indefinite length of time.

\(^{11}\) Storage requirements for a finite state machine are $O(n \log n)$ in the number of states, compared to $O(c^n)$ in the length of memory for the simple finite memory representation.
Strategies' complexity can be measured easily in this representation, by the number of states (Abreu & Rubinstein 1988, Binmore & Samuelson 1992), or by the number of transitions (Linster 1992).

This representation is only useful when the machines are allowed to have at least three states. The set of two state machines is equivalent to the set of strategies with single turn memories\(^{12}\). These strategies can be represented more simply by tables as shown in figure 1.4. The use of finite state machines in this case distracts from the fact that the results are due to the simplicity of the strategies, as it does in Linster (1992).

**Lisp Programs**

Strategies can be represented as programs. Using Lisp\(^{13}\) programs makes it possible to apply the genetic algorithm to these programs (Scali 2006), a process known as genetic programming. This approach produces simple strategies like TIT-FOR-TAT or GRIM, rather than the complicated programs typical of the tournament submissions.

**The Beaufils-Delahaye-Mathieu Genotype**

The strategy GRADUAL (Beaufils et al. 1996) responds to defections with increasingly long sequences of retaliatory defections. This strategy cannot be represented with finite memory or as a finite state machine. A genetic

---

\(^{12}\)Linster claims (Linster 1992) that it is incorrect to say these machines have only a one turn memory. GRIM, for example, can remember its opponent's first defection forever. This is technically true, but it misses the point: while the two state machine representation may implement a strategy by using an "infinite" memory, the strategy can also be implemented as a function of only the previous move.

\(^{13}\)A programming language designed for studying artificial intelligence.
code was created that allows GRADUAL, as well as most common strategies, to be represented.

Because this encoding was specifically designed to accommodate GRADUAL, it is unlikely to find genuinely new strategies. The optimal strategies found with this encoding were subtle variations on GRADUAL.

1.4.2 Evolutionary Results

Agents learn to defect, then to retaliate, then to cooperate (Axelrod 1997). In the early stages of simulation (10-20 generations) agents will learn to defect. This is the optimal response to more-or-less random initial strategies. Agents are defecting, and their opponents are defecting. Some agents will encode this as “defect in response to defection”. These agents are provicable. Some of the provicable agents will cooperate in response to cooperation. They will be rewarded when interacting with other cooperators. Since they are provicable, they will not lose too much against non-cooperators. Eventually most of the population will be provicable and cooperative.

Strategies have been found that outperform TIT-FOR-TAT in plausible settings, such as PAVLOV (Nowak & Sigmund 1993), GRADUAL (Beaufils et al. 1996), and FORGIVING (O’Riordan 2000). These strategies are nice, provicable, and forgiving. This underlines the fact that TIT-FOR-TAT is not the best strategy, but is a simple example of the qualities required for a strategy that is good in a wide variety of plausible settings. PAVLOV has these properties, and is also robust in a noisy environment (Nowak & Sigmund 1993). TIT-FOR-TAT does not perform well with noise since it can get trapped in alternating retaliation or mutual defection. The modified strategy GTFT\(^{14}\) is robust with noise (Axelrod 2000), but cedes the advantage of simplicity to PAVLOV.

When the population is limited to the 26 single turn memory strategies, GRIM is the most successful (Linster 1992). GRIM is nice and provicable, but not forgiving. Forgiveness is only useful if strategies are sophisticated enough to repent after probing for an exploitable strategy. Niceness and

\(^{14}\)Generous TIT-FOR-TAT, which has a small but positive probability of cooperating whenever TIT-FOR-TAT would defect.
provocability are more fundamental than forgiveness. All four\textsuperscript{15} nice and provocable strategies do well in this simple setting. GRIM does better than the other three because it gets high scores against implausible strategies like DC*\textsuperscript{16}.

\section*{1.5 Further Analytical Results}

\subsection*{1.5.1 Criticism from Game Theory}

Social simulations often converge to something other than a Nash equilibrium of the underlying game. For instance, many simulations of the finitely repeated prisoner’s dilemma fail to converge to the Nash equilibrium ALL-D. Backwards induction proves that perfectly rational agents would choose the Nash equilibrium ALL-D, and if all strategies are available, simple learning agents will converge to the Nash equilibrium (Nachbar 1992).

