



# The Coconut Model with Heterogeneous Strategies and Learning

Sven Banisch<sup>1</sup> and Eckehard Olbrich<sup>1</sup>

<sup>1</sup>Max Planck Institute for Mathematics in the Sciences, Inselstraße 22, 04103 Leipzig, Germany  
Correspondence should be addressed to [sven.banisch@UniverseCity.de](mailto:sven.banisch@UniverseCity.de)

*Journal of Artificial Societies and Social Simulation* 20(1) 14, 2017  
Doi: 10.18564/jasss.3142 Url: <http://jasss.soc.surrey.ac.uk/20/1/14.html>

Received: 15-04-2016 Accepted: 12-07-2016 Published: 31-01-2017

**Abstract:** In this paper, we develop an agent-based version of the Diamond search equilibrium model – also called Coconut Model. In this model, agents are faced with production decisions that have to be evaluated based on their expectations about the future utility of the produced entity which in turn depends on the global production level via a trading mechanism. While the original dynamical systems formulation assumes an infinite number of homogeneously adapting agents obeying strong rationality conditions, the agent-based setting allows to discuss the effects of heterogeneous and adaptive expectations and enables the analysis of non-equilibrium trajectories. Starting from a baseline implementation that matches the asymptotic behaviour of the original model, we show how agent heterogeneity can be accounted for in the aggregate dynamical equations. We then show that when agents adapt their strategies by a simple temporal difference learning scheme, the system converges to one of the fixed points of the original system. Systematic simulations reveal that this is the only stable equilibrium solution.

**Keywords:** Search Equilibrium Model, Agent-Based Models, Model Alignment, Heterogeneous Agents, Adaptive Agents, Temporal Difference Learning

## Introduction

- 1.1 Aggregate dynamical equations derived from individual-based models are usually exact only if the micro-level dynamics satisfy certain rather restrictive symmetry conditions brought about by homogeneity of the agent properties and interaction rules (Banisch 2016). Micro-level heterogeneity or complex interaction structures often lead to correlations in the dynamics that a macro-level description may not account for. This means that it is impossible to describe the system evolution exactly as a closed set of macroscopic equations. This requires to understand the consequences of heterogeneous agent properties on macroscopic formulations of the system dynamics. In this paper, we present results of this effort for the Diamond search equilibrium model (Diamond 1982; Diamond & Fudenberg 1989) – also known as Coconut Model – which has been introduced in 1982 by the 2010 Nobel laureate Peter Diamond as a model of an economy with trade frictions.
- 1.2 Imagine an island with  $N$  agents that like to eat coconuts. They search for palm trees and harvest a nut from it if the tree is not too tall, meaning that its height does not exceed an individual threshold cost ( $c_{tree} < c_i$ ). However, in order to consume the nut and derive utility  $y$  from this consumption, agents have to find a trading partner, that is, another agent with a nut. Therefore, agents have to base their harvest decision *now* (by setting  $c_i$ ) on their expectation to find a trading partner *in the future*. More precisely, agents are faced with production decisions that depend on their expectations about the future utility of the produced entity, which in turn depends on the global production level via a trading mechanism. For this reason, the Coconut Model is useful not only to incorporate heterogeneity, but also to analyse adaptive agents that – rationally or not – form expectations about the future system state in order to evaluate their decision options.
- 1.3 In original work by (Diamond 1982; Diamond & Fudenberg 1989), this problem of inter-temporal optimization was formulated using dynamic programming principles and in particular the Bellmann equation. This allowed them to get a differential equation (DE) that described the evolution of the cost threshold along an optimality path (where the individual thresholds are all equal  $c_i = c$ ) which was coupled to a second DE describing the

evolution of the number of coconuts in the population. However, knowing the optimal dynamics, that is, the differential equations that an optimal solution has to fulfil, is not sufficient to study problems such as equilibrium selection or stability in general. This is because the optimality conditions do not say anything about the behaviour of the system when it is perturbed into a suboptimal state. On the other hand, the Bellman equation is also at the root of reinforcement learning algorithms and temporal difference (TD) learning, which are known to converge to this optimality under certain conditions (Sutton & Barto 1998). The incorporation of learning in the agent-based version of the Coconut Model and the assessment of its adequacy by comparison to the original solution are a second contribution of this paper.

- 1.4 Almost 40 years ago, (Simon 1978) suggested the need for considering not only the result of rational choice but also the processes that may lead to it. About ten years later, the notion of artificial adaptive agents has been proposed by Holland & Miller (1991), who defined *adaptive* agents by two criteria: (1) agents assign a value (fitness, accumulated reward, etc.) to their actions and (2) they intend to increase this value over time (p. 365). Virtually, all subsequent models with adaptive agents have followed these principles. In genetic algorithms, for instance, an evolutionary mechanism is implemented by which least fit strategies are replaced by fitter ones and genetic operators like recombination and mutation are used to ensure that potential improvements are found even in high-dimensional strategy spaces, (e.g. Holland 1975; Vriend 2000). Another approach, which is popular recently, is strategy switching (e.g. Brock & Hommes 1997; Hommes 2013; Landini et al. 2015). Here, agents constantly evaluate a set of predefined decision heuristics by reinforcement mechanisms and chose the rule that performs best under the current conditions. For a very instructive account, we refer to the excellent introductory chapter to "Behavioral Rationality and Heterogeneous Expectations in Complex Economic Systems" by Hommes (2013).
- 1.5 The TD approach used here differs mildly from these models but fits well with the abstract specification of adaptive behaviour proposed in Holland & Miller (1991). In our case, agents learn the value associated to having and not a coconut in form of the expected future reward and use these values to determine their cost threshold  $c_i$ . This means that agents are forward-looking by trying to anticipate their potential future gains. While checking genetic algorithms or strategy switching methods in the context of the Coconut Model is an interesting issue for future research, in this first paper we would like to derive an agent-based version of the model that is as closely related to the original model as possible. The motivation behind this is well-captured by a quote from Holland & Miller (1991) (p. 366):

"As a minimal requirement, wherever the new approach overlaps classical theory, it must include verified results of that theory in a way reminiscent of the way in which the formalism of general relativity includes the powerful results of classical physics."
- 1.6 In our opinion, this relates to the tradition in ABM research to verify and validate computational models by replication experiments (e.g. Axtell et al. 1996; Hales et al. 2003; Grimm et al. 2006; Wilensky & Rand 2007). The main idea is that the same conceptual model implemented on different machines and possibly using different programming languages should always yield the same behaviour. Our point of view here is that developing scientific standards for ABMs should not be restricted to comparing different computer implementations of the same conceptual model, but it should also aim at aligning or comparing the results of an ABM implementation to analytical formulations of the same processes. This, at least, whenever such descriptions are available.
- 1.7 This is the case for the Coconut Model, and for its rich and sometimes intricate behavior on the one hand and the availability of a set of analytical results on the other the model is well-suited as a testbed for this kind of model comparison. In particular, when it comes to extend a model that is formulated for an idealized homogeneous population so as to incorporate heterogeneity at the agent level, we should make sure that it matches with the theoretical results that are obtained for the idealized case.
- 1.8 Therefore, the main objective of this paper is to derive an agent-based version of the Coconut Model as originally conceived by (Diamond 1982; Diamond & Fudenberg 1989), where the model dynamics have been derived for an idealized infinite and homogeneous population. In Section 3, we will see that this implementation does not lead to the fixed point(s) of the original system. In order to align the ABM to yield the right fixed point behaviour, we have at least two different options which again lead to slight differences in the dynamical behaviour that are not obvious from the DE description.
- 1.9 Once we have such an aligned model, it is possible to study the effects that result from deviating from the idealized setting of homogeneous strategies (see Section 4). Heterogeneity may arise from interaction structure, information available to agents as well as from heterogeneous agent strategies due to learning in finite populations, which will be our main point here. In particular, we will show that heterogeneity can be accounted for in a macroscopic model formulation by the correction term introduced in Olbrich et al. (2012).

**1.10** Section 5 will present agents using TD learning to learn the optimal strategy. As the learning scheme used in this paper can in fact be derived from the Bellman equation used to set up the original model, an agent population that adapts according to this method should converge to the same equilibrium solution in a procedural way. However, as such an approach implements rationality as a process (Simon 1978) it describes the route to optimality and allows to analyse questions related to equilibrium selection and stability. However, by implementing rationality as a process (Simon 1978), this approach describes the route to optimality and allows to analyse questions related to equilibrium selection and stability. However, before proceeding, let us describe the original model more carefully.

## Description of the Original Model

**2.1** Consider an island populated by  $N$  agents. On the island, there are many palm trees and agents wish to consume the coconuts that grow on these trees. The probability that agents find a coco tree is denoted by  $f$  and harvesting a nut from the tree bears a cost  $c_{tree}$  (the metaphor is the height of the tree) that is described by a cumulative distribution  $G(c)$  defining the probability that the cost of a tree  $c_{tree} < c$ . In what follows we consider that the costs that come with a tree are uniformly distributed in the interval  $[c_{min}, c_{max}]$  such that  $G(c) = (c - c_{min}) / (c_{max} - c_{min})$ . Agents cannot store coconuts such that only agents without a nut may climb and harvest a new one. On encountering a tree (with probability  $f$ ), an agent without coconut climbs if the cost  $c_{tree}$  of harvesting the tree does not exceed a strategic cost threshold  $c_i$  (referred to as strategy) above which the agent assumes that harvesting the coconut would not be profitable. In other words, the probability with which an agent without nut will harvest a tree is given by  $fG(c_i)$ . In the original model, the agent strategy is endogenously determined (and then written as  $c_i^t$ ) as described below.

