



OEFAI-TR-2003-03

Simulation and Validation of an Integrated Markets Model

Brian Sallans

ÖFAI Neural Computation Group

Alexander Pfister

Vienna University of Economics and Business Administration

Alexandros Karatzoglou

Vienna University of Technology

Georg Dorffner

ÖFAI Neural Computation Group

Abstract

The behavior of boundedly rational agents in two interacting markets is investigated. A discrete-time model of coupled financial and consumer markets is described. The integrated model consists of heterogeneous consumers, financial traders, and production firms. The production firms operate in the consumer market, and offer their shares to be traded on the financial market. The model is validated by comparing its output to known empirical properties of real markets. In order to better explore the influence of model parameters on behavior, a novel Markov chain Monte Carlo method is introduced. This method allows for the efficient exploration of large parameter spaces, in order to find which parameter regimes lead to reproduction of empirical phenomena. It is shown that the integrated markets model can reproduce a number of empirical “stylized facts”, including learning-by-doing effects, fundamental price effects, low autocorrelations, volatility clustering, high kurtosis, and volatility-volume correlations.

Simulation and Validation of an Integrated Markets Model

Brian Sallans

ÖFAI Neural Computation Group

Alexander Pfister

Vienna University of Economics and Business Administration

Alexandros Karatzoglou

Vienna University of Technology

Georg Dorffner

ÖFAI Neural Computation Group

1 Introduction

The study of economic phenomena involves not just the domain of economics, but also dynamical systems theory, game theory, the theory of adaptive learning systems, psychology and many others. Beginning with the seminal work of Herbert Simon [1982], there has been a realization that classical economic theory, based on rational equilibria, is limited. In reality, economic agents are bounded both in their knowledge and in their computational abilities. Recent work in simulation-based computational economics has sought to implement boundedly rational economic actors as learning agents, and to study the implications on the resultant economic systems. See Tesfatsion [2002] for a review of agent-based computational economics.

Our goal is to study a discrete-time agent-based economic model which incorporates three types of boundedly rational agents: Production firms, Consumers, and Financial traders. These three agents operate in two coupled markets: a consumer market and a financial equities market. In the consumer market, production firms offer goods for sale, and customers purchase the good. The financial equities market consists of stock traders who can buy and sell shares in the production firms. The two markets are coupled through the production firms, which try to increase shareholder value. They might do this by increasing profits, or by taking actions which directly boost their stock price. Each firm explicitly implements a boundedly-rational agent which learns from experience, and has limited knowledge and computational power.

Models of consumers [Baier and Mazanec 1999], financial traders [Steiglitz, Honig, and Cohen 1995; Arthur, Holland, LeBaron, Palmer, and Taylor 1997; LeBaron, Arthur, and Palmer 1999; Gaunersdorfer 2000], and production firms [Natter, Mild, Feurstein, Dorffner, and Taudes 2001] have been studied previously. Usually, the focus is on a single type of actor (firm, consumer or trader). The other actors are typically modeled as exogenous inputs, or simple random processes. We focus here on the integration of the two markets including explicit models of all three actors. Specifically, we build on the work of Steiglitz, Honig, and Cohen [1995], Arthur, Holland, LeBaron, Palmer, and Taylor [1997], Brock and Hommes [1998], Gaunersdorfer [2000], and Dangl, Dockner, Gaunersdorfer, Pfister, Soegner, and Strobl [2001] in financial market modeling; Baier and Mazanec [1999] in consumer modeling; and Natter, Mild, Feurstein, Dorffner, and Taudes [2001] in production firm modeling. Our approach is to simplify and integrate these previous models, while still retaining their empirical behavior. In addition, the integrated model should address new phenomena that can not be investigated in separate models.

Our ultimate goal is to investigate the mutual influence of the two markets. In particular, because the firms learn based on feedback from the financial market, we can examine the influence of the financial market on production firm behavior. Given rational expectations of firms and financial traders, one might expect that it does not matter whether a firm bases

its actions on its own estimate of future performance, or on its stock price (the shareholders estimate of future performance). This would be the case if firms and traders were fully rational, since both would have the same estimate of the value of a firm and its actions. However, when both are only boundedly rational, then their estimators might be in disagreement. In this case the financial market could have a positive or negative influence on firm performance. The type and degree of influence will depend on how firms and stock traders estimate future performance, and how managers of the firm are compensated.

Before we can use the model to investigate inter-market effects, we have to satisfy ourselves that it behaves in a reasonable way. We validate our computational model by comparing its output to known “stylized facts” in consumer and financial markets. Stylized facts are robust empirical effects that have been identified in a number of examples of real markets. Successful reproduction of empirical phenomena suggests that the dynamical properties of the model are similar to those of the real markets that it tries to emulate. We can then use the model to better understand the underlying dynamics and causes behind the observed effects. For example, by looking at what model parameter settings encourage particular behavior, we can get some insight into the underlying mechanism which causes it.

This article introduces a novel validation technique based on Markov chain Monte Carlo (MCMC) sampling. By using the technique, we can investigate how model parameters influence model behavior, even for large parameter spaces. We can explicitly investigate how different model parameters are correlated, and under what conditions the model reproduces empirical “stylized facts”. This new validation and exploration technique is widely applicable to agent-based simulation models, and is an important contribution of this paper.

The goal of this article is to introduce the integrated markets model, and present validation results suggesting that it is a good combined model of the two markets. After describing the model in detail, we introduce the new model exploration and validation technique based on MCMC sampling. Finally, we describe a number of stylized facts, and show simulation results from the integrated markets model. Using MCMC exploration, we show that the dynamics of competition in the consumer market are an important part of the overall dynamics in the financial market. Similarly, the dynamics of the financial market have an impact on the learning abilities of firms in the consumer market.

2 The Model

The model consists of two markets: a consumer market and a financial equities market. The consumer market simulates the manufacture of a product by *production firms*, and the purchase of the product by *consumers*. The financial market simulates trading of shares. The shares are traded by *financial traders*. The two markets are coupled: The financial traders buy and sell shares in the production firms, and the managers of firms may be concerned with their share price. The traders can use the performance of a firm in the consumer market in order to make trading decisions. Similarly, the production firms can potentially use positioning in product space and pricing to influence the decisions of financial traders (see figure 1).

The simulator runs in discrete time steps. Simulation steps consist of the following operations:

1. Consumers make purchase decisions.
2. Firms receive an income based on their sales and their position in product space.
3. Financial traders make buy/hold/sell decisions. Share prices are set and the market is cleared.
4. Every N_p steps, production firms update their products or pricing policies based on performance in previous iterations.

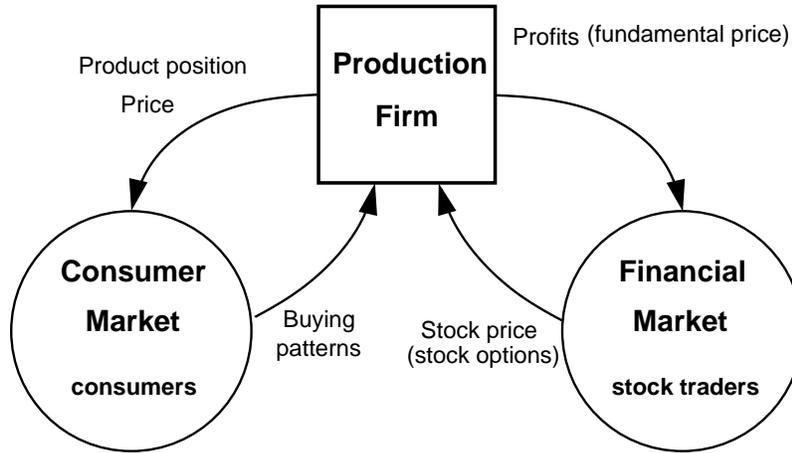


Figure 1: The Integrated Markets Model. Consumers purchase products, and financial traders trade shares. Production firms link the consumer and financial markets, by selling products to consumers and offering their shares in the financial market.

We describe the details of the markets, and how they interact, in the following sections.