Such results conflict with the “underlying theory” (Binmore 1998). But a game theoretic solution to the underlying game does not make simulation unnecessary. Nash equilibrium assumes Nash-rational agents, which may be an oversimplification. A simulation solution can be an alternative to an unsatisfactory game theoretic solution (Marney & Tarbert 2000).

\subsection*{1.5.2 Pseudo-equilibria}

A simulation may fail to converge to a Nash equilibrium if it is not run for long enough. When the N-turn prisoner’s dilemma is played by agents using the strategies TFT\textsuperscript{n} the population passes through a series of pseudo-equilibria (Nachbar 1992). This process is shown for the 8-turn prisoner’s dilemma in figure 1.6.

\textsuperscript{15}TIT-FOR-TAT, PAVLOV, GRIM, and the strategy that defects only after a sucker payoff.

\textsuperscript{16}DC* defects once then always cooperates.

\textsuperscript{17}TFT\textsubscript{n} starts by playing TFT but defects in the last n turns. These strategies are only available in the finitely repeated prisoner’s dilemma, since the player must know how many turns remain.
In a population playing mostly TFT\(_n\), the optimal strategy is TFT\(_{(n+1)}\). The only strategy immune to invasion is ALL-D. Since TFT\(_{n+1}\) is weak in a population playing TFT\(_{(n-1)}\), it will be uncommon when TFT\(_n\) invades, allowing a pseudo-equilibrium of TFT\(_n\) to persist for some time. Note that each successive pseudo-equilibrium in figure 1.6 lasts for longer than the previous. This may however be a result of the specific evolutionary dynamic.\(^{18}\)

This insight provides us with two lessons. First, apparent convergence by a simulation may be temporary. Second, the dynamics of a model can be more important than its equilibria.

1.5.3 Equilibrium Refinement

Relaxations of the ESS concept can give more satisfactory analytical results. For example, Binmore & Samuelson’s (1992) modified ESS allows for \(V(x, x) = V(y, x)\) and \(V(x, y) = V(y, y)\), so long as \(y\) is no more complex than \(x\). In a slightly modified form of the game, mutual cooperation is the

\(^{18}\)A continuous population with no mutation. In a finite population the potential invader might be extinct, making the pseudo-equilibrium stable. Mutation can shorten the pseudo-equilibria by supplying the invading strategy in larger numbers.
unique modified ESS. But this requires limit-of-the-mean scoring, a complexity measure for strategies, and complexity in the agents’ utility functions.

Bendor & Swistak (1997) introduces the weak ESS. A strategy is a weak ESS if it cannot be invaded by an arbitrarily small population of another strategy. Amongst weak ESS’s, nice and retaliatory strategies have a minimal stabilising frequency that approaches ½ as the discount rate approaches one. If more than half the population is nice and retaliatory, the population will converge to an equilibrium with full cooperation. No strategy has a minimal stabilising frequency less than ½, and any strategy that is neither almost-nice nor almost-retaliatory requires a higher frequency. Under this scheme ALL-D is the least stable, with a minimal stabilising frequency approaching one as \( r \to 1 \).

1.6 Conclusion

The iterated prisoner’s dilemma shows how a problem, intractable to traditional analytic techniques, can be approached through agent-based simulations. The widespread acceptance of Axelrod’s TIT-FOR-TAT result shows that simulated solutions can be satisfactory. And if a rigorous solution is required, simulation results can act as a guide for new analytical solutions.

Agent-based modelling can be used as an alternative to traditional theoretical analysis, or as a complement. Chapter 2 outlines the methodology of agent-based modelling, contrasting it with traditional methods.

---

\(^{19}\)This is dependent on the particular evolutionary dynamic, and no strategy is a weak ESS under every dynamic.

\(^{20}\)A strategy is almost-nice if its score against another player using the same strategy approaches the score of two nice strategies as \( r \to 1 \).

\(^{21}\)A strategy is almost-retaliatory if its score against any opponent approaches at least its opponents score as \( r \to 1 \).
2.1 Deduction, Induction, Simulation

Theoretical economics uses a deductive methodology. A set of assumptions is formalised as a mathematical model. Then, conclusions are drawn using more or less rigorous proofs. Mathematical certainty comes at a high price: only the simplest models are analytically tractable.

Empirical economics uses an inductive approach. Statistical methods are used to find patterns in real-world data. The strength of this approach is its contact with the real world, but it lacks the certainty associated with analytical proofs.