**2.2** In what follows, we denote the state of an agent  $i$  by the tuple  $(s_i, c_i)$  where  $s_i \in \{0, 1\}$  encodes whether agent  $i$  holds a coconut or not and  $c_i$  is the strategy of the agent. We define the macroscopic quantities  $e = \sum_i s_i$  and  $\epsilon = e/N$  with the first being the number of coconuts in the population and the second the ratio of agents having a coconut, respectively. The time evolution in the limit of an infinite population can then be written as a differential equation (DE)

$$\dot{e} = f(1 - \epsilon)G(c^*) - \epsilon^2 \quad (1)$$

where the first term corresponds to the rate at which agents harvest a coconut and the second to trading with  $\epsilon^2$  being the probability of randomly choosing two agents with coconut Diamond (1982).  $c^*$  corresponds to agents' optimal strategy. In the original works that inspired our model, this is homogeneous for the population but endogenously defined (time-dependent).  $G(c^*)$  is the fraction of trees that will be harvested at this cost threshold.

**2.3** A crucial ingredient of the coconut model is that agents are not allowed to directly consume the coconut they harvested. They rather have to search for a trading partner, that is, for another agent that also has a coconut. The idea behind this is that agents have to find buyers for the goods they produce. If an agent that possesses a nut encounters another agent with a nut both of them are supposed to consume instantaneously and derive each a reward of  $y$  from this consumption. In effect, this means that the expected value of climbing a tree depends on the total number of coconuts in the population or, more precisely, on the time agents have to wait until a trading partner will be found.

**2.4** Rational agents are assumed to maximize their expected future utility:

$$V_i(t) = \mathbb{E} \int_t^\infty e^{-\gamma(\tau-t)} r_i(\tau) d\tau$$

where  $r_i(\tau)$  corresponds to the cost of climbing or respectively to the utility  $y$  from consumption of agent  $i$  at time  $\tau$  and  $\gamma$  to the discount factor. A fully rational agent has to find the strategy  $c_i^*$  that maximizes its expected future reward and given that agents cannot consume their coconut instantaneously, this reward depends on expectations about trading chances. This can be formulated as a dynamic programming problem with  $dV_i(t)/dt = -\langle r_i(t) \rangle + \gamma V_i(t)$ . Considering that there are two states (namely,  $s_i = 0$  or  $s_i = 1$ ) there is a value associated to having ( $V_i(s_i = 1, t) := V_i^t(1)$ ) and to not having ( $V_i(s_i = 0, t) := V_i^t(0)$ ) a coconut at time  $t$ . As a rational agent accepts any opportunity that increases expected utility, a necessary condition for an optimal strategy is  $c_i^* = V_i^t(1) - V_i^t(0)$ . Following this reasoning, by assuming homogeneous strategies

$c_i^* = c^*$ , Diamond derives another DE that describes the evolution of the optimal strategy

$$\frac{dc^*}{dt} = \gamma c^* + \epsilon(c^* - y) + f \left[ \int_0^{c^*} (c^* - c) dG(c) \right] \quad (2)$$

- 2.5** The model (1)-(2) is interesting thanks to its a rather rich solution structure and because many macroeconomic models incorporating inter-temporal optimization have a similar form. First, the model may give rise to a market failure equilibrium which arises as a self-fulfilling prophecy where everyone believes that nobody else invests (climbs the coco trees). It has multiple equilibria and it has been shown in Diamond & Fudenberg (1989) that the model could also exhibit cyclic equilibria. This has been interpreted as an abstract model for endogenous business cycles. A complete stability analysis of the system (1) and (2) has been presented by Lux (1992).
- 2.6** The formulation of the model as a two-dimensional system of DEs (1)-(2) (Diamond 1982; Diamond & Fudenberg 1989; Lux 1992) assumes an infinite and homogeneously adapting population. While many aspects of the DE system have been understood by using corresponding analytical tools, some other important aspects of general theoretical interest cannot be addressed within this setting. This includes equilibrium selection but also stability and out-of-equilibrium perturbations, bounded or procedural rationality (Simon 1978) and learning, and finally the influence of micro-level heterogeneity.
- 2.7** In order to address the first of these points – equilibrium selection, Aoki & Shirai (2000) have re-examined the model with finite populations. They related equilibrium selection to finite-size stochastic fluctuations and showed that transitions between different basins of attraction occurred with positive probability. The benefit of implementing Diamond’s model as an ABM is that we can eventually relax the other assumptions as well.

## Homogeneous Strategies

- 3.1** As argued in the introduction, when reimplementing the model in an ABM, we should make sure that the model incorporates the solution(s) of the original model in that the reimplementations of original idealizations leads to the same behaviour. Therefore, let us first look at the dynamics of the model when all agents adopt the same strategy  $c_i = c$ . The fixed point solution of the DE (1) is

$$\epsilon^*(c) = \frac{fG(c)}{2} \left[ \sqrt{1 + \frac{4}{fG(c)}} - 1 \right] \quad (3)$$

which characterizes the equilibrium point  $\epsilon^*$  for a given  $c$ . If we implement the coconut model with homogeneous strategies  $c$  in form of an ABM we can expect that the average level of coconuts in the population approaches and fluctuates around the point  $\epsilon^*(c)$ .

## Model Alignment

### Intuitive Implementation

- 3.2** Intuitively, the Diamond model as set up in the original paper Diamond (1982) could be implemented in the following way (we refer to this «intuitive» implementation as IM):

(0) Random set-up: set strategies  $c_i^0 = c_i$  and initial states  $s_i^0$  according to the desired initial distribution.

(1) Iteration loop:

(a) random choice of an agent  $i$  with probability  $\omega(i) = 1/N$

(b) if  $s_i^t = 0$  climb a coco tree with probability  $fG(c_i)$  and harvest a nut, i.e.,  $s_i^{t+1} = 1$

(c) else if  $s_i^t = 1$

\* choose a second agent  $j$  with probability  $\omega(j) = 1/(N - 1)$  (random matching)

\* if  $s_j^t = 1$  trade such that  $s_i^{t+1} = 0$  and  $s_j^{t+1} = 0$

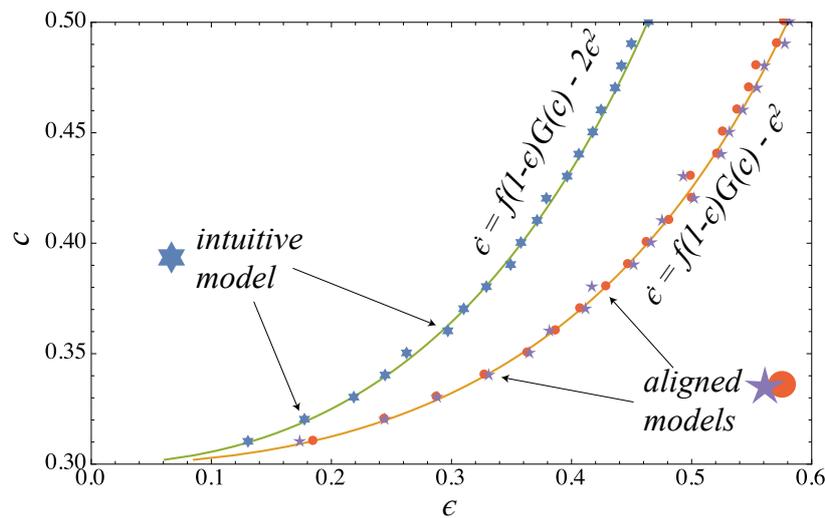


Figure 1: Comparison of the mean level of coconuts (in the stationary regime) of the intuitive implementation of the Diamond model with the original differential equation and an adapted equation that accounts for the fact that two agents may trade but only one agent may climb in one time step. The behaviour of two ABM implementations aligned to the original DE is also shown.