3 The Consumer Market

The consumer market consists of firms which manufacture products, and consumers who purchase them. The model is meant to simulate production and purchase of non-durable goods, which the consumers will re-purchase at regular intervals. The product space is represented as a two-dimensional simplex, with product features represented as real numbers in the range $[0,1]$. Each firm manufactures a single product, represented by a point in this two-dimensional space. Consumers have fixed preferences about what kind of product they would like to purchase. Consumer preferences are also represented in the two-dimensional product feature space. There is no distinction between product features and consumer perceptions of those features (see figure 2).

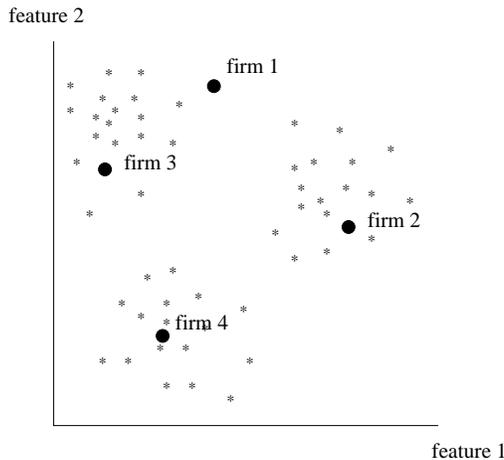


Figure 2: The two-dimensional product space. Consumers have fixed product preferences (denoted by “*”). Firms can position their products (denoted by “.”) in the feature space.

3.1 Firms

The production firms are adaptive learning agents. They adapt to consumer preferences and changing market conditions via a *reinforcement learning* algorithm [Sutton and Barto 1998].

Every N_p iterations of the simulation the firms must examine market conditions and their own performance in the previous iterations, and then modify their product or pricing.

A boundedly rational agent can be subject to several kinds of limitations. We focus here on limits on knowledge, and representational and computational power. How these limitations are implemented is detailed below.

3.1.1 State Description

The firms do not have complete information about the environment in which they operate. In particular, they do not have direct access to consumer preferences. They must infer what the consumers want by observing what they purchase. Purchase information is summarized by performing “k-means” clustering on consumer purchases. The number of cluster centers is fixed at the start of the simulation. The current information about the environment consists of the positions of the cluster centers in feature space, along with some additional information. The information is encoded in a bit-vector of “features”. The features are summarized in Table 1.

Table 1: Features Available to Production Firms.

Feature	Description
Assets	1 if assets increased in the previous iteration, 0 otherwise
Share Price	1 if share price increased in the previous iteration, 0 otherwise
Mean Price	1 if product price is greater than mean price of competitors products, 0 otherwise
Cluster Center 1	A bit-vector that encodes the position of cluster center 1
...	...
Cluster Center N	A bit-vector that encodes the position of cluster center N

The cluster centers are encoded as binary vectors. Each cluster center can be described as a pair of numbers in $[0, 1] \times [0, 1]$. Two corresponding binary vectors are generated by “binning”. Each axis is divided in to K bins. For all of our experiments, K was set to 10. The bit representing the bin occupied by each number is set to 1. All other bits are 0. For example, given 10 bins per axis and a cluster center (0.42, 0.61), the resulting bit vector is (0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0) (see figure 3).

This information gives a summary of the environment at the current time step. Firms make decisions based on the current “state”, which is a finite history of H_s bit vectors. In other words, firms make decisions based on a limited memory of length H_s . This limited history window represents an additional explicit limit on the firm’s knowledge.

3.1.2 Actions

In each iteration the firms can take one of several actions. The actions are summarized in Table 2. The “Do Nothing” and “Increase/Decrease price” actions are self-explanatory. The “random” action is designed to allow the firm to explicitly try “risky” behavior. The “Move product” actions move the features of the product produced by the firm a small distance in a

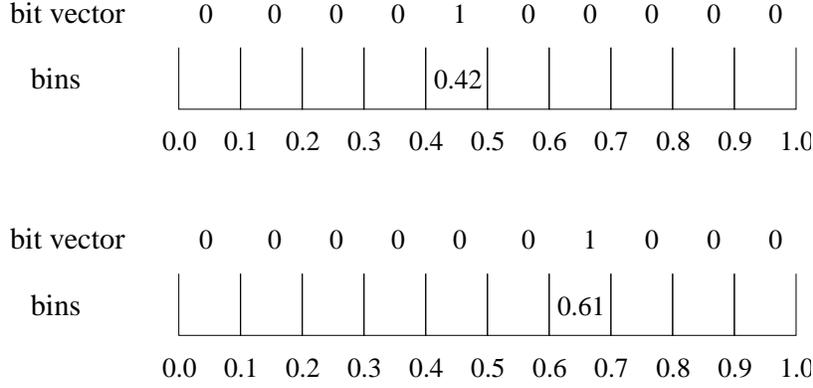


Figure 3: Computing the bit vector representation of a cluster center. In this case the cluster center is located at (0.42, 0.61) and the resulting bit-vector is (000010000000000001000).

Table 2: Actions Available to Production Firms.

Action	Description
Random Action	Take a random action from one of the actions below, drawn from a uniform distribution.
Do Nothing	Take no actions in this iteration.
Increase Price	Increase the product price by 1.
Decrease Price	Decrease the product price by 1.
Move down	Move product in negative Y direction.
Move up	Move product in positive Y direction.
Move left	Move product in negative X direction.
Move right	Move product in positive X direction.
Move Towards Center 1	Move the product features towards cluster center 1.
...	...
Move Towards Center N	Move the product features towards cluster center N.

direction along the chosen axis or towards or away from the chosen cluster center. For example, if the action selected by firm i is “Move Towards Center j ” then the product is modified as follows:

$$b_{i,k,t+1} \leftarrow b_{i,k,t} + \nu (c_{j,k} - b_{i,k,t}) \quad (1)$$

where $k \in \{1, 2\}$ enumerates product features, and $b_{i,k}$ and $c_{j,k}$ are the k^{th} product feature and feature of cluster center j respectively. The update rate $\nu \in (0, 1]$ is a small fixed constant.

3.1.3 Reward Function

A firm’s manager seeks to modify its behavior so as to maximize an external reward signal. The reward signal takes the form of a fixed reward, a variable amount based on the firm’s profitability, and a variable amount due to change in the value of the firms stock. The reward received by firm i at time t is given by:

$$r_{i,t} = S_f + \alpha_\phi \phi_{i,t} + \alpha_p (p_{i,t} - p_{i,t-1}) \quad (2)$$

where S_f is the fixed reward, $\phi_{i,t}$ denotes the profits of firm i at time t , and $p_{i,t}$ denotes the share price of firm i at time t . For all of our experiments, S_f was set to zero for simplicity. The constants α_ϕ and α_p sum to unity. They are fixed at the beginning of the simulation and held constant throughout. They trade off the relative importance of profits and stock price in a firm’s decision-making process. The constant reward signal S_f can be interpreted as a fixed salary paid to the manager of the firm. The profit-based reward can be interpreted as an performance-based bonus given to the manager of the firm, and the stock-based reward as a stock grant or stock option.¹

3.1.4 Utility Function

Given the state history of the simulator for the previous H_s time steps, a production firm makes strategic decisions based on its *utility function*. Utility functions (analogous to “cost-to-go” functions in the control theory literature, and value functions in reinforcement learning) are a basic component of economics, reinforcement learning and optimal control theory [Bertsekas and Tsitsiklis 1996; Sutton and Barto 1998]. Given the “reward signal” or payoff r_t at each time step, the learning agent attempts to act so as to maximize the total expected discounted reward, called expected discounted return, received over the course of the task:²

$$R_t = E \left[\sum_{\tau=t}^{\infty} \gamma^{\tau-t} r_\tau \right]_{\pi} \quad (3)$$

Here $E[\cdot]_{\pi}$ denotes taking expectations with respect to the distribution π , and π is the *policy* of the firm. The policy is a mapping $\pi : \mathcal{S} \rightarrow \Delta^{|\mathcal{A}|}$ from states to distributions over actions. In our case \mathcal{S} is the set of possible state histories, and \mathcal{A} is the set of possible actions taken by a firm. The range of the policy $\Delta^{|\mathcal{A}|}$ is the set of probability distributions over actions in \mathcal{A} . Note that the discount factor γ encodes how “impatient” the firm is to receive reward. It dictates how much future rewards are devalued by the agent. If desired, the discount factor can be set to the rate of inflation or the interest rate in economic simulations, such that the loss of interest on deferred earnings is taken into account by the firm’s manager. In our simulations, the discount factor was found using the Markov chain Monte Carlo validation technique (see section 5).