Simulation is a third approach (Axelrod 2003) to economics. Simulation is a response to the limits of the analytical approach, especially the difficulties associated with heterogeneous agents and interactions amongst agents.

2.1.1 Analytical Methods

Using an analytical approach allows us to prove theorems. If the assumptions of a theorem are true, then the consequences of the theorem are true in
general. Simulation results, in contrast, provide us with anecdotal evidence of the consequences of our assumptions.

The certainty of the analytical approach comes at a price: in order that the model be analytically tractable, the assumptions often have to be simplified beyond credibility. Simulation places fewer restrictions on the assumptions we can investigate.

### 2.1.2 Empirical Methods

Empirical investigations use inductive techniques to find patterns in real world data. The use of data from real economies is an obvious advantage of this approach. The main downside is the extreme difficulty in separating causality from correlation. Experiments can achieve this, but natural experiments are rare, and planned experiments with economies are difficult due to costs and ethical considerations. These problems are made worse when data is scarce and difficult to collect.

Inductive techniques are required to extract patterns from the output of simulations. While this might make the analysis of an agent-based model appear similar to an empirical study, the analysis of simulated data has several advantages. The most obvious is that large amounts of clean, high quality data can be generated. If there is insufficient data, more can be generated. There is no missing data, and measurements are accurate. More importantly, running counterfactual simulations allows genuine experiments to be conducted with agent-based models (Axelrod 2003).

### 2.2 Other Simulation Methods

#### 2.2.1 Simulation for Forecasting

Simulations are often associated with models used for forecasting. But agent-based and forecasting models have significant differences.

Agent-based models are intended for theoretical research. Therefore, *qualitative* similarities with real economies are important; but forecasting requires
quantitative similarity. Complexity in a forecasting model is limited only by implementation costs and concerns of over-fitting, whereas an agent-based model should aim to provide as simple an explanation as possible.

2.2.2 Equation-based Simulation

Agent-based models have several advantages over models based on systems of equations among macroeconomic variables. Most importantly, they answer the Lucas critique by having explicit microfoundations (Richiardi 2003a). The correspondence between modelled agents and familiar real-world objects (as opposed to abstract quantities like GDP or the price level) makes agent-based models easier to implement, easier to experiment with, and easier to draw normative conclusions from. The use of agents also allows validation of the model at both the individual level and at the level of the full system being studied (Parunak et al. 1998).

2.3 A Substitute for Deductive Methods

2.3.1 Generative Explanation

Epstein (2007) claims that in order to explain a phenomenon, we must be able to generate that phenomenon. An analytical result that proves the existence, and even the stability of, an equilibrium is insufficient; we must also show that an economy will arrive at that state in the first place.

Generating the phenomenon is necessary for explanation, but it is not sufficient. For a full explanation, we also need to understand the mechanisms that lead from our assumptions to the generated outcome. Creating a simulation cannot satisfy this; the simulation must be analysed to find those mechanisms.

The same holds true for analytical results; we must take care that the conclusion is not contained trivially in the assumptions.
2.3.2 Inference to the Best Explanation

Simulation methodology employs \textit{inference to the best explanation}, also known as \textit{abduction} (Ladyman 2002). A simulation that generates a stylised fact is a potential explanation for that fact. A "best" explanation is chosen from competing explanations based on some criterion such as empirical adequacy or parsimony.

Macy & Willer (2002) describe the aim of agent-based modelling "to explore the simplest set of behavioural assumptions required to generate a macro pattern of explanatory interest". Simplicity is an important property in agent-based models (and in models in general). In order to add to theoretical understanding, it is not sufficient that a model be able to generate the pattern in question. We would also like to know why it generates that pattern. Unless the model is simple, that may prove no easier than the corresponding question about the real world.

2.3.3 Existence and Counter-examples

While a single example is insufficient to prove a general theorem, proving a \textit{negative} theorem only requires a single counter-example.

If a simplifying assumption from an analytical model is relaxed in an agent-based version, and the conclusion no longer holds, this shows that a generalisation of the analytical model will fail (Axtell 2000).

2.4 A Complement to Deductive Methods

2.4.1 Two Kinds of Generality

The conclusions of a deductive model hold with mathematical certainty over its entire parameter space. In contrast, the conclusions from a simulation run are contingent on specific parameters and initial conditions. In this sense, deduction is more general than simulation (Richiardi 2003b).
If the specific parameters, the initial conditions, and the random number seed are treated as "assumptions" of the model, then each run of the simulation can be considered a deductive "theorem" (Epstein 2006). Such a theorem can hardly be called general, since the number of assumptions is so large. But the difference is reduced to one of degree, rather than of kind.