- 3.3** Some short comments on this procedure are in order. For the initialization step (0), we usually define a desired level of coconuts in the initial population ( $\epsilon^0$ ) and a random initial state adhering to this level is obtained by letting each agent have a coconut with probability  $\epsilon^0$ . The initialization of the strategies  $c_i$  will be according to the different scenarios described in the next section but in general the strategies will lie in the interval  $[c_{min}, c_{max}]$ . In this section, the strategies are homogeneous and do not change over time so that all agents have identical strategies during all times. Point (1a) in the iteration process means that at each step we randomly choose one agent from the population. Note that this means that within  $N$  iteration steps some agents may be chosen more than once whereas others might not be chosen at all. For point (1b) the climbing decision with probability  $fG(c_i)$  is evaluated by drawing two random numbers, one for the rate of coco trees  $f$  and another for the cost of the coconut which is uniformly distributed in the interval  $c_{tree} \in [c_{min}, c_{max}]$  for all the experiments we perform throughout this paper. Agents will climb the tree if  $c_i \leq c_{tree}$ . Finally, if agent  $i$  has a nut to trade (1c), a second agent  $j$  is randomly chosen from the remaining agent set which means that there are no specific trading relations between the agents that direct the search for trading partners.
- 3.4** In this section, we compare the outcome of the simulation model as specified above with homogeneous strategies  $c_i = c$  to the behaviour predicted by the DE (1). We look at a small system of only 100 agents in order to see how well the theoretical results obtained for infinite populations approximates the small-scale system. For the comparison the system is initialized with  $\epsilon^0 = 0$  (no coconuts in the population) and run for 4000 steps to reach the stationary regime. Then another 10000 update steps are performed during which we measure the mean level of coconuts  $\epsilon$  in the population. We run a series of such experiments for different strategy values  $c$  in between  $c_{min}$  and  $c_{max}$ . The results of this are shown by the stars points in Fig. 1.
- 3.5** In addition, we show the strategy dependent solution curve  $\epsilon^*(c)$  (3) obtained for the original DE (orange curve). It is clear that for the intuitive implementation the average coconut level is considerably below the DE solution. This is due to the fact that in a single time step, on average, two agents may trade but only one agent may climb a coco tree and harvest a nut which means that the rate  $-\epsilon^2$  actually underestimates the decrease of coconuts by a factor of two. Consequently, we should correct this term for  $-2\epsilon^2$  in order to obtain a DE corresponding to the behaviour of the ABM. That is,

$$\dot{\epsilon} = f(1 - \epsilon)G(c) - 2\epsilon^2 \quad (4)$$

yielding the fixed point curve

$$\epsilon^*(c) = \frac{fG(c)}{4} \left[ \sqrt{1 + \frac{8}{fG(c)}} - 1 \right] \quad (5)$$

which is also shown (green curve) in Fig. 1. It is clear that the simulation of the IM with homogeneous strategies reproduces fairly well the fixed point solution (5) of the adjusted DE.

**3.6** Note that the adjusted DE (4) does not change the behaviour of the system (1)-(2) in a qualitative way. Note also that instead of multiplying  $-\epsilon^2$  by two, we could also rescale  $f$  to  $f/2$  so that the dynamical structure of the model is not really affected. For the experiments in Section 4 we will stick to the intuitive model implementation (IM) as described above and use (5) for the comparison with the DE description.

### Two Ways of Aligning the ABM

**3.7** In order to obtain an ABM that matches well with the original DE description, we have at least two different options. We could first allow two agents to climb at each time step by changing the iteration scheme to

(1) Iteration loop:

- (a) random choice of an agent pair  $(i, j)$  with probability  $\omega(i, j) = 1/N(N - 1)$
- (b<sub>1</sub>) if  $s_i^t = 0$  with probability  $fG(c_i)$  climb tree and harvest, i.e.,  $s_i^{t+1} = 1$
- (b<sub>2</sub>) if  $s_j^t = 0$  with probability  $fG(c_j)$  climb tree and harvest, i.e.,  $s_j^{t+1} = 1$
- (c) else if  $s_i^t s_j^t = 1$  trade such that  $s_i^{t+1} = 0$  and  $s_j^{t+1} = 0$

**3.8** Let us refer to this scheme as aligned model one (AM1). Then, we could decide that an agent  $i$  with a nut ( $s_i^t = 1$ ) trades that nut with a probability proportional to the number  $\epsilon^t$  of coconuts in the population so that at most one coconut will be cleared at a single time step.<sup>1</sup> The iteration (we shall call this scheme AM2) becomes:

(1) Iteration loop:

- (a) random choice of an agent  $i$  with probability  $\omega(i) = 1/N$
- (b) if  $s_i^t = 0$  with probability  $fG(c_i)$  climb tree and harvest, i.e.,  $s_i^{t+1} = 1$
- (c) else trade (consume) with probability  $\epsilon^t$  such that  $s_i^{t+1} = 0$

**3.9** In a sense, this mechanism does not really comply with what we usually do in an ABM. However, it is a way to reproduce the fixed point solution (3) of the original DE, which is shown in Fig. 1 as well. Indeed, the two aligned schemes match very well with the DE solution. AM2 will be used in Section 5 where we incorporate learning dynamics into the model.

### Finite-Size Markov Chain Formulation

**3.10** In addition to the DE and the ABM formulation we shall consider here a third description in form of a Markov chain (MC). For homogeneous populations one can show that an MC description accounting for the transitions between the different numbers of coconuts in the population provides an exact formulation of a finite size ABM with  $N$  agents, Banisch (2016). In the finite size case, the possible  $\epsilon$  are given as the finite set  $\{0/N, 1/N, \dots, N/N\}$  which we shall write as  $\mathcal{Y} = \{0, 1, \dots, e, \dots, N\}$  to simplify notation. We denote by  $e$  the (absolute) number of agents with a coconut (i.e.,  $e = \sum_i s_i$ ).

**3.11** For the intuitive scheme, the transition probabilities for the IM are given by

$$\begin{aligned}
 P(e + 1|e) &= f \frac{(N - e)}{N} G(e) \\
 P(e - 2|e) &= \left( \frac{e(e - 1)}{N(N - 1)} \right) \\
 P(e|e) &= 1 - P(e + 1|e) - P(e - 2|e)
 \end{aligned}
 \tag{6}$$

and zero elsewhere. Note that trading decreases the number of coconuts by two whereas only one coconut can be harvested in a single transition as implemented by the intuitive model. If, instead, we choose  $P(e - 1|e) = \left( \frac{e(e-1)}{N(N-1)} \right)$  and  $P(e - 2|e) = 0$  we obtain a MC representation that is aligned to the AM2 scheme with only one agent consuming a nut on trade.

**3.12** A MC representation can be useful because it provides us with a good understanding about the finite size fluctuations around the equilibrium point. In Fig. 2, this is shown by comparing the observed coconut level in

<sup>1</sup>A decision of that kind seems to be made in Diamond (1982), see footnote 4.

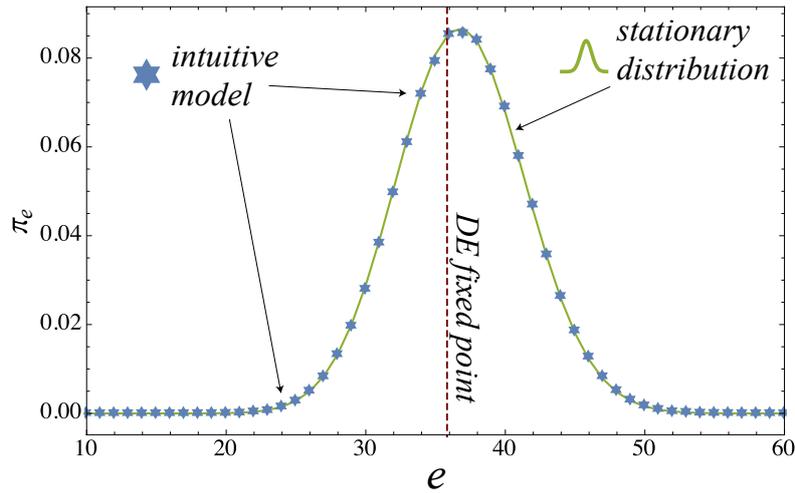


Figure 2: Stationary distribution of the MC for  $N = 100$ ,  $f = 0.8$  and  $c = 0.4$  compared to the frequencies observed in 10 realizations (each 100000 steps) of the simulation model.

the intuitive model to the stationary vector of (6) for the homogeneous strategy  $c = 0.4$ . Note, however, that the peak in the MC stationary vector and the solution of the DE (5) do not match precisely. This is due to the fact that the trading probabilities (with  $\epsilon^2 = (e^2/N^2)$ ) in the mean-field formulation do not explicitly exclude self-trading, whereas this is excluded in the MC formulation as well as in the ABM (and therefore the respective probability is  $(e(e-1)/N(N-1))$ ). While the difference between the two probabilities is not significant if  $N$  is large, there is a notable effect for  $N = 100$ .

## Heterogeneous Strategies

- 4.1** We now relax the assumption of homogeneous strategies by assigning a random individual strategy  $c_i$  to each agent drawn from a certain probability distribution. As before, the strategies are fixed for the entire simulation. The four different scenarios considered in the second part of this section differ with respect to the distribution from which the individual strategies are drawn (in general, in the interval  $[c_{min}, c_{max}]$ ). Before looking at these specific scenarios, however, we derive a correction term that accounts for the effect of strategy heterogeneity.

### Heterogeneity Correction

- 4.2** Namely, if strategies are different we can expect that those agents in the population with a lower  $c_i$  will also climb less often and are therefore less often with a coconut  $s_i = 1$ . That is, there is a correlation between the agent strategy  $c_i$  and the probability  $Pr(s_i = 1)$  that an agent has a coconut.
- 4.3** To account for this we have to consider that the rate  $P(e+1|e)$  that an agent of the population will climb from one time step to the other is given by

$$P(e+1|e) = \frac{f}{N} \sum_{i=1}^N (1-s_i)G(c_i) \quad (7)$$

- 4.4** The equation covering the homogeneous case (6) is satisfied because  $G(c_i) = G(c)$  is equal for all agents and can be taken out of the sum. For heterogeneous strategies, this is not possible but we can come to a similar expression by formulating  $P(e+1|e)$  as the expected value (denoted as  $\langle \rangle$ )

$$\begin{aligned} P(e+1|e) &= f \langle (1-s_i)G(c_i) \rangle \\ &= f (\langle G(c_i) \rangle - \langle s_i G(c_i) \rangle) \\ &= f [\langle G(c_i) \rangle - \langle s_i \rangle \langle G(c_i) \rangle - \sigma[s_i, G(c_i)]] \\ &= f [(1 - \langle s_i \rangle) \langle G(c_i) \rangle - \sigma[s_i, G(c_i)]] , \end{aligned} \quad (8)$$

where  $\sigma[s_i, G(c_i)]$  is the covariance between agent states  $s_i$  and individual climbing probability  $G(c_i)$ . With  $\langle G(c_i) \rangle = G(\bar{c})$  and  $\langle s_i \rangle = e/N$  (the former being true due to the linearity of  $G$  and the latter by definition) this yields

$$P(e+1|e) = f \left[ \underbrace{\frac{N-e}{N} G(\bar{c})}_{\text{mean term}} - \underbrace{\sigma(s_i, G(c_i))}_{\text{covariance correction term}} \right] \quad (9)$$

which corresponds to the original term (6) except for the additional covariance term  $\sigma(s_i, G(c_i))$ .