Given the above definitions, the *action-value function* (or Q-function [Watkins 1989; Watkins and Dayan 1992]) is defined as the expected discounted return conditioned on the current state and action:

$$Q^{\pi}(\mathbf{s}, a) = E \left[\sum_{\tau=t}^{\infty} \gamma^{\tau-t} r_\tau \mid \mathbf{s}_t = \mathbf{s}, a_t = a \right]_{\pi} \quad (4)$$

where \mathbf{s}_t and a_t denote the current state information and action respectively (see tables 1 and 2). The action-value function tells the firm how much total discounted reward it should expect to receive, starting now, if it executes action a in the current state \mathbf{s} , and then follows policy π . In other words, it is the firm’s expected discounted utility (under policy π) conditioned on the current state and the next action. Note that this is not a myopic measure of reward. This utility function takes into account all future (discounted) rewards.

The coding scheme used for world states makes the overall state space quite large. However, in practice, the number of world states observed during a typical simulation is not very

¹The stock based reward changes linearly with stock return. This would be consistent with a limited stock grant, or with a call option where the current price of the underlying stock is significantly above the strike price of the option.

²We will drop the firm index i in this section for clarity. The same reinforcement learning algorithm is used for each firm, with the same parameter settings. Each firm learns its own value function from experience.

large. We can therefore represent the action-value function as a table indexed by unique state histories of length H_s and actions.

3.1.5 Reinforcement Learning

Reinforcement learning provides a way to estimate the action-value function from experience. Based on observations of states, actions and rewards, the learner can build up an estimate of the long term consequences of its actions.

By definition, the action-value function at time $t - 1$ can be related to the action-value function at time t :

$$Q^\pi(\mathbf{s}, a) = E \left[\sum_{\tau=t-1}^{\infty} \gamma^{\tau-t+1} r_\tau | \mathbf{s}_{t-1} = \mathbf{s}, a_{t-1} = a \right]_\pi \quad (5)$$

$$= E [r_{t-1} | \mathbf{s}_{t-1} = \mathbf{s}, a_{t-1} = a]_\pi + \gamma E \left[\sum_{\tau=t}^{\infty} \gamma^{\tau-t} r_\tau | \mathbf{s}_{t-1} = \mathbf{s}, a_{t-1} = a \right]_\pi \quad (6)$$

$$= E [r_{t-1} | \mathbf{s}_{t-1} = \mathbf{s}, a_{t-1} = a]_\pi + \gamma \sum_{\mathbf{s}', a'} P(\mathbf{s}_t = \mathbf{s}' | \mathbf{s}_{t-1} = \mathbf{s}, a_{t-1} = a) \pi(a_t = a' | \mathbf{s}_t = \mathbf{s}') Q^\pi(\mathbf{s}', a') \quad (7)$$

The first line is just the definition of the action-value function. The second line simply unrolls the infinite series one step, and the third line explicitly replaces the second term by the expected action-value function, again by the definition of expectations under policy π . The last line is called the Bellman equations [Bellman 1957]. It is a set of self-consistency equations (one for each state-action pair) relating the utility at time t to the utility at time $t - 1$.

One way to compute the action-value function is to solve the Bellman equations. We use a reinforcement learning technique called SARSA [Rummery and Niranjan 1994; Sutton 1996]. SARSA can be viewed as using a Monte Carlo estimate of the expectations in Eq.(7) in order to iteratively solve the Bellman equations. At time t , the estimate of the action-value function $\widehat{Q}_t(s, a)$ is updated by:

$$\widehat{Q}_t^\pi(\mathbf{s}_{t-1}, a_{t-1}) = (1 - \lambda) \widehat{Q}_{t-1}^\pi(\mathbf{s}_{t-1}, a_{t-1}) + \lambda \left(r_{t-1} + \gamma \widehat{Q}_{t-1}^\pi(\mathbf{s}_t, a_t) \right) \quad (8)$$

where λ is a small learning rate. For all of our experiments, λ was set to 0.1. In words, the utility function is updated as a linear mixture of two parts. The first part is the previous estimate. The second part is a Monte Carlo estimate of future discounted return. It includes a sample of the reward at time $t - 1$ (instead of the expected value of the reward), and the utility for a sampled state and action at time t (instead of the expected utility based on the policy and state transition probability). Finally, the current estimate of the action-value function is used, in place of the true action-value function.

Intuitively this learning rule minimizes the squared error between the action-value function and a bootstrap estimate based on the current reward and the future discounted return, as estimated by the action-value function. Theoretically, this technique has been closely linked to stochastic dynamic programming and Monte Carlo approximation techniques [Bertsekas and Tsitsiklis 1996; Sutton and Barto 1998].

After each action-value function update, a new policy π' is constructed from the updated action-value function estimate:

$$\pi'(a|\mathbf{s}) \leftarrow \frac{\exp(\widehat{Q}^\pi(\mathbf{s}, a)/T)}{\sum_{a'} \exp(\widehat{Q}^\pi(a'|\mathbf{s})/T)} \quad (9)$$

This policy selects actions under a Boltzmann distribution, with better actions selected more frequently. The “temperature” T can be reduced over the course of the simulation. At the

beginning of the simulation, a large temperature makes the policy quite “flat” and encourages exploration. Later, at a low temperature, optimal actions are most often selected.

In theory, using the SARSA algorithm, the action-value function estimate will converge to the optimal action-value function. However, convergence relies on the learner operating in a stationary stochastic Markov environment [Sutton and Barto 1998]. When there is more than one adaptive firm in the environment, the stationarity assumption is violated. Nevertheless, it has been shown that reinforcement learning can be used to approximately solve arbitrary-sum competitive games [Tesauro 1999]. As noted previously, the Markov assumption is also violated, since the state vector of the firm does not include all information necessary for solving the task. For example, explicit consumer preferences and exact product positions are not known by the firm. Such limited-memory reinforcement learning algorithms have been used previously to approximately solve partially observable problems [Jaakkola, Singh, and Jordan 1995]. This approximation can be seen as another source of “boundedness” in a boundedly rational firm. As well as having a limited knowledge base (represented by the partial state history and state aggregation), the firm has a limited model of the capabilities of other firms (they are assumed to have a stationary policy). Thus our firms implement an algorithm designed to iteratively improve their strategies, given the constraints on their knowledge and computational power.

3.2 Consumers

Consumers are defined by their product preference. Each consumer agent is initialized with a random preference in product feature space. During each iteration of the simulation, a consumer must make a product purchase decision. For each available product, the consumer computes a measure of “dissatisfaction” with the product. Dissatisfaction is a function of product price and the distance between the product and the consumer’s preferred product. Consumer i ’s dissatisfaction with product j is given by:

$$\text{DIS}_{i,j} = \alpha_c \frac{D(\beta_i, b_j)}{\max_{b'} D(\beta_i, b')} + (1 - \alpha_c) \frac{\rho_j}{\max_j \rho_j} \quad (10)$$

where ρ_j denotes the price of product j , and α_c trades off the importance of product features and price. For all of our experiments, α_c was set to 0.5. The measure $D(\beta_i, b_j)$ is the Euclidean distance in feature space between the ideal product of customer i and product j :

$$D(\beta_i, b_j) = \frac{(\beta_i - \mathbf{b}_j)^\top \mathbf{W}(\beta_i - \mathbf{b}_j)}{\beta_i^\top \mathbf{W} \beta_i} \quad (11)$$

Here bold-faced letters denote the feature-vector representations of products and preferences. The diagonal matrix \mathbf{W} is common to all consumers and models the relative importance of features in the feature space. In all of our simulations, the matrix \mathbf{W} was initialized to the identity matrix.