Deductive models may also have more assumptions than we want. "Heroic" assumptions (LeBaron 2002) are often required to make a deductive model tractable; for example, rational expectations, representative agents, or Walrasian tatonnement. The "general" deductive result tells us nothing about what happens when these assumptions are not met.

Simulation can be used to test a deductive model's sensitivity to its simplifying assumptions (Axelrod 2000). A simulation model can be built to replicate the deductive result, then assumptions can be relaxed. For example, homogeneous agents might be replaced by a heterogeneous population. If the model is not robust to these changes, simulation may suggest generalisations of the model that are (Richardi 2003a).

2.4.2 Thought Experiments

Simulation can be used to conduct thought experiments (Axelrod 2003). If an economist wants to know the consequences of a set of assumptions, they might deduce those consequences mathematically. But depending on the assumptions, such an analysis may be difficult or impossible. Simulations can provide an easier alternative, acting as a powerful aid to intuition. An interesting simulation can identify fruitful models to which more rigorous techniques can then be applied.

2.4.3 Exploration of Analytical Models

Simulation can be used to explore models that could in principle (or even in practice) be solved by analytical methods. Simulation allows the researcher to get a feel for the properties of the model in question without having to prove these properties formally. This can help the researcher decide which models are promising enough to pursue in greater detail.
2.4.4 Out-of-equilibrium Dynamics

Even if a model can be solved analytically for its equilibria, the out-of-equilibrium behaviour might be more difficult to determine. Out-of-equilibrium behaviour of a model is important when there are multiple equilibria, when the equilibrium is very long run (e.g. Solow growth models), or when the fluctuations about the equilibrium are of particular interest (e.g. business cycles). Simulation allows us to explore the out-of-equilibrium dynamics of a model.

The out-of-equilibrium behaviour of a model might be necessary to reproduce real-world features. In the Santa Fe Artificial Stockmarket (LeBaron 2002), agents had to allocate a portfolio between a risk-free bond and a risky stock. The analytical equilibrium was already known, but simulating with boundedly rational agents, rather than solving with rational agents, gave volatility, trading volume, and excess kurtosis similar to real-world data.

2.5 Problems with Agent-based Modelling

2.5.1 Communication

Since agent-based modelling is a new field, and is highly interdisciplinary (Colander et al. 2004), it has not developed the sort of common language and assumptions typical of a more mature field (Axelrod 2003). This means that the model and procedures used to arrive at a result must be specified in detail and at length. This can draw attention away from, or leave too little room for, a discussion of the results (Leombruni et al. 2006).

Communicating the results can be difficult too. Summary statistics will not convey the interesting agent-level interactions that lead to them. There are too many agents to describe each one's behaviour. They are heterogeneous, so choosing “representative” agents to describe is difficult. Describing a narrative “history” of a simulation run (Axelrod 2003) may impose an interpretation that is not present in the data.

Results are best understood if readers can explore the model themselves (Colander et al. 2004). Ideally, the simulation program should be available on
the internet, although finding a sufficiently permanent URL for the software may be difficult.

2.5.2 Publication

Very few agent-based models appear in the top economics journals (Colander et al. 2004, Leombruni et al. 2006). Some referees object to the methodology entirely (Leijonhufvud 2006), although the technical details of an agent-based paper may simply make it difficult to tell which papers are good. In order to gain acceptance as a legitimate method, agent-based modellers should try to explain things which orthodox economics cannot (Colander et al. 2004, Leijonhufvud 2006).

The papers that do get published tend to be scattered across many journals (Axelrod 2003, Leombruni et al. 2006). In addition, many important early papers remained unpublished (Tesfatsion 2002). This situation is improving, with dedicated journals such as the Journal of Artificial Societies and Social Simulation1.

2.5.3 Replication

Claims made by researchers ought to be open to scrutiny by other researchers. For deductive results, this is as simple as checking that the proof is valid. Empirical studies can be repeated if other researchers have access to the dataset being studied, and this does in fact happen. Replication of simulation results is more difficult and less common.

Docking

An early attempt at replication is documented in Axtell et al. (1996). An attempt was made to reproduce Axelrod’s cultural transmission model (Axelrod 1995) within the larger Sugarscape model (Epstein 2006). The authors named the process docking.