**4.5** Note that the covariance depends on the individual agent states ( $s_i$  and  $G(c_i)$ ) so that the transition rate is no longer a pure function of the macroscopic average state.

**4.6** This correction term accounting for the correlation between the agent strategy and state should also be included into the infinite-size formulation (4). This is accomplished by Olbrich et al. (2012)

$$\dot{\epsilon} = f[(1-\epsilon)G(\bar{c}) - \sigma(t)] - 2\epsilon^2. \quad (10)$$

Note that covariance term  $\sigma(t)$  in Eq. (10) is here written as a time-dependent parameter because it depends on the actual microscopic agent configuration at time  $t$  to be computed. In other words, this is no longer a closed description in the macroscopic variable  $\epsilon$  as it is coupled to the evolution of the heterogeneity in the system contained in the covariance term.

## Time Evolution of the Heterogeneity Term

**4.7** One way to address this issue is to derive an additional differential equation that describes the time evolution of  $\sigma$  and is coupled to (10). To this purpose, let us denote the probability for an agent  $i$  to be in state  $s_i = 1$  as  $Pr(s_i = 1) := p_i$ . The evolution of this probability is described by

$$\dot{p}_i = f(1-p_i)G(c_i) - \epsilon p_i \quad (11)$$

which is very similar to the macroscopic equation (1) with the first term the probability of climbing and the second that of trading. The correction  $\sigma$  can be written as

$$\sigma = \left( \frac{1}{N} \sum_i G(c_i) p_i \right) - \langle G \rangle \epsilon \quad (12)$$

where  $\langle G \rangle = \frac{1}{N} \sum_i G(c_i)$  ( $= G(\bar{c})$  in our case) is the mean climbing probability. The differentiation is

$$\dot{\sigma} = \left( \frac{1}{N} \sum_i G(c_i) \dot{p}_i \right) - \langle G \rangle \dot{\epsilon} \quad (13)$$

and substituting (11) and (10) yields

$$\dot{\sigma} = \langle G \rangle \epsilon + f(1-\epsilon)(\langle G^2 \rangle - \langle G \rangle \langle G \rangle) + \sigma(f \langle G \rangle - \epsilon) - f\nu(t) \quad (14)$$

where  $\nu(t) = \left( \frac{1}{N} \sum_i G^2(c_i) p_i \right) - \langle G^2 \rangle \epsilon$  is a new higher order covariance term between the second power of  $G^2(c_i)$  and the state probability  $p_i$ .

**4.8** More generally, let us further denote the mean over powers of  $G$  as

$$\langle G^k \rangle := \frac{1}{N} \sum_i G^k(c_i) \quad (15)$$

and write

$$\sigma_k := \left( \frac{1}{N} \sum_i G^k(c_i) p_i \right) - \langle G^k \rangle \epsilon. \quad (16)$$

When we express the dynamical equations for the correction term and then for the higher moment that emerges at each step, we arrive at the following infinite system of equations:

$$\begin{aligned} \dot{\epsilon} &= f(1-\epsilon)\langle G \rangle - f\sigma_1 - 2\epsilon^2 \\ \dot{\sigma}_1 &= \langle G \rangle \epsilon + f(1-\epsilon)(\langle G^2 \rangle - \langle G \rangle \langle G \rangle) - f\sigma_2 - \epsilon\sigma_1 + f\langle G \rangle \sigma_1 \\ \dot{\sigma}_2 &= \langle G \rangle \epsilon + f(1-\epsilon)(\langle G^3 \rangle - \langle G^2 \rangle \langle G \rangle) - f\sigma_3 - \epsilon\sigma_2 + f\langle G^2 \rangle \sigma_1 \\ &\vdots \\ \dot{\sigma}_k &= \langle G \rangle \epsilon + f(1-\epsilon)(\langle G^{k+1} \rangle - \langle G^k \rangle \langle G \rangle) - f\sigma_{k+1} - \epsilon\sigma_k + f\langle G^k \rangle \sigma_1 \end{aligned}$$

4.9 Therefore, we see that a relatively simple form of heterogeneity – namely, a fixed heterogeneous cost threshold – leads to a rather complicated system when we aim at a closed description in terms of macroscopic or average entities. On the other hand, if  $G(c_i)$  is strictly below one, these higher moments  $\sigma_k$  tend to zero as  $k$  increases which allows, under certain conditions, to close the system. However, this goes beyond the scope of this paper and will be addressed in a future work.

### Estimation of Correction Term from Simulations

4.10 Another way to illustrate the usefulness of the correction term derived for the heterogeneous model is to estimate  $\sigma$  from simulations. To do so, we run the ABM with a certain strategy distribution and compute  $\sigma(t) = cov(s_i^t, G(c_i))$  for each step. We then computed the time average over 2000 simulation steps which we may denote by  $\bar{\sigma}$  and replaced the correction terms in (9) and (10) by this average heterogeneity term. The respective transition probability for the Markovian description then becomes

$$P(e + 1|e) = f \left[ \frac{(N - e)}{N} G(c) - \bar{\sigma} \right] \tag{17}$$

and we can compute the corrected stationary vector on that basis. Likewise, from (10) we obtain for the fixed point solution

$$e^*(c^*) = \frac{fG(c^*)}{4} \left[ \sqrt{1 + \frac{8}{fG(c^*)} - \frac{8\bar{\sigma}}{fG(c^*)^2}} - 1 \right]. \tag{18}$$

4.11 These terms are now confronted with simulations initialized according to different distributions of individual strategies. Namely, we consider four cases:

1. strategies uniformly distributed within  $[c_{min}, c_{max}]$ ,
2. there are two different strategies distributed at equal proportion over the population
3. the probability of a strategy  $c_i$  decreases linearly from  $c_{min}$  to  $c_{max}$  and reaches zero at  $c_{max}$
4. probability of a strategy  $c_i$  decreases according to a  $\Gamma$ -distribution with shape 1 and scale  $1/5$

4.12 Please, note that the first two cases are chosen such that the mean climbing probability is  $\langle G \rangle = 0.5$  whereas lower thresholds and therefore a lower average climbing probability are implemented with the latter two.

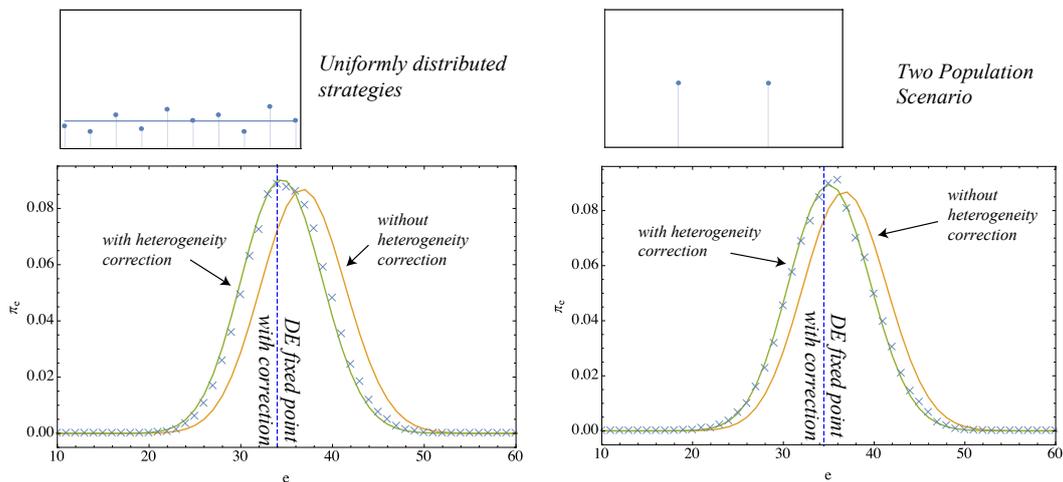


Figure 3: Stationary distribution for  $N = 100$ ,  $f = 0.8$ ,  $c_{min} = 0.3$ ,  $c_{max} = 0.5$  and the comparison to the frequencies observed in a series of 10 realization (10000 steps) of the simulation model.