Every consumer is also initialized with a different “ceiling” dissatisfaction MAXDIS_i . If all product dissatisfactions are above its ceiling, a consumer will simply make no purchase in that iteration. For all of our simulations, MAXDIS was set to 0.8. Given dissatisfaction ratings for all products, and given the set of products with dissatisfactions below the ceiling in iteration t , consumer i selects from this set the product j with the lowest dissatisfaction rating:

$$j = \arg \min_k \{\text{DIS}_{i,k}\}, k \in \{l : \text{DIS}_{i,l} < \text{MAXDIS}_i\} \quad (12)$$

4 The Financial Market

One important paradigm of modern finance is the efficient market hypothesis. It is assumed, that the actual market price contains all available information. Especially past prices cannot

help in forecasting future price changes. This is in contrast to the empirical trading behavior of many investors. Chartists or technical traders believe that prices may be predicted by extrapolation of trends, technical trading rules or other patterns generated by past prices.

Another paradigm is the rational expectation equilibrium (REE), introduced by Muth [1961]. Agents are expected to be fully informed and to know all equations of the economic model. Perfectly rational agents maximize their utility function and are able to solve complicated optimization problems. This seems to be unrealistically demanding for real-world agents, and therefor bounded rationality models have been proposed. In our model bounded rationality enters via the agent's formation of expectations over future prices and variances. Investors only use publicly available information, i.e. past stock prices and dividends and they do not make systematic mistakes.

Within this section we will present a standard capital market model (see e.g. [Arthur, Holland, LeBaron, Palmer, and Tayler 1997; Brock and Hommes 1998; Dangl, Dockner, Gaunersdorfer, Pfister, Soegner, and Strobl 2001]). Myopic investors maximize their next period's utility subject to a budget restriction. At time t agents invest their wealth in a risky asset with price p_t and in bonds, which are assumed to be risk free. Each agent only trades with a stock of a single firm. Within this section we therefor drop the index i for the firms. There are S stocks paying a dividend d_t . It is assumed that firms pay out all of their profits if positive, therefor each stock gets a proportion of $1/S$ of the profits. The risk free asset is perfectly elastically supplied and earns the risk free and constant interest rate κ . Investors are allowed to change their portfolio in every time step. The wealth of investor m at time $t + 1$ is given by

$$W_{m,t+1} = (1 + \kappa) W_{m,t} + (p_{t+1} + d_{t+1} - (1 + \kappa) p_t) q_{m,t} \quad (13)$$

where $W_{m,t+1}$ is the wealth at time t and $q_{m,t}$ the number of stocks of the risky asset hold at time t . As in Brock and Hommes [1998], Levy and Levy [1996], Chiarella and He [2001], and Chiarella and He [2002] the demand functions of the following models are derived from a Walrasian scenario. This means that each agent is viewed as a price taker (see [Brock and Hommes 1997] and [Grossman 1989])

Denote

p_t :	Price per share of the risky asset at time t	
d_t :	Dividend at time t	
κ :	Risk free rate	
S :	Total number of shares of the risky asset	
M :	Total Number of investors	Let an investor m with wealth W_m
$q_{m,t}$:	Number of shares investor m holds at time t	
$W_{m,t}$:	Wealth of investor m at time t	
ζ_m	Risk aversion of investor m	

maximize his/her utility of the form

$$u(W_m) = -e^{-\zeta_m W_m} \quad (14)$$

with ζ_m as constant absolute risk aversion. Denote by $F_t = \{p_{t-1}, p_{t-2}, \dots, d_{t-1}, d_{t-2}\}$ the information set available at time t .³ Let $E_{m,t}$ and $V_{m,t}$ be the conditional expectation and conditional variance of investor m at time t based on F_t . Then the demand for the risky asset $q_{m,t}$ solves

³Note that at time t price p_t and dividend d_t are not included in the information set F_t

$$\max_{q_{m,t}} \left\{ E_{m,t} [W_{m,t+1}] - \frac{\zeta_m}{2} V_{m,t} [W_{m,t+1}] \right\} \quad (15)$$

i.e.,

$$q_{m,t} = \frac{E_{m,t} [p_{t+1} + d_{t+1}] - p_t (1 + \kappa)}{\zeta_m V_{m,t} [p_{t+1} + d_{t+1}]} \quad (16)$$

Let S be the total number of shares, then the market clearing price p_t is implicitly given by the equilibrium equation

$$S = \sum_{i=1}^M q_{m,t} \quad (17)$$

4.1 Formation of Expectations

It is well known, that expectations play a key role in modeling dynamic phenomena in economics. Heterogeneous expectations are introduced in the following way

$$\begin{aligned} E_{m,t} [p_{t+1} + d_{t+1}] &= I_m (p_{t-1}, \dots, p_{t-h_m}, d_{t-1}, \dots, d_{t-h_m}) \\ V_{m,t} [p_{t+1} + d_{t+1}] &= G_m (p_{t-1}, \dots, p_{t-h_m}, d_{t-1}, \dots, d_{t-h_m}) \end{aligned} \quad (18)$$

As in many other heterogeneous agent models we assume that two kinds of investors exist: fundamentalists and chartists. Additionally, investors have different time horizons h_m which are modeled via the time length agents look back into the past. In our simulations h_m is distributed between 1 and 250. The proportion of fundamentalists and chartists in the market are denoted N_f and N_c , where $N_f + N_c = 1$.

In our model we want to focus on the formation of expectations about prices and not on the formation of expectations about variances. Therefore we assume homogeneous and time independent expectations about the variance $V_{m,t}$. This means that both types of investors, fundamentalists and chartists, determine $V_{m,t}$ in the following way

$$V_{m,t} [p_{t+1} + d_{t+1}] = v \quad (19)$$

where v is a constant. Now we split $E_{m,t} [p_{t+1} + d_{t+1}]$ into $E_{m,t} [p_{t+1}]$ and $E_{m,t} [d_{t+1}]$. First investors form their expectations about the next periods dividend d_{t+1} . Agents take the average of the last h_m dividends as a forecast, i.e.

$$E_{m,t} [d_{t+1}] = \frac{1}{h_m} \sum_{j=1}^{h_m} d_{t-j} \quad (20)$$

Fundamentalists determine their price expectations according to a model based on fundamental information, which in our model are past dividends. The dividends are based on the earnings of the firms in the consumer market. Fundamentalists calculate a fair price and expect that the current price will gradually move towards it with a rate of α_f . A fundamentalist m assumes that the fair price $p_{m,t}^{\text{Fair price}}$ is a linear function of past dividends, i.e.

$$\begin{aligned}
p_{m,t}^{\text{Fair price}} &= H_m(d_{t-1}, \dots, d_{t-h_m}) \\
&= f E_{m,t}[d_{t+1}] + e \\
&= f \frac{1}{h_m} \sum_{j=1}^{h_m} d_{t-j} + e
\end{aligned} \tag{21}$$

where $1/f$ is the fair dividend yield. In our simulation we set $e = 0$ and $f = 50$, which corresponds to a fair dividend yield per period of 2%.⁴ This leads to the following price expectation

$$E_{m,t}[p_{t+1}] = (1 - \alpha_f) p_{t-1} + \alpha_f p_{m,t}^{\text{Fair price}} \tag{22}$$

Chartists use the history of the stock prices in order to form their expectations. They assume that the next period's price change will equal the average price change during the last h_m periods, scaled by a constant α_n .⁵

$$E_{m,t}[p_{t+1}] = p_{t-1} + \alpha_n \left(\frac{p_{t-1} - p_{t-h_m}}{h_m - 1} \right) \tag{23}$$

Note that at time t , p_t is not included in the information set F_t , therefore the investor has to form his/her expectation on the basis of last periods price.

4.2 The Market Clearance Mechanism

The market uses a sealed-bid auction, where the clearance mechanism chooses the price at which trading volume is maximized. The first step is to construct supply and demand curves based on the transaction requests. Example supply and demand curves are shown in figure 4.