1http://jasss.soc.surrey.ac.uk
Three grades of equivalence are suggested (Axelrod 2003): Numerical identity requires that the two models produce exactly the same output. For stochastic models, this requires using the same pseudo-random number generator. This level of equivalence is fairly strict, and so it will be uncommon in practice.

Distributional equivalence means that the distributions of the two models’ outputs are statistically indistinguishable. Unfortunately, success at creating distributional equivalence usually means failing to reject the null hypothesis of identical distributions. If two distributions are not identical, it is easier to tell them apart if large samples are taken. So using small samples make it more likely that the null will not be rejected and the two distributions will be considered indistinguishable. Therefore sample sizes large enough to reliably detect differences should be used.

Relational equivalence only requires that qualitative features of the model be reproduced. Since agent-based modelling is primarily concerned with the explanation of qualitative stylised facts, this may be sufficient. But whether or not relational equivalence has been achieved is a subjective judgement call.

The computer programs implementing two models would each have thousands of lines of code\(^2\). When a model is described in an article, some amount of detail is likely to be lost. When Axtell and Epstein tried to replicate Axelrod’s model, they used selection without replacement, instead of selection with replacement as used in the original model. This small error led to a failure of distributional equivalence from some initial conditions (although relational equivalence still held). Another discrepancy (the order the transmitting and receiving agent were chosen in) was caught in face-to-face discussions with Axelrod. Not all researchers can expect to have that advantage.

\(^2\)For example, Axelrod’s Cultural Model program has 5,000 lines of code, and Axtell and Epstein’s SugarScape program has 20,000.
Encouraging Replication

A decade ago Axtell et al. (1996) found little evidence of agent-based modellers attempting to replicate the results of older models. Their paper has been cited over 100 times, which suggests that activity has increased.

Access to an implementation’s source code gives future researchers a better reference for a model’s details than a paper can. Source code should be available online, and it should be well documented and freely distributable and modifiable (Leombruni et al. 2006). UML may be useful for specifying or documenting a model or its implementation.

Using widely available toolkits to implement models reduces the likelihood of implementation errors and makes it easier to share models with other researchers. No toolkit is considered standard yet, but at this early stage competition between a number of options is desirable (Colander et al. 2004).

2.5.4 Sensitivity

The limited replication that has been done reveals another problem with agent-based methodology: results may be specific to a particular implementation, model specification, and parameter set. Solving this problem requires extensive sensitivity analysis, both at the parameter level and the model specification level, as well as replication, which addresses sensitivity at the implementation level (Leombruni et al. 2006).

This criticism can also be made against analytical models, where results may be due to the simplifying assumptions required for tractability. Sensitivity analysis of docked agent-based models offers a solution to this problem as well.

---

3 The extensive replication and extension of Axelrod’s prisoner’s dilemma simulation notwithstanding.
4 Unfortunately the paper is not in the Social Sciences Citation Index. This estimate of the number of citations comes from Google Scholar.
5 Unified Modelling Language, a type of diagram used to describe object oriented systems.
6 For example, Repast (North et al. 2005a, North et al. 2005b)
Miller (1998) suggests the use of active nonlinear testing (ANTs) for sensitivity analysis in complex models. An ANT requires a nonlinear optimisation algorithm and an objective function for the algorithm to maximise. Miller suggests a genetic algorithm or a hill climbing algorithm for general use, unless a more efficient algorithm is well suited to a particular model. A typical objective function would be the distance (e.g. difference or difference squared) between the value of a given aggregate variable and the value of that variable under a set of default parameters. The algorithm is used to search over a set of allowed perturbations of the default parameters; the result indicates the sensitivity of the variable to the allowed perturbations.

2.5.5 Nonlinearity

Along with other nonlinear models, agent-based models suffer from the problem of equifinality (Richiardi 2003b). This is where many potential models explain the in-sample behaviour equally well, but make different predictions out-of-sample.

2.6 Conclusion

Analytical tractability requires heroic simplifying assumptions, which reduce the credibility of a model. If an intractable model is investigated via simulation, we are denied mathematical certainty. There is a trade-off here; each approach has its strengths and weaknesses. But both can tell us something useful about the world.