4.13 The results of applying the heterogeneity correction are shown in Figures 3 and 4. For the four scenarios, the fixed point with correction (18) is shown by the vertical line along with the two stationary distributions of the MCs with and without correction. The crosses in the plot correspond to simulation results measured on 10 simulation runs on 10000 steps. In general, while significant deviations from the simulation results are visible

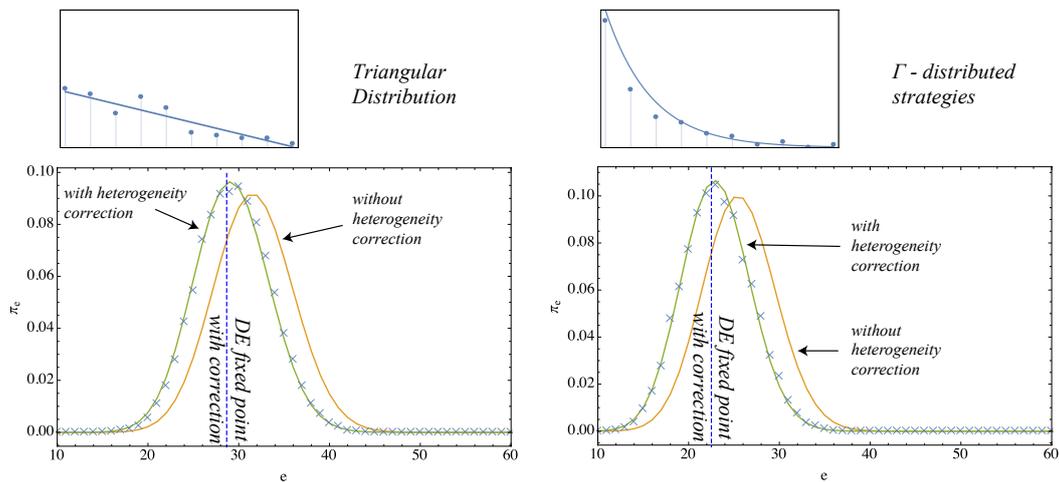


Figure 4: Stationary distribution for  $N = 100$ ,  $f = 0.8$ ,  $c_{min} = 0.3$ ,  $c_{max} = 0.5$  and the comparison to the frequencies observed in a series of 10 realization (10000 steps) of the simulation model.

for the descriptions without correction (homogeneous case), the stationary statistics of the model are well-matched after the covariance correction is applied. This shows that very effective macroscopic formulations of heterogeneous agent systems may be possible by including correction terms that efficiently condense the actual micro-level heterogeneity in the system.

## Adaptive Strategies and Learning

- 5.1 Having gained understanding about how to deal with heterogeneous strategies in the finite-size Coconut Model, we shall now turn to an adaptive mechanism by which the strategies are endogenously set by the agents. As in Section 3, we follow in this implementation the conception of Diamond (1982) as closely as possible. This requires, first, that the threshold  $c_i$  has to trade off the cost of climbing against the expected future gain of earning a coconut from it. In other words, agents have to compare the value (or expected performance if you wish) of having a coconut  $V_i^t(1)$  with the value  $V_i^t(0)$  of staying without a nut. If the difference between the expected gain from harvesting at time  $t$  and that of not harvesting ( $V_i^t(1) - V_i^t(0)$ ) is larger than the cost of the tree  $c_{tree}$ , agents can expect a positive reward from harvesting a nut now. Therefore, following Diamond (1982), it is reasonable to set  $c_i^t = V_i^t(1) - V_i^t(0)$ .
- 5.2 Now, how do agents arrive at reliable estimates of  $V_i^t(1)$  and  $V_i^t(0)$ ? We propose that they do so by a simple temporal difference (TD) learning scheme that has been designed to solve dynamic programming problems as posed in the original model. Note that for single-agent Markov decision processes temporal difference schemes are proven to converge to the optimal value functions Sutton & Barto (1998). In the Coconut Model with agents updated sequentially, it is reasonable to hypothesize that we arrive at accurate estimates of  $V_i^t(1)$  and  $V_i^t(0)$  as well.
- 5.3 However, it is also important to note that the decision problem as posed in Diamond (1982) is not the only possibility to formulate the problem. Indeed, the model assumes that agents condition their action only on their own current state neglecting previous trends and information about other agents the consideration of which might lead to a richer set of solutions. In our case agents do not learn the  $\epsilon$  dependence explicitly, which means that the agents will only learn optimal stationary strategies. The consideration of more complex (and possibly heterogeneous) information sets points certainly to interesting extensions of the model. However, we think that it is useful to first understand the basic model and relate it to the available theoretical results as this will also be needed to understand additional contributions by model extensions.

## Learning the Value Functions by Temporal Differences

- 5.4 The learning algorithm we propose is a very simple value TD scheme. Agents use their own reward signal  $r_i^t$  to update the values of  $V_i^t(s_i^t = 1)$  and  $V_i^t(s_i^t = 0)$  independently from what other agents do. In each iteration,

agents compute the TD error by comparing their current reward plus the discounted expected future gains to their current value estimate

$$\delta_i^{t+1} = \underbrace{r_i^{t+1}}_{\text{reward}} + \underbrace{e^{-\gamma N^{-1}} [s_i^{t+1} V_i^t(1) + (1 - s_i^{t+1}) V_i^t(0)]}_{\text{estimated discounted future value}} - \underbrace{[s_i^t V_i^t(1) + (1 - s_i^t) V_i^t(0)]}_{\text{current estimate}} \quad (19)$$

which becomes

$$\begin{aligned} (0 \rightarrow 0) : \delta_i^{t+1} &= e^{-\gamma N^{-1}} V_i^t(0) - V_i^t(0) \\ (0 \rightarrow 1) : \delta_i^{t+1} &= -c_{tree} + e^{-\gamma N^{-1}} V_i^t(1) - V_i^t(0) \\ (1 \rightarrow 0) : \delta_i^{t+1} &= y + e^{-\gamma N^{-1}} V_i^t(0) - V_i^t(1) \\ (1 \rightarrow 1) : \delta_i^{t+1} &= e^{-\gamma N^{-1}} V_i^t(1) - V_i^t(1) \end{aligned} \quad (20)$$

for different possible transitions ( $s_i^t \rightarrow s_i^{t+1}$ ) of the agents.

**5.5** Note that the discount factor  $\gamma$  as defined for the time continuous DE system is rescaled as  $\gamma_r = e^{-\gamma N^{-1}}$  for the discrete-time setting and in order to account for the finite simulation with asynchronous update in which only one (out of  $N$ ) agents is updated in each time step ( $N^{-1}$ ). The iterative update of the value functions is then given by

$$\begin{aligned} V_i^{t+1}(1) &= V_i^t(1) + \alpha \delta_i^{t+1} s_i^t \\ V_i^{t+1}(0) &= V_i^t(0) + \alpha \delta_i^t (1 - s_i^t) \end{aligned} \quad (21)$$

such that  $V_i(1)$  ( $V_i(0)$ ) is updated only if agent  $i$  has been in state 1 (0) in the preceding time step.

**5.6** The idea behind this scheme and TD learning more generally is that the error between subsequent estimates of the values is reduced as learning proceeds which implies convergence to the true values. The form in which we implement it here is probably the simplest one which does not involve update propagation using eligibility traces usually integrated to speed up the learning process (Sutton & Barto 1998). In other words, agents update only the value associated with their current state  $s_i^t$ . While simplifying the mathematical description (the evolution depends only on the current state) we think this is also plausible as an agent decision heuristic.

**5.7** All in all the model implementation is

- (0) Initialization: set initial values  $V_i^0(1)$ ,  $V_i^0(0)$  and states  $s_i^0$  according to the desired initial distribution. Set initial strategies  $c_i^0 = V_i^0(1) - V_i^0(0)$ .
- (1) Iteration loop I (search and trade):
  - (a) random choice of an agent  $i$  with probability  $\omega(i) = 1/N$
  - (b) if  $s_i = 0$  climb a coco tree with probability  $fG(c_i)$  and harvest a nut, i.e.,  $s_i^{t+1} = 1$
  - (c) else trade (consume) with probability  $\epsilon^t$  such that  $s_i^{t+1} = 0$
- (2) Iteration loop II (learning):
  - (a) compute TD error  $\delta_i^{t+1}$  for all agents with reward signal  $r_j = 0, \forall j \neq i$  and  $r_i$  depending on the action of  $i$  in part (1)
  - (b) update relevant value function by  $V_i^{t+1}(s_i^t) = V_i^t(s_i^t) + \alpha \delta_i^{t+1}$  for all agents
  - (c) update strategy by  $c_i^{t+1} = V_i^{t+1}(1) - V_i^{t+1}(0)$

**5.8** Note that for trading we adopt the mechanism introduced as AM2 in Section 3.6. If not stated otherwise, the simulation experiments that follow have been performed with the following parameters. The interval from which the tree costs are drawn is given by  $c_{max} = 0.5$  and  $c_{min} = 0.3$ . A strategy  $c_i$  larger than  $c_{max}$  hence means that the agent accepts any tree,  $c_i < c_{min}$  that no tree is accepted at all. The rate of tree encounter is  $f = 0.8$  and the utility of coconuts is  $y = 0.6$ . We continue considering a relatively small system of 100 agent and the learning rate is  $\alpha = 0.05$ . The parameter much of the analysis will be concentrated on is the discount rate  $\gamma$  with small values indicating farsighted agents whereas larger values discount future observations more strongly. The system is initialized (if not stated otherwise) with  $\epsilon^0 = 0.5$ ,  $V_i^0(1) = y$  and  $V_i^0(0) = 0$  for all agents such that  $c_i^0 = y > c_{max}$ .