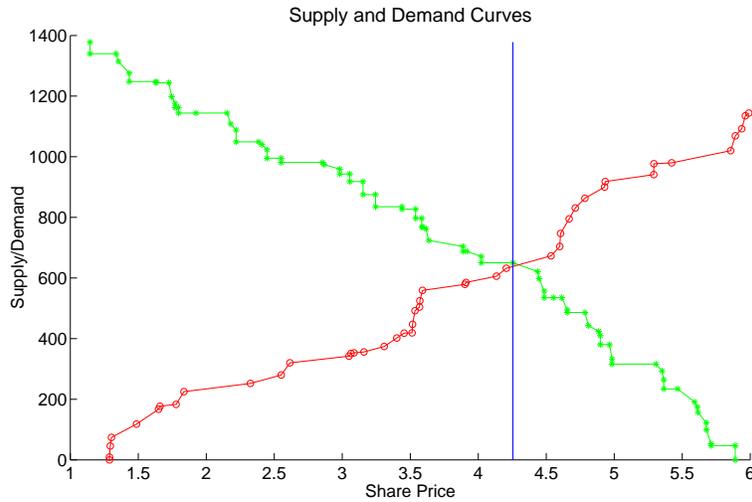


Figure 4: Supply and demand curves. Supply is marked with “O” and increases with price. Demand, marked with “*”, decreases with price. The market price (vertical line) is set to a price which maximizes the volume traded. In this case, the market price is 4.25.

⁴The risk free interest rate κ was set to 1.5%. Because of the fact that holding the stock is riskier, the fair dividend yield should be above the risk free rate.

⁵Note that for chartists $h_m > 1$ for computing an average price change.

Note that there may be a range of prices that would maximize volume. We select the maximum price in this range. If there are buy orders but no sellers then the share price is set to the maximum bid. If there are only sell orders then the price is set to the minimum ask. If there are no orders in a time period, then the price remains unchanged.

Each trader specializes in a single firm, and only buys or sells shares in this firm. Each trader is initialized with a supply of shares in its firm of interest.

4.3 Sequence of Events

Let us have a look at the timing of the events within the financial model. The first step is the formation of expectations. Based on past prices and dividends an investor m forms his/her expectation about the distribution of the next period's price and dividend, i.e. $E_{m,t} [p_{t+1} + d_{t+1}]$ and $V_{m,t} [p_{t+1} + d_{t+1}]$. Plugging the expectations into Equation 16 the agent is able to determine the demand function, which is submitted to the stock market via limit buy orders and limit sell orders.⁶ After the orders of all agents are submitted the stock market calculates this period's equilibrium price p_t , i.e. the price where supply equals demand. At the end of the period the current dividend d_t is announced and becomes public information.

5 Model Validation

One goal of constructing agent-based economic models is to gain some insight into the mechanisms that cause observed market behaviors. Agent-based economic models offer a kind of economic laboratory, in which parameters can be changed, and the results observed. Useful models will reproduce known market behaviors for reasonable parameter settings. Knowing the behavior of the model in different parameter regimes is therefore important both for validating that a model is reasonable, and using the model to understand economic phenomena. However, in complicated models with many parameters, it may be difficult to discover relationships between model parameters, and find regions in parameter space where the model has interesting behavior.

We will validate our model by confirming that it can indeed reproduce empirically observed market behaviors, or "stylized facts". In this section we propose a novel algorithm for exploring the relationship between model parameters and stylized facts. The algorithm is based on Markov chain Monte Carlo (MCMC) sampling. We describe a number of empirical phenomena that have been observed in consumer and financial markets, and give corresponding simulation results. We show that a number of stylized facts within the two markets can be reproduced by our model under reasonable parameter settings. We further show that the behavior of each of the markets is dependent on the dynamics of the other market. In other words, the integrated model is not simply two separate models joined together. The behavior of each market is intimately tied to the parameters and dynamics of the other market. We explore the mechanisms behind some stylized facts by examining correlations between model parameters.

5.1 Model Parameters

Although it has been our intention to keep the model simple, the firm's learning algorithm and trader's decision rules have tuning parameters. Parameter values must be selected before a simulation can be run. These parameters have been introduced in earlier sections describing each of the agents in the model. Using preliminary simulations, some of the parameters were found to have a large influence on the outcome of the simulation, and others were found

⁶A limit order is an instruction stating the maximum price the buyer is willing to pay when buying shares (a limit buy order), or the minimum the seller will accept when selling (a limit sell order).

to be relatively unimportant. All parameters are summarized for convenience in table 3. The “reference” column indicates where in the text the parameter was introduced, and where more details can be found. The “value” column indicates the value used for simulations (see section 5.3). The values of parameters in the first group (above the double line) were found using the Markov chain simulation technique described in the next section. Those in the second group were found to be relatively unimportant. These values were set based on initial trial simulations, and held fixed for all simulation runs.

Table 3: Parameters for Integrated Markets Simulator

Parameter	Description	Range	Value	Reference
α_ϕ	strength of profitability reinforcement	$[0, 1]$	0.47	Eq.(2)
α_p	strength of stock price reinforcement	$[0, 1]$	0.53	Eq.(2)
N	Number of cluster centers	\mathbb{N}	2	section 3.1.1
ν	product update rate	$\mathbb{R} \geq 0$	0.03	Eq.(1)
γ	reinforcement learning discount factor	$[0, 1]$	0.83	Eq.(3)
H_s	History window length for firms	\mathbb{N}	3	section 3.1.1
N_f	Proportion of fundamentalists	$[0, 1]$	0.57	section 4
N_c	Proportion of chartists	$[0, 1]$	0.43	section 4
α_f	Fundamentalist price update rate	$[0, 1]$	0.18	Eq.(22)
α_n	Chartist price update rate	$[0, 1]$	0.36	Eq.(23)
K	Number of bins	\mathbb{N}	10	figure 3
N_p	product update frequency	\mathbb{N}	8	section 2
S_f	base salary	$\mathbb{R} \geq 0$	0	Eq.(2)
λ	reinforcement learning rate	$\mathbb{R} \geq 0$	0.1	Eq.(8)
ϵ	reinforcement learning temperature	$[0, 10]$	$5 \rightarrow 0.2$	Eq.(9)
α_c	Consumer feature/price tradeoff	$[0, 1]$	0.5	Eq.(10)
MAXDIS_i	Maximum dissatisfaction for consumer i	$[0, 1]$	0.8	Eq.(12)
f	inverse fair dividend yield	\mathbb{R}	50	Eq.(21)

We would like to understand the effect of parameters on model behavior. We could “grid” the space of parameters, and then run a large number of repetitions of the simulator, one for each grid point. However, this approach would very quickly become infeasible with more than a few parameters. Ten parameters, each taking on one of ten values, would require 10^{10} runs to cover the grid. Many of these parameter combinations will not be of particular interest.

Instead we would like a way to focus computational power on areas of parameter space that are “interesting”. We will define as interesting areas where a stylized fact is well-reproduced. To this end, we will adapt Markov chain Monte Carlo sampling to do a “directed” random walk through parameter space.

5.2 The Metropolis Algorithm

Consider the problem of evaluating the expected value of some multivariate function with respect to a probability distribution or density. In some cases (such as linear functions and Gaussian distributions) expectations can be computed analytically. In many cases this is not possible. Monte Carlo algorithms allow for the approximate evaluation of expectations in more difficult circumstances. In the following, bold face will denote a vector and subscripts will denote elements of a vector or set: $\mathbf{x} = \langle x_1, \dots, x_J \rangle$. Given a set of multivariate samples $\{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ from a distribution $P(\mathbf{x})$, we can approximate the expected value of a function

$f(\mathbf{x})$ as follows:

$$E[f(\mathbf{x})]_{P(\mathbf{x})} \approx \frac{1}{N} \sum_{i=1}^N f(\mathbf{x}_i) \quad (24)$$

Before this approximation can be employed, we need a set of samples $\{\mathbf{x}_1, \dots, \mathbf{x}_N\} \sim P(\mathbf{x})$. In many cases we do not have a closed-form distribution from which samples can be drawn. The Metropolis algorithm [Metropolis, Rosenbluth, Rosenbluth, Teller, and Teller 1953] is a method for drawing a set of samples from a distribution $P(\mathbf{x})$. Further, we need not have access to $P(\mathbf{x})$, but only need an unnormalized energy function $\Phi(\mathbf{x})$, where:

$$P(\mathbf{x}) = \frac{\exp\{-\Phi(\mathbf{x})\}}{\sum_{\mathbf{x}'} \exp\{-\Phi(\mathbf{x}')\}} \quad (25)$$

Given an initial point \mathbf{x}_0 , the i^{th} step of the Metropolis algorithm operates as follows:

1. Select a dimension k . Select a proposed sample \mathbf{x} from a *proposal distribution* $\text{Pr}_k(\mathbf{x}; \mathbf{x}_{i-1})$. The proposal distribution can be a function of the previous point, and leaves all of the elements of \mathbf{x}_{i-1} unchanged except for the k^{th} element.
2. Set $\mathbf{x}_i \leftarrow \mathbf{x}$ with probability $\min\{1, \exp\{-(\Phi(\mathbf{x}) - \Phi(\mathbf{x}_{i-1}))\}\}$. This is called *accepting* the proposed sample. Set $\mathbf{x}_i \leftarrow \mathbf{x}_{i-1}$ otherwise (*rejecting* the proposed sample).