The economy is a complex system, made up of interacting, heterogeneous, people. To understand the economy, we need to model people and their interactions. In Chapter 3 we examine the sort of agents that can be used for this modelling.
3.1 Rational Agents

Typically, economic models assume that agents are “rational”, that is, they maximise some utility function and have rational expectations about the future (Becker 1962). Rational agents do not describe the behaviour of real individuals well. However, they provide a useful simplification that allows some models to be solved analytically (Axelrod 2003). Rational agents also have the advantage of being the default choice in economic models. Agents can be non-rational in a number of ways, and the use of a particular form of non-rational behaviour must be justified by the modeller. For a variety of reasons, agent-based models seldom use rational agents.

In some models it is impossible to use rational agents. Simulation can be used to study the behaviour of models that are analytically intractable; in such models the rules of rational behaviour are unknown to the modeller (e.g. the iterated prisoner’s dilemma).

The use of rational agents may not serve the purposes of the modeller. The modeller may wish to know the implications of certain types of behaviour (Gode & Sunder 1993), or to test the robustness of a model to changes in the agents’ behaviour.
Just as rational agents are a simplifying assumption in many analytic models, simple non-rational agents may be simpler to implement in an agent-based model.

3.2 Simple Rule-based Agents

3.2.1 Zero Intelligence Agents

Zero Intelligence Agents draw at random from a uniform distribution over allowable choices. Using agents with zero intelligence means that the effects of market structure can be separated from the effects of agent behaviour. They can be considered a worst-case scenario, compared to the best-case of fully rational agents.

Random behaviour by individuals and firms can lead to downward sloping market demand and upward sloping market supply curves (Becker 1962). Automated agents making random bids and asks (but subject to budget constraints) in a double auction are nearly as efficient as human subjects (Gode & Sunder 1993), although this requires that the slopes of the market demand and supply curve have similar magnitudes (Cliff & Bruten 1996).

3.2.2 Rule-of-thumb Agents

Supplying agents with rules-of-thumb is the simplest way to implement an agent-based model. Even simple agent descriptions can lead to interesting emergent behaviour. If the rules are based on research from behavioural economics, agent behaviour may in fact be more realistic than that of fully rational agents.

3.2.3 Learning

The emergent properties of a population of simple agents may be sufficient to generate the stylised fact that is being studied. Using simple agents can make
it easier to understand an agent-based model. Any additional complexity makes it less likely the model will yield an actual explanation.

However, rule-of-thumb agents have no way to adapt to changes in their economic environment, and so they are subject to the Lucas critique. An agent-based model only provides proper microfoundations if its agents can change their behaviour in response to both exogenous changes imposed by the modeller and endogenous changes emerging from agent behaviour. An agent-based model is more credible if it models learning by its agents.

3.3 Artificially Intelligent Agents

3.3.1 Genetic Algorithms

The genetic algorithm operates on a population where each agent is described by a string of numbers — the agent's genome. This string of numbers determines the agent's behaviour, or phenotype. The agent's phenotype determines the agent's fitness, a number denoting how successful an agent is. The agent's fitness may be calculated directly, or may require simulation with the other agents. Operations — including selection, cross-over, mutation, or election — are performed on the population. Successful behaviours should become more common in the population, unsuccessful behaviours less common (Holland 1992, Axelrod 1997).

Selection The selection operation chooses the strings that will form the next population. At the end of each generation, the fitness scores of the various strings are compared. In proportional selection, each string is used in proportion to its fitness score. The proportion may be adjusted by putting the fitness through a monotonic transformation. The alternative is tournament selection. In this case, each string's ranking determines its use in the next generation. For example, the top half of the population might be kept, and the rest discarded.

Cross-Over The cross-over operation takes two parent strings and forms two child strings from them. A break-point is chosen randomly. One child
is formed from the first parent's genes, up to the break-point, followed by the second parent's genes, from after the break-point. The second child is formed from the remaining material. Figure 3.1 shows a cross-over between SUSPICIOUS PAVLOV\textsuperscript{1} and TIT-FOR-TAT giving SUSPICIOUS TIT-FOR-TAT and PAVLOV.

![Cross-over example.](image)

Figure 3.1: An example of cross-over.

An additional \textit{election} operation (Arifovic 1994) may be performed. The fitness of the children and parents is calculated. The fittest two are kept, and the others are discarded.

\textbf{Mutation} The mutation operation randomly changes the numbers in the string. This is done with small probability (1/1000 for example). Mutation provides greater variability for selection to act on, but reduces the speed of convergence. The mutation rate may begin high and reduce over time to ensure sufficient variability and fast convergence at the appropriate stages of the process.