## Simulative Testing of Fixed Point Curves $\dot{c} = 0$ for Different $\gamma$

**5.9** The first part of this paper (exogenously fixed strategies) has shown that the ABM reproduces well the fixed point curve obtained for the coconut dynamics by setting  $\dot{c} = 0$ . Here we want to find out whether the simulation

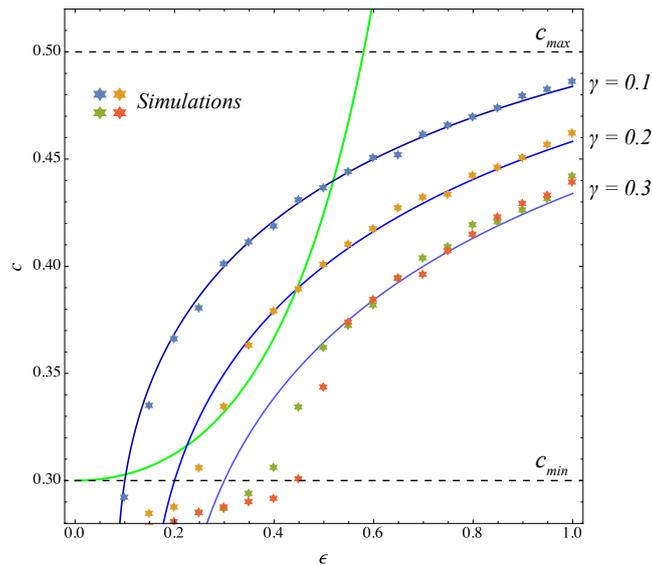


Figure 5: Fixed point curves for  $\gamma = 0.1, 0.2, 0.3$  and comparison to the TD learning dynamics implemented in the agent model.

with TD learning (adaptive strategies) matches with the fixed point behaviour of the strategy dynamics of the original model obtained by setting  $\dot{c} = 0$ . Fig. 5 shows the respective curves for three different  $\gamma = 0.1, 0.2, 0.3$ . Note that the last value  $\gamma = 0.3$  is so large that the  $\epsilon$  and  $c$ -curves do not intersect so that there is actually no fixed point solution.

- 5.10 In order to check these curves in the simulations, we fixed the expected probability of finding a trading partner by  $b(\epsilon) = \epsilon_{fix}$ . Independent of the actual level of coconuts in the population, an agent finds a trading partner with that probability, consumes the coconut and derives a reward of  $y$ . In Fig. 5, for each  $\epsilon_{fix} = 0.0, 0.05, 0.1, \dots, 1.0$  the ABM has been run a single time for 200000 steps and the last system configuration (namely,  $c_i$  at the final step) is used to compute the mean strategy  $c$  which is then plotted against  $\epsilon_{fix}$ .
- 5.11 The model generally matches with the theoretical behaviour, especially when  $\gamma$  is small (farsighted agents). However, for  $\gamma = 0.2$  and  $\gamma = 0.3$  we observe noticeable differences between the simulations and the fixed point curve of the theoretical model. Note that the number of coconuts  $\epsilon$  (which we fix for the trading step) actually also affects the probability with which an agent is chosen to climb and that the actual level of coconuts in the simulation is generally different from  $\epsilon_{fix}$ . This might explain the deviations observed in Fig. 5. Setting up the experiment so that the level of coconuts is constant at  $\epsilon_{fix}$ , however, is not straightforward because an additional artificial state-switching mechanism would have to be included that has no counterpart in the actual model.
- 5.12 However, Fig. 5 indicates that agents which adapt according to TD learning align with the theoretical results in converging to the same (optimal) strategy for a given  $\epsilon$ . The next logical step is now to compare the overall behaviour of the ABM with learning to the theory.

## Overall Fixed Point behaviour with Learning

- 5.13 We first checked the overall convergence behaviour of the ABM as a function of  $\gamma$  and compare it to the fixed point solution of Diamond (1982), see also Lux (1992). There are two interesting questions here:
  1. what happens as we reach the bifurcation value  $\gamma > \gamma^*$  at which the two fixed point curves  $\dot{\epsilon} = 0$  and  $\dot{c} = 0$  cease to intersect?
  2. in the parameter space where they intersect, which of the two solutions is actually realized by the ABM with TD learning?
- 5.14 Fig. 6 answers both questions. First, if  $\gamma$  becomes large, the ABM converges to the state in which agents do not climb any longer, i.e.,  $\epsilon^* = 0$  and  $c^* < c_{min}$ . However, as the close-up view on the right-hand side shows,

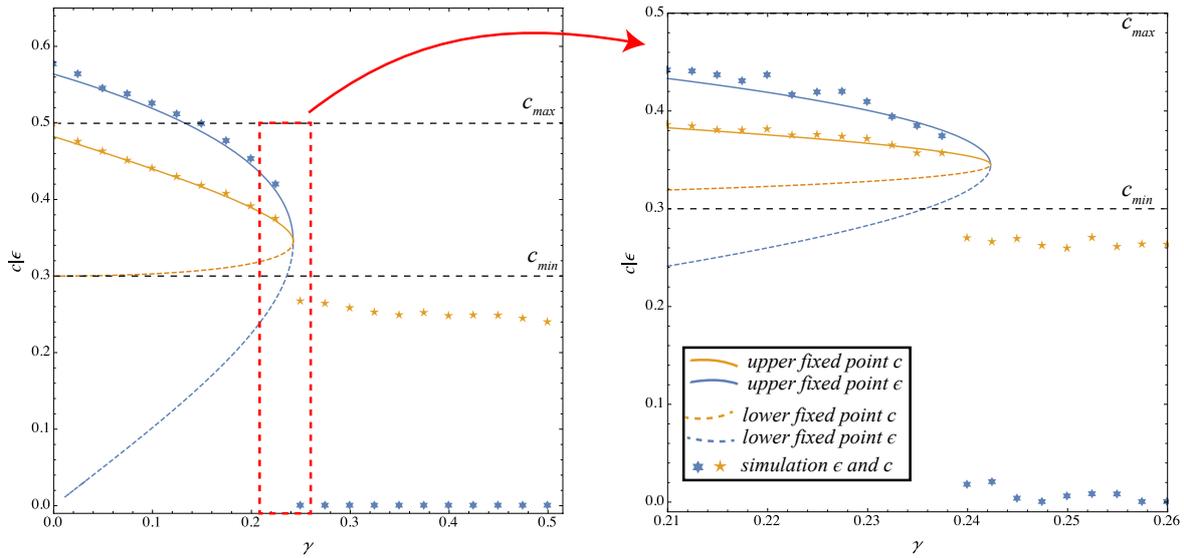


Figure 6: L.h.s.: The fixed point behaviour of the DE system (1) - (2) for  $\gamma \in [0, 0.5]$  is compared to single model realizations (200000 steps) for different  $\gamma$ . The agent model with TD learning scheme converges closely to the theoretic fixed point values. R.h.s.: Close-up view  $\gamma \in [0.21, 0.26]$ . In the experiments the model always reached the upper fixed point.

the bifurcation takes place at slightly lower values of  $\gamma$ . This is probably related to the deviations observed in Fig. 5. In fact, further experiments revealed that the learning rate  $\alpha$  governing the fluctuations of the value estimates plays a decisive role (the larger  $\alpha$ , the smaller the bifurcation point). The larger  $\alpha$  is, the more likely a perturbation takes place on the values of an agent ( $i$ ) which takes  $c_i < c_{min}$  meaning that agent  $i$  does not climb any longer. Besides this small deviation, however, Fig. 6 shows that on the whole the ABM reproduces the theoretical results with considerable accuracy.

**5.15** Regarding the second question – that is, equilibrium selection – it seems that the only stable solution for the simulated dynamics is the upper fixed point, sometimes referred to as «optimistic» solution. We will confirm this below by providing numerical arguments for the instability of the lower fixed point by a series of simulation experiments.

### Instability of the Lower Fixed Point

**5.16** Previous experiments indicated that the lower fixed point derived in the original system is generally unstable under learning dynamics. In this section, we present some further results to confirm this observation by initializing the model at the lower fixed point. We concentrate again on the parameterization used in the previous sections with  $f = 0.8, y = 0.6, c_{min} = 0.3, c_{max} = 0.5$ , climbing costs uniformly distributed in  $[c_{min}, c_{max}]$  and stick to a discount rate  $\gamma = 0.1$ . Fig. 5 shows the respective «pessimistic» equilibrium solution is given by  $\epsilon^* \approx 0.102$  and  $c^* \approx 0.303$  just slightly above  $c_{min}$ . However, for the true fixed point initialization of the simulation model, we have to use the respective values at the fixed point  $V^*(1), V^*(0)$  to initialize  $V^0(1)$  and  $V^0(0)$ . They can be obtained by solving the three-dimensional system from which Eq. (2) has been derived Diamond (1982):

$$\begin{aligned} \dot{\epsilon} &= f(1 - \epsilon)G(c) - \epsilon^2 \\ \dot{V}(1) &= \gamma V(1) + \epsilon(c - y) \\ \dot{V}(0) &= \gamma V(0) - f \int_0^c (c - c')dG(c'). \end{aligned} \quad (22)$$

where  $c = V(1) - V(0)$  as before. With the parameters as given above, we obtain  $V^*(1) \approx 0.303065$  and  $V^*(0) \approx 0.000168$ .

**5.17** Fig. 7 shows the evolution (200000 steps) of the ABM with TD learning for various initial conditions close to the low fixed point (shown by the dashed dark line). There are 100 agents and the learning rate is  $\alpha = 0.025$ .

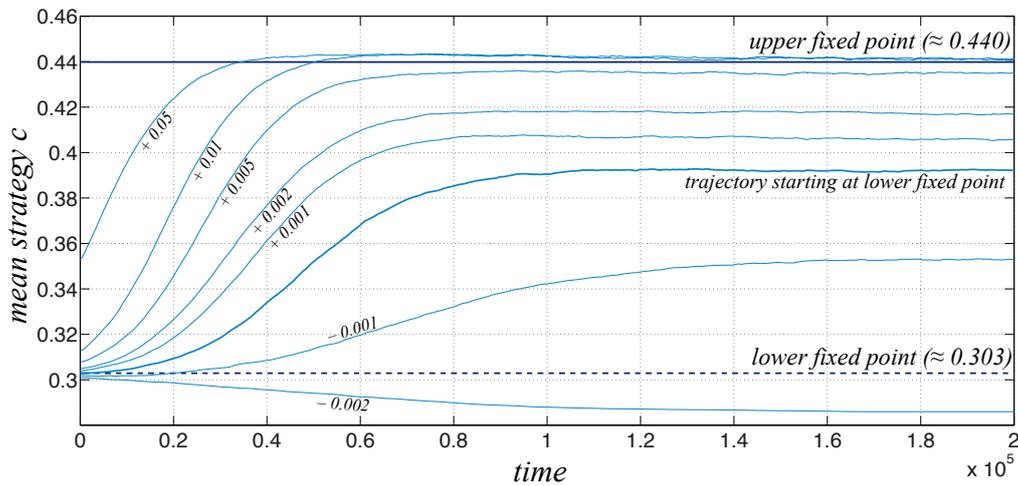


Figure 7: 200000 steps of the ABM with TD learning for various initial conditions close to the low fixed point (shown by the dashed dark line).

Each curve in the plot is an average over 5 simulation runs. We first look at the bold curve corresponding to the trajectory starting at the lower fixed point (see indication in the figure). It is clearly repelled from the low fixed point into the direction of the «optimistic» solution; however, it fails to reach the upper state. The figure also shows trajectories that are initialized with slightly higher  $V^0(1)$  leading to a slight increase of the initial strategy  $c^0$ . The larger this initial deviation, the closer the trajectories converge to the expected »optimistic« strategy  $c^* \approx 0.44$ , but for a deviation smaller than 0.005 the asymptotic behaviour of the learning dynamics does not converge to this value. Interestingly, also simulations initialized slightly below with an initial deviation of  $V^0(1) = V^*(1) - 0.001$  evolve into the direction of the upper solution, however, settling at a still smaller final strategy value.

- 5.18** While this shows that the lower fixed point (or at least a point very close to it) is repelling, this effect of convergence to almost arbitrary states in between the two expected fixed points seems somewhat surprising at the first glance. One possible explanation is that TD learning schemes may converge to suboptimal solutions if agents do not sufficiently explore the space of possibilities. In order to check if this is the reason for the unexpected convergence behaviour when starting with the low initial values, we integrated a form of exploration by adding a small amount of noise to the strategies  $c_i$  each time after the agents computed their new values. This can be seen as if agents are not completely perfect in determining the values and the respective value difference  $V(1) - V(0)$  or just as well that they believe to be able to form estimates only of a given finite precision. In the simulations shown below, a random value uniformly distributed in between  $-0.0015$  and  $+0.0015$  has been added.
- 5.19** Fig. 8 shows this «>exploration» by comparing the time evolution of the model starting exactly in the theoretical low fixed point with respect to different system sizes. In comparison with Fig. 7, where only  $N = 100$  has been considered and the initial deviation from the fixed point has been varied across realizations, we now observe an S-shaped curve that approaches the upper fixed point and stabilizes there with high accuracy. This is observed for all  $N$ .
- 5.20** As the size of the system increases, the initial period in which the system stays close to the lower fixed point increases. However, as shown on the right of Fig. 8, the differences between the learning curves in systems of different size vanish when time is rescaled by the number of agents such that one time step accounts for  $N$  individual updates. This provides further evidence for the instability of the lower fixed point and shows that it is not merely a finite size effect but inherent in the agent system with procedural rationality based on TD learning.
- 5.21** To summarize the analysis performed in this section, we computationally constructed the phase diagrams for the dynamics with learning for a system of 100 agents. Therefore, we performed a suite of systematic computations with samples of initial conditions in the plane spanned by  $\epsilon^0 \in [0, 1]$  and  $c^0 \in [c_{min}, c_{max}]$ . We computed  $26 \times 26$  samples where  $c^0$  was determined by letting  $V^0(0) = 0$  and  $V^0(1) = c^0$  set homogeneously for the entire population. The initial level of coconuts was set randomly with  $\epsilon^0 \in [0, 1]$  such that the probability for each agent to have a coconut in the beginning is  $\epsilon^0$ . For each initial combination, we computed 10 simulations à 10000 steps and compare the initial point  $(\epsilon^0, c^0)$  with the respective outcome after 10000 steps. Fig 9 shows

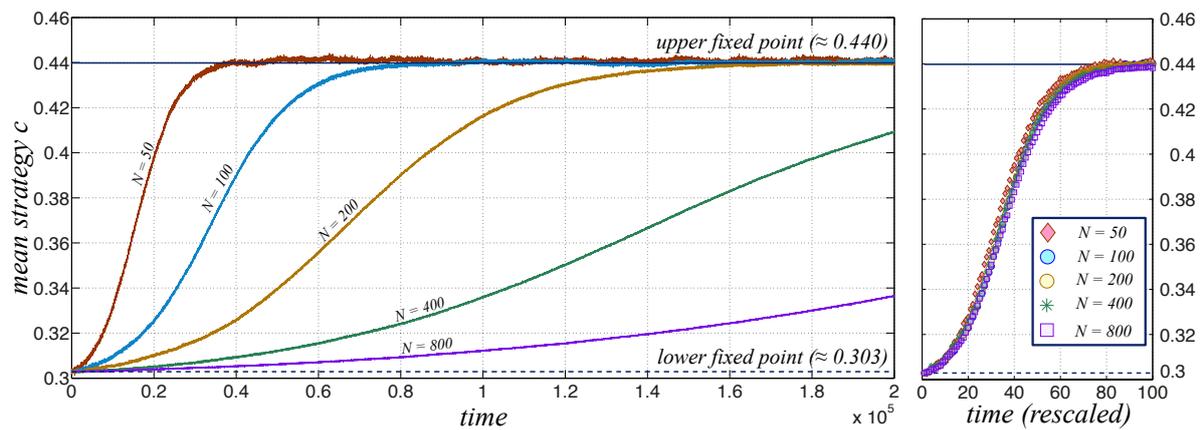


Figure 8: R.h.s.: Time evolution of the system initialized at the low fixed point (shown by the dashed dark line) for different system sizes. L.h.s.: The same learning curve is obtained when rescaling time by the number of agents.

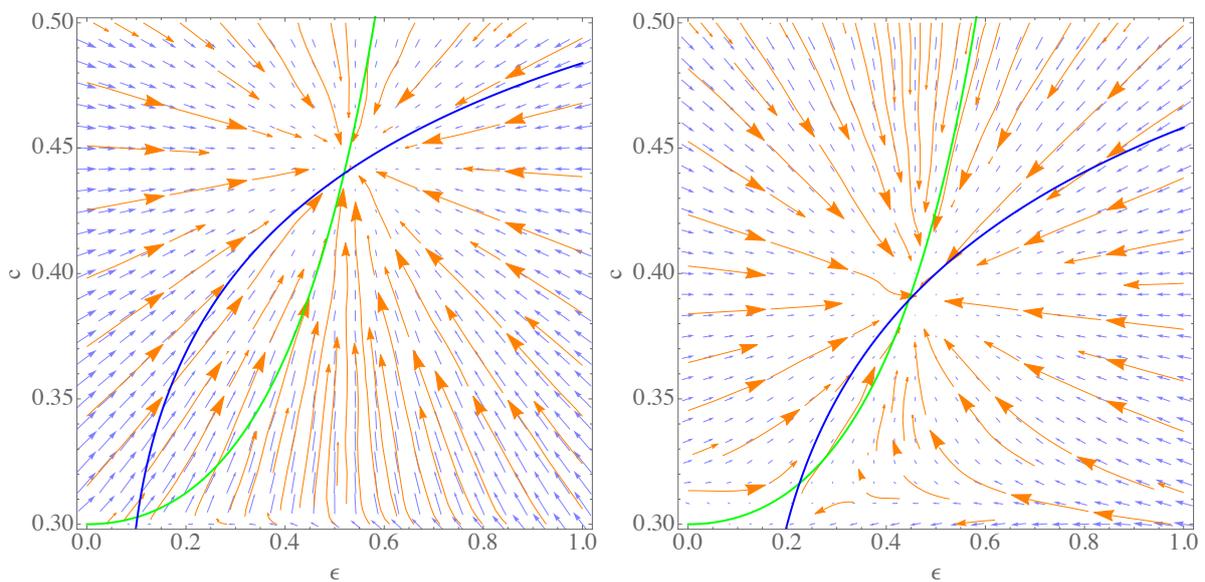


Figure 9: Numerically constructed vector field for the dynamics of the agent model for  $\gamma = 0.1$  (left) and  $\gamma = 0.2$  (right). The fixed point curves of the DE system are also shown.

our results for a discount rate of  $\gamma = 0.1$  (l.h.s.) and  $\gamma = 0.2$  (r.h.s.).

**5.22** The vector field indicates convergence to a state close to the upper fixed point for most of the initial conditions. For  $\gamma = 0.1$ , this is true even for very small initial strategies  $c^0 < 0.303$ . However, note that this point is very close to  $c_{min}$  and that the sampling does not resolve the region around the low fixed point well enough. For  $\gamma = 0.2$ , where the strategy value in the lower fixed point increases to  $c^* \approx 0.316$ , the dynamics around that point become visible. In this case, we observe that initial strategies below this value lead to convergence to  $c < c_{min}$ , that is, to the situation in which agents do not climb any longer (and therefore  $\epsilon = 0$ ). However, if the initial level of coconuts is sufficiently high, the system is capable of reaching the stable upper solution because there is at least one instant of learning that having a nut is profitable ( $V(1)$ ) for agents initially endowed with a nut.

**5.23** Finally, a close-up view on this region is provided in Fig. 10 for  $\gamma = 0.2$ . It shows that the lower fixed point acts as a saddle under the learning dynamics. As indicated above, the exact fixed point values  $\epsilon^*$  and  $c^*$  are slightly different for the DE system and the learning agents model which may be attributed to small differences in the models such as explicit exclusion of self-trading (see Section ) or the discrete learning rate  $\alpha$  (this section).

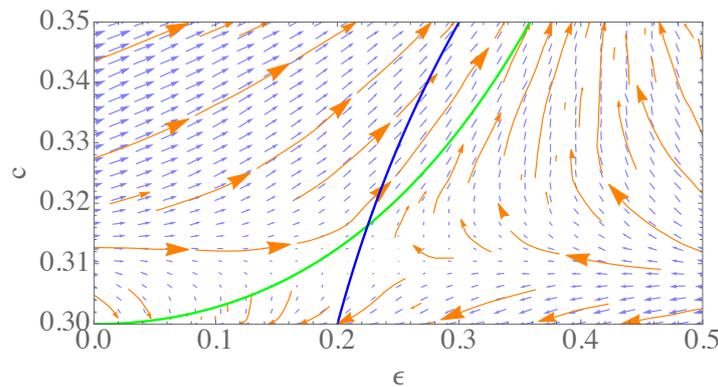


Figure 10: Close-up view on the vector field. The lower fixed point acts as a saddle point under the learning dynamics.

## Outline and Conclusions

- 6.1** This paper makes four contributions. First, it develops a theory-aligned agent-based version of Diamond's coconut model (Diamond 1982). In the model, agents have to make investment decisions to produce some good and find buyers for that good. Step by step, we analysed the effects of single ingredients in that model – from homogeneous to heterogeneous to adaptive strategies – and related them to the qualitative results obtained from the original dynamical systems description. We computationally verified that the overall behaviour of the ABM with adaptive strategies aligns to a considerable accuracy with results of the original model. The main outcome of this exercise is the availability of an abstract baseline model for search equilibrium which allows to analyse more realistic behavioural assumptions such as trade networks, heterogeneous information sets and different forms of bounded rationality but contains the idealized solution as a limiting case.
- 6.2** Secondly, this work provides insight on the effects of micro-level heterogeneity on the macroscopic dynamics and shows how heterogeneous agents can be taken into account in aggregate descriptions. We derived a heterogeneity correction term that condenses the present heterogeneity in the system and show how this term should be coupled to the mean-field equation. These mathematical arguments show that a full characterization of the system with heterogeneity leads to an infinite dimensional system of differential equations the analysis of which will be addressed in the future. In this paper, we have provided support for the suitability of the heterogeneity term by simulation experiments with four different strategy distributions. We envision that the heterogeneity correction may be useful for other models such as opinion dynamics with heterogeneous agent susceptibilities as well.
- 6.3** The third contribution is the introduction of temporal difference (TD) learning as a way to address problems that involve inter-temporal optimization in an agent-based setting. The coconut model serves this purpose so well because the strategy equation in the original paper is based on dynamic programming principles that are also at the root in this branch of reinforcement learning. Due to this common foundation we arrive at an adaptive mechanism for endogenous strategy evolution that converges to one of the theoretical equilibria, but provides, in addition to that, means to understand how (and if) this equilibrium is reached from an out-of-equilibrium situation. Such a characterization of the model dynamics is not possible in the original formulation.
- 6.4** Our fourth contribution is providing new insights into equilibrium selection and stability of equilibria in the coconut model. Under learning dynamics, only the upper «optimistic» solution with a high coconut level (high productivity) is realized. Furthermore, convergence to this equilibrium takes place for a great proportion of out-of-equilibrium states. In fact, the phase diagrams presented at the end of the previous section show that in a system with farsighted agents ( $\gamma = 0.1$ ) the market failure equilibrium (no production, no trade) is reached only if agents are exceedingly pessimistic. If agents are less farsighted ( $\gamma = 0.2$ ), this turning point increases slightly and makes market failure probable if the production level ( $fG(c_i^t)$ ) is currently low for some reason.
- 6.5** However, we do not want to make general claims about the absence of cyclic equilibria in the artificial search and barter economy that the coconut model exemplifies. It is possible – even likely – that a richer behaviour is obtained when agents learn not only based on their own state but take into account information about the global state of the system, trends or the strategy of others. This paper has been a necessary first step to address such question in the future.

## Model Availability

All models described in this paper have been implemented and analysed using Mathematica and Matlab. The Matlab version is available at the OpenABM model archive (Banisch & Olbrich 2016, May 30) (See <https://www.openabm.org/model/5045/version/1>).

## References

- Aoki, M. & Shirai, Y. (2000). A new look at the diamond search model: Stochastic cycles and equilibrium selection in search equilibrium. *Macroeconomic Dynamics*, 4(04), 487–505
- Axtell, R., Axelrod, R., Epstein, J. M. & Cohen, M. D. (1996). Aligning Simulation Models: A Case Study and Results. *Computational and Mathematical Organization Theory*, 1(2), 123–141.
- Banisch, S. (2016). *Markov Chain Aggregation for Agent-Based Models*. Understanding Complex Systems. Springer
- Banisch, S. & Olbrich, E. (2016, May 30). Peter Diamond's Coconut Model (Heterogeneity and Learning). Retrieved from: <https://www.openabm.org/model/5045/version/1>
- Brock, W. A. & Hommes, C. H. (1997). A rational route to randomness. *Econometrica: Journal of the Econometric Society*, (pp. 1059–1095)
- Diamond, P. & Fudenberg, D. (1989). Rational expectations business cycles in search equilibrium. *Journal of political Economy*, (pp. 606–619)
- Diamond, P. A. (1982). Aggregate demand management in search equilibrium. *The Journal of Political Economy*, (pp. 881–894)
- Grimm, V., Berger, U., Bastiansen, F., Eliassen, S., Ginot, V., Giske, J., Goss-Custard, J., Grand, T., Heinz, S. K., Huse, G., Huth, A., Jepsen, J. U., Jorgensen, C., Mooij, W. M., Muller, B., Pe'er, G., Piou, C., Railsback, S. F., Robbins, A. M., Robbins, M. M., Rossmanith, E., Ruger, N., Strand, E., Souissi, S., Stillman, R. A., Vabo, R., Visser, U. & Deangelis, D. L. (2006). A standard protocol for describing individual-based and agent-based models. *Ecological Modelling*, 198, 115–126
- Hales, D., Rouchier, J. & Edmonds, B. (2003). Model-to-Model Analysis. *Journal of Artificial Societies and Social Simulation*, 6(4)
- Holland, J. H. (1975). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. U Michigan Press
- Holland, J. H. & Miller, J. H. (1991). Artificial adaptive agents in economic theory. *The American Economic Review*, 81(2), 365–370
- Hommes, C. (2013). *Behavioral rationality and heterogeneous expectations in complex economic systems*. Cambridge University Press
- Landini, S., Gallegati, M. & Stiglitz, J. E. (2015). Economies with heterogeneous interacting learning agents. *Journal of Economic Interaction and Coordination*, 10(1), 91–118
- Lux, T. (1992). A note on the stability of endogenous cycles in diamond's model of search and barter. *Journal of economics*, 56(2), 185–196
- Olbrich, E., Bertschinger, N. & Wolpert, D. H. (2012). Game theoretic foundations of macroeconomic rational expectation equilibria. In *76. Jahrestagung der DPG und DPG-Frühjahrstagung*
- Simon, H. A. (1978). Rationality as process and as product of thought. *The American economic review*, 68(2), 1–16
- Sutton, R. S. & Barto, A. G. (1998). *Reinforcement learning: An introduction*. MIT press
- Vriend, N. J. (2000). An illustration of the essential difference between individual and social learning, and its consequences for computational analyses. *Journal of economic dynamics and control*, 24(1), 1–19
- Wilensky, U. & Rand, W. (2007). Making Models Match: Replicating an Agent-Based Model. *Journal of Artificial Societies and Social Simulation*, 10(4), 2