Note that when a proposal is rejected, the old point is added to the sample in its place. In the algorithm as described above, the proposal distributions should be symmetric. That is, $\forall k \forall \mathbf{x} \forall \mathbf{x}' \text{Pr}_k(\mathbf{x}; \mathbf{x}') = \text{Pr}_k(\mathbf{x}'; \mathbf{x})$.

In the limit, the sequence of samples will converge to a unique stationary distribution with marginal distribution $P(\mathbf{x})$. Thus the set of samples can be used for the approximation in Eq.(24). In practice, the speed of convergence of the chain to the stationary distribution will depend on the dimensionality of \mathbf{x} , the energy function of interest and the proposal distribution. Assessing convergence can be problematic. If a non-convergent set of samples is used, then the estimate will be biased. The algorithm can also be extended to include non-symmetric proposal distributions.

5.3 Markov Chain Model Exploration

In our application, we do not want to evaluate expectations of a function. Instead, we want to find settings for model parameters that reproduce stylized facts. The Metropolis sampler has the following property: Samples are more likely to be drawn from low-energy areas.

Given a stylized fact, we can define an energy function such that low energy corresponds to good reproduction of the fact. Then, we implement a Metropolis sampler using this energy function. In the limit, parameter samples are drawn according to the normalized probability distribution defined by the energy function. In practice, we will not generate Markov chains which are sufficiently long to reach the equilibrium distribution. But even without theoretical guarantees on the distribution of sampled parameters, the sampler can find good model parameter settings, and reveal interesting correlations between model parameters. The Metropolis sampler acts as a “directed” random walk through parameter space, avoiding high energy areas.

We have constructed energy functions for several stylized facts including: learning-by-doing in the consumer market, low autocorrelations in stock returns, high kurtosis in marginal returns, and volatility clustering. The sampler operated over the parameters in the first group of Table 3. We used symmetric Gaussian proposal distributions over real-valued parameters, and uniform distributions over discrete parameters. It was assumed that the energy function

took on a value of $+\infty$ wherever parameters fell outside of their valid range, ensuring that such values would be rejected by the sampler. One thousand samples were drawn using the Metropolis sampler. While this is too short to allow for convergence, we can still examine the sample set to identify regions where stylized facts are well-reproduced, and look for significant correlations between parameters.

As it turns out, two of the four Markov chain experiments were uninteresting. These were the runs trying to achieve high kurtosis in the stock market returns, and getting high autocorrelations in the absolute stock returns. The simulated stock market had both of these features for almost all parameter values, and there were no interesting correlations or relationships between parameters for these energy functions. The only parameters of interest were the proportion of fundamentalists and chartists. If the number of chartists fell below 20%, the returns looked Gaussian. This suggests that high kurtosis and volatility clustering are very robust features of the artificial stock market, and are driven by the interaction between fundamentalists and chartists.

In the sections below, we show the results for two energy functions: The “learning-by-doing” effect, and low-autocorrelations in the stock market returns.

5.3.1 Learning by Doing

The “learning-by-doing” effect [Argote 1999] encapsulates the idea that firms gain knowledge and optimize their behavior over the course of performing a task. Empirically, costs go down, and efficiency and profits go up as a function of the number of units of a particular product produced. Our model explicitly includes learning by doing in the production firm. As the firm produces its product, it learns what sells in the marketplace and at what price. This results in an increase in profits over time. Note that this is very different from models which include a “learning” component in populations of agents, implemented as an evolutionary algorithm. Our individual firms learn over the course of the task.

We investigated which parameter settings influence the learning-by-doing effect using our adapted Metropolis algorithm. The energy function was the negative profits:

$$E = \frac{1}{Z_p} \sum_i \sum_{t=2}^T -(\phi_{i,t}) \quad (26)$$

where i indexes the firms, and Z_p is simply a scaling factor designed to bring the energies into a reasonable range (set to 10000 for our simulations).

We found that the learning effect was quite robust to parameter settings. In general, firms learned to perform well in the market place for almost all parameter settings (see figure 5).

There was a significant negative correlation between the proportion of fundamentalist traders N_f in the simulation, and the adaption rate α_f of the fundamentalists (see figure 6).⁷⁸ Note what this implies: Two parameters of the stock market are correlated when trying to maximize a quantity from the consumer market (profits). This suggests that the feedback mechanism from the stock market to the production firms (via stock price) is having an influence on the behavior of the firms in the consumer market, and that some intermediate behavior of the financial market is optimal from the point of view of firm learning. This may be because of an exploration/exploitation tradeoff: A certain amount of noise or uncertainty in the financial market could help the firms avoid shallow local minima and prompt them to find products

⁷The density plots were generated using the kernel density estimator for Matlab provided by C.C. Beardah at <http://science.ntu.ac.uk/msor/ccb/densest.html> [Beardah and Baxter 1996].

⁸Significance was measured in the following way: First, the sequence of parameter values was subsampled such that autocorrelations were insignificant. Given this independent sample, the correlations between parameters could be measured, and significance levels found.

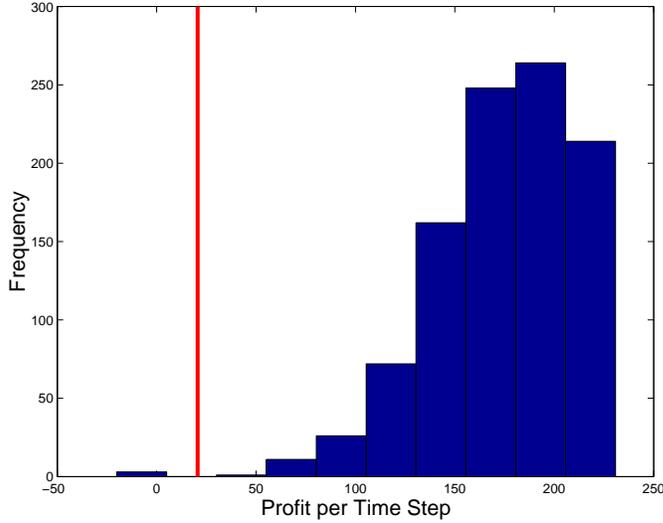


Figure 5: The “Learning by Doing” effect was robust across almost all parameter settings. The bar graph shows the per-time-step profits of the firms sampled by the Metropolis algorithm. The vertical line shows mean profits achieved by a randomly-behaving firm. The learning firms do better than a randomly acting firm for nearly all parameter settings.

and prices that are clearly winning in the consumer market. Too much noise can inhibit learning. The proportion and adaptation rate of the fundamentalists influences the volatility of the financial market, and therefore the noise in a firm’s reward signal.

5.3.2 Low Predictability and Volatility Clustering

A fundamental feature of financial markets is that they are not easily predictable. The perfect market hypothesis claims that new information is immediately factored into prices, so that the price at any given time reflects all prior knowledge. Under this assumption, it is in principle impossible to predict market movements. In practice, it has been found that many financial return series have insignificant autocorrelations. Unlike most artificial stock markets, our model does not include any extrinsic noise (such as randomized trading strategies [Gaunderdorfer 2000; Raberto, Cincotti, Focardi, and Marchesi 2001], or a randomized dividend process [Arthur, Holland, LeBaron, Palmer, and Taylor 1997]). Interestingly, the autocorrelations are nevertheless very low. This is due to a combination of heterogenous trading strategies in the market, and the difficulty in predicting profits in the consumer market.

Unlike price movements, price volatility is highly autocorrelated. Empirically, market volatility is known to come in “clusters”. That is, periods of high or low volatility tend to follow one another. This is the basis of conditional heteroskedasticity models, which predict volatility in the next time step based on volatility in previous time steps. In our model technical traders will tend to adjust their market position during large price movements. This will in turn cause greater price movements. Similarly, when the price is near the fundamental price, fundamentalists are satisfied and hold their stock. This in turn stabilizes prices, and causes the chartists to hold their stock as well.

We investigated which parameter settings lead to low autocorrelations in the returns of the artificial stock market. The energy function used was the squared error between the actual autocorrelations in the returns, and an idealized set of autocorrelations:

$$E = \sum_{i=1}^A (v_i - v_i^*)^2 \quad (27)$$

where v_i denotes the autocorrelation at lag i , and v_i^* is the idealized autocorrelation. We used $A = 5$, and $v^* = \{-0.05, 0.0, 0.0, 0.0, 0.0\}$. That is, a slight negative autocorrelation at the first lag, and zero autocorrelation thereafter.

After sampling with this energy function, we found significant correlations between some sampled production firm parameters. This is particularly interesting, because it indicates that

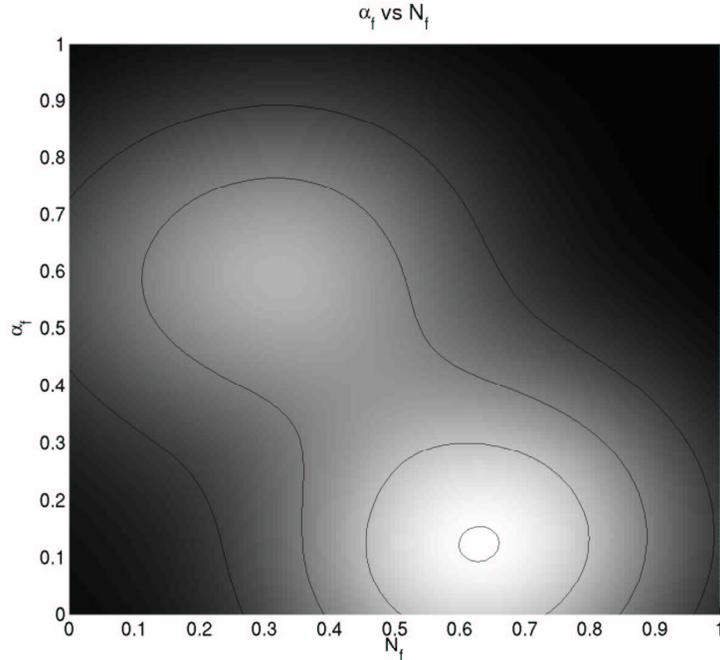


Figure 6: Negative correlation between adaption rate of fundamentalist traders α_f and proportion of fundamentalist traders N_f . In order for firms to maximize their profits, there is an optimal influence of fundamentalists on the stock market. When more fundamentalists are in the market, their per-time-step influence is decreased. The plot shows the density of samples at various parameter values for the Markov chain Monte Carlo simulation.

the statistical properties of the stock returns are substantially affected by the dynamics in the consumer market. Specifically, there is a significant negative correlation between the firm's "history depth" parameter H_s , and the weighting placed by the firm on profits α_ϕ (at the 95% confidence level). That is, in order to get low autocorrelations in the stock returns, it is best to have either a short history depth, or to place the most weight on improving the stock price (at the expense of profits) (see figure 7).

This is likely related to how hard it is for the firms to learn to do well in the market. Recall, the sampler is trying to find parameter values for which the stock returns have low autocorrelations. That is, the sampler prefers stock prices that are unpredictable. If the firms do very well or very poorly, then their fundamental price is predictable, and the stock returns have higher autocorrelation. There is a regime in which firms have variable performance. The amount of information available to firms (the history length H_s) and the kind of information available (profits or stock price) appear to trade-off in determining firm performance.

5.4 Ideal Parameters

We identified a set of parameter settings for which all of the stylized facts were well reproduced (see figure 8 and column "Value" in Table 3). We did this by intersecting the histograms of parameter values from the MCMC simulation runs, and finding common parameter settings. Since nearly all parameter settings gave good kurtosis and volatility clustering behavior, these have been omitted from the figure for clarity. After identifying a set of parameter settings for which all of the stylized facts were well reproduced we ran 20 repetitions of the simulation at these ideal parameter settings. The simulation consisted of two competing firms, 50 stock traders, and 200 consumers.

The "ideal" parameter values are reasonable. There are no parameters which must take on

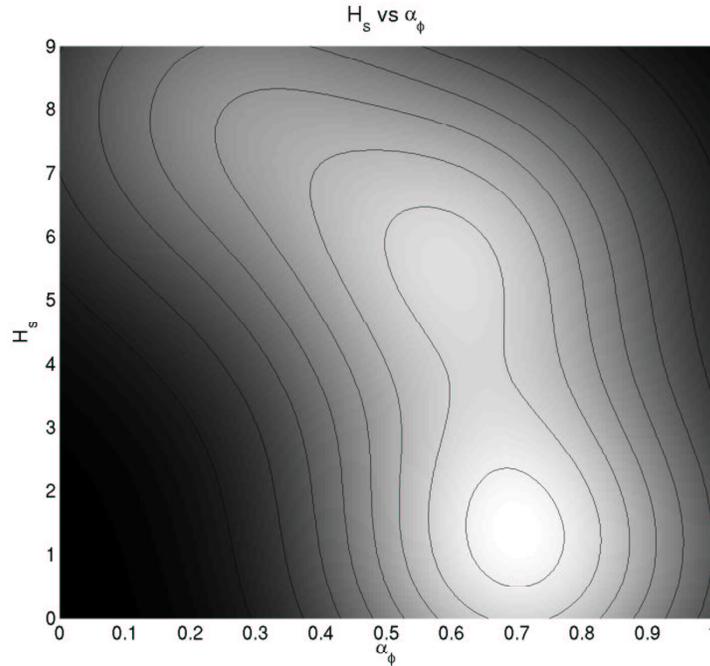


Figure 7: Negative correlation between history depth and importance of profits. “History depth” H_s is the number of past states available to the firm when making decisions. The parameter α_ϕ is the proportion of firm reward that comes from profits. In order to get low autocorrelations in the stock price returns, either history depth should be high and profit importance low, or profit importance should be high and history depth low.

extreme or unlikely values in order to get good simulation behavior. We discuss each of the parameter histograms below:

firm learning rate (ν) : The firm learning rate ν has the least area of overlap between MCMC runs. In order to get good learning, the learning rate should be low. In order to get low autocorrelations, the learning rate should be high. We chose an intermediate learning rate which seemed to work well enough.

Number of cluster centers (N) : Again, there is tension between firm “learning-by-doing” and making the stock market process more random. The firms learned best when they did not have to deal with too much information (fewer clusters). Again, an intermediate value was used.

Profit strength (α_ϕ) : Interestingly, both energy functions were quite insensitive to whether firms learned based on their profits or their stock price, although both had peaks for intermediate values of α_ϕ . Again, an intermediate value was chosen.

Firm discounting (γ) : Firms learned best with either a low or high discounting value. That is, they should either focused on the near term, or the long term, but not the intermediate term. The stock market, in contrast, had lower autocorrelations when firms took the long view.

History depth (H_s) : Firms learned best with shorter histories, and the stock market autocorrelations were better when the firms used longer histories. This is the same effect as seen with the number of cluster centers: Firms learn better when their information is somewhat compressed. The autocorrelations in the financial market seem to be lower when the firms have too much information, and therefore do not learn as well.

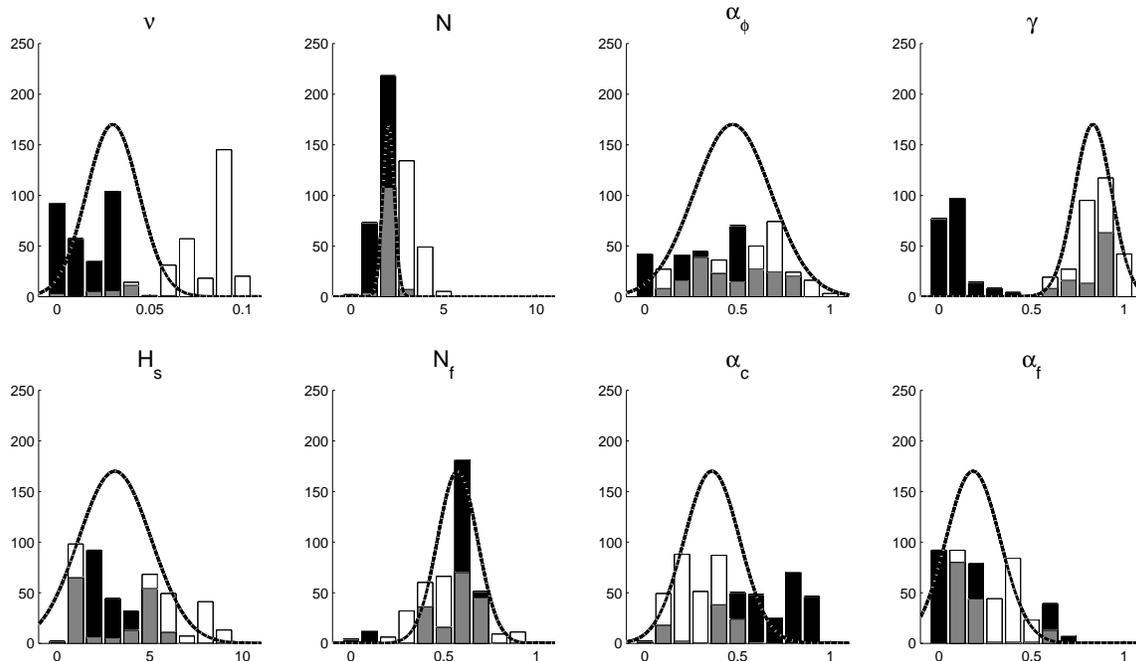


Figure 8: Histograms of parameter values from Markov chain Monte Carlo sampling. The plot for each parameter shows three histograms: black for the “learning-by-doing” energy function (section 5.3.1), white for the low-autocorrelations energy function (section 5.3.2), and gray for the intersection of the other two. Each histograms includes the top 30% of samples from the MCMC sampler, ranked by the negative energy. The curve shows a Gaussian fit to this intersection. The “ideal” parameters were taken to be the means of these best-fit Gaussians.

Proportion of Fundamentalists (N_f) : The market had the best autocorrelation structure for intermediate numbers of fundamentalists. The firms seemed to have a strong preference for a particular value (around $N_f=0.6$). This is interesting, because it indicates that the stock market is not just “noise” in the firms learning process. Certain market dynamics make it easier for the firms to learn. This might be because of an “exploration-exploitation” trade-off inherent in firm learning. The firms must discover what strategies work well, and then exploit them. Intermediate noise in the financial market could allow them to escape local behavioral optima, and find better strategies.

Chartist adaptation rate (α_c) : One might expect that firms learn better, and the market is less random, when the adaptation rate of the chartists is lower. In fact, the opposite seems to be the case. Again, it could be that certain levels of noise in the market are optimal for firm learning.

Fundamentalist adaptation rate (α_f) : Similarly to above, these values seem counterintuitive: Firms learn better when the fundamentalists adapt more slowly, and the market appears more random when the fundamentalists adapt faster.

The following sections show stylized facts reproduced by simulation runs at the ideal parameter settings.

5.4.1 The Learning Effect

Figure 9 shows simulated profits as a function of time, across the 20 simulation runs at the parameter settings specified in Table 3. Median profits increase as a function of time, indicating

that firms learn to identify good product positions and prices. The increase is significant at the 5% level, as tested with a Wilcoxon signed rank test.

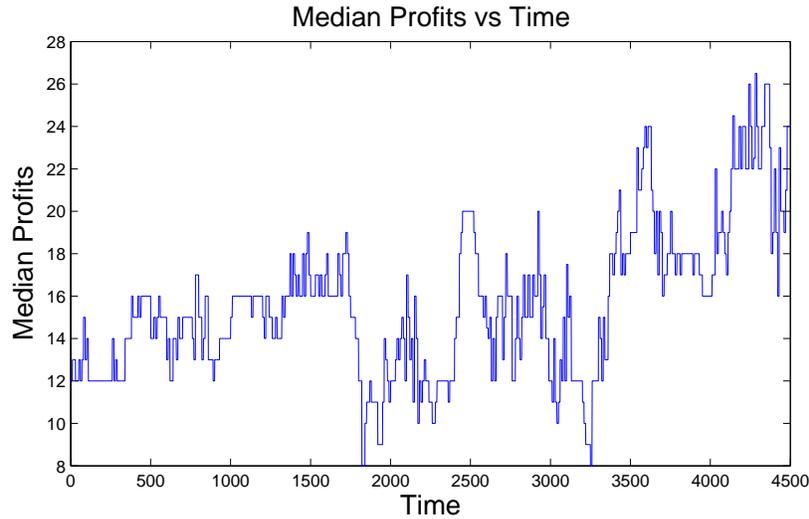


Figure 9: “Learning by Doing” in the consumer market. The plot shows median profits as a function of time, across 20 simulation runs. The longer a firm spends in the market, the higher its profits.

5.4.2 Autocorrelations of Returns

Figure 10 shows autocorrelations of returns and absolute returns for the artificial market. The autocorrelations were computed for the last 2000 periods of each runs, and averaged over 20 runs. For these plots, p_t is stock price at time t , and returns at time t are defined as $ret_t = \log(p_t/p_{t-1})$. There are small negative autocorrelations in the first few lags, followed by zero autocorrelations. The kurtosis of the market returns was quite high at 57.8. The error bars show 95% confidence bounds.

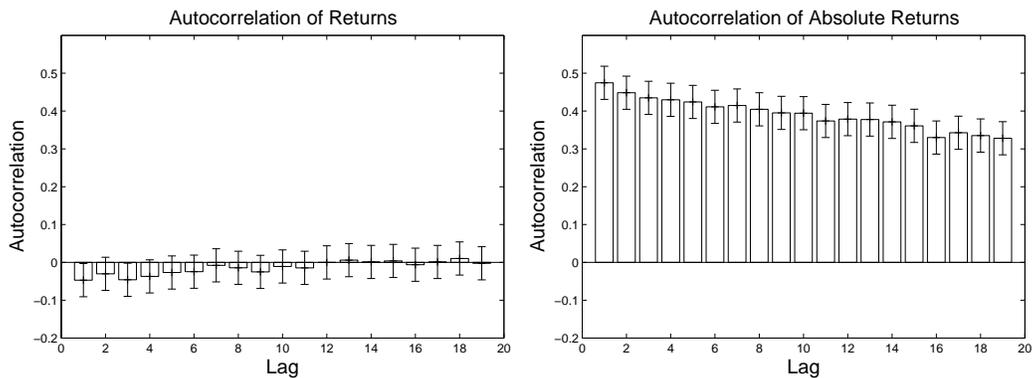


Figure 10: Autocorrelations of log returns and absolute log returns in the artificial stock market.

5.4.3 Fundamental Price

In financial markets, it is generally assumed that share price oscillates around a “fundamental” fair value, or fundamental price. This price can be related to dividends, cash flow or profits made by the firm. Empirically, it has been shown that models of the fundamental price can

account for some of the variance in share price [Shiller 1981; Kim and Koveos 1994]. Computational models of stock markets have typically assumed either a static fundamental price, or a simple time-varying price such as a first-order autoregressive process [Arthur, Holland, LeBaron, Palmer, and Taylor 1997; Gaunersdorfer 2000]. Because our model includes a consumer market, our fundamentalist traders construct a fundamental price based on the actual past profits of the firm.

Figure 11 shows a simulated stock price and the associated fundamental price, as calculated by the fundamentalist traders, from a sample run. The simulation used the parameter settings from table 3. The fundamental price