\textbf{Interpretation}

Selection can be interpreted as imitation. Agents abandon unsuccessful strategies and select another agent's strategy to imitate. Cross-over with election can be viewed as communication, where two agents share the strong

\textsuperscript{1}The \textit{suspicious} version of a nice strategy defects first but plays the same otherwise.
parts of their strategies. In this framework mutation is analogous to innovation (Dawid 1996).

### 3.3.2 Classifier Systems

A classifier system uses a set of rules that link an agent’s observations to parameters determining the agent’s actions (LeBaron 2002). The agent’s observations are encoded into bitstrings. Each rule consists of a potential observation, and a response to that observation. The potential observation may be encoded with *wildcards*, so that it can match several possible observations. If an observation is matched by more than one rule, the more specific rule (the rule with fewer wildcards) is used. If an agent’s observations do not match any rule, a default rule is used.

In figure 3.2 the observation matches the last three rules, but not the first two. The third rule only has two wildcards, compared to three and five for the last two rules. So the agent’s behaviour will be determined by the third rule.

![Figure 3.2: An observation is mapped to a response by a classifier.](image)

<table>
<thead>
<tr>
<th>Observation</th>
<th>Stimulus-Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>11001</td>
<td>→</td>
</tr>
<tr>
<td></td>
<td>10##1--001</td>
</tr>
<tr>
<td></td>
<td>11#1##010</td>
</tr>
<tr>
<td></td>
<td>11#0##011</td>
</tr>
<tr>
<td></td>
<td>11###--100</td>
</tr>
<tr>
<td></td>
<td>######--101</td>
</tr>
</tbody>
</table>

A classifier agent can learn by applying a genetic algorithm to its set of rules, in response to its rules’ performance in the simulation.

This sort of agent is best used when observations of its environment give binary true or false answers. Responding to real valued observations requires setting arbitrary breakpoints - greater or less than \( x \) (LeBaron 2002).
3.3.3 Realism

Learning by agents can be interpreted in two ways. The agents may simply be selecting one of many equilibria. We are interested in which equilibrium is selected, but so long as the correct equilibrium is selected, we are not concerned with the realism of the process. Alternatively, we might be interested in the learning process itself. In this case the learning process needs to be a realistic simplification of the way real people learn (Brenner 2006).

Agent-based models can be very sensitive to the agents' learning. In the Santa Fe Artificial Stock Market, faster learning agents gave more empirically accurate results, whereas slower learning agents would converge to the rational expectations equilibrium (LeBaron 2002).
Conclusions

Orthodox theoretical economics provides us with a language for discussing economics. According to the Sapir-Whorf hypothesis (Kay & Kempton 1984), the language we speak shapes the way we think. Since theoretical economics is essentially a dialect of mathematics, it should be possible to express any idea in this language. But some ideas, especially unorthodox ideas, are more difficult to express than others.

Parsimonious explanations are valued in science. But parsimony is relative to the language used to explain. An explanation is no likelier to be correct because it is easily phrased as a model of representative agents with rational expectations.

Agent-based modelling levels the playing field by allowing models to be analytically intractable.

This gain in expressiveness comes at a cost. The freedom to choose any set of assumptions makes it hard for common standards to emerge. Agent-based models cannot give us mathematically certain results, but must instead be examined using statistical techniques. Agent-based models require intricate implementations which may hide subtle errors. As a new field, agent-based modelling must meet high standards to be accepted into the mainstream.

Where agent-based models reproduce the same results as traditional theory, they are at a disadvantage. We learn nothing new, and we relearn it with less certainty. A new methodology is only useful if it can explain phenomena unexplained by the old. The additional expressiveness of agent-based modelling allows us access to inadequately explored areas of economics.
Agent-based modelling lets us explore the consequences of imperfectly rational agents. It provides explanations of empirical regularities that are the result of a world where people’s rationality is bounded, and which has not necessarily arrived at equilibrium. It highlights the importance of the processes which lead an economy to equilibrium, and which equilibrium will be chosen. Agent-based modelling gives us macroeconomic models with microfoundations that are more than just an empty formalism. It allows us to assume a heterogeneous population of agents, and it allows us to easily aggregate these agents.
REFERENCES


Tesfatsion, L. (2002), 'Agent-Based Computational Economics: Growing Economies From the Bottom Up', *Artificial Life* 8(1), 55–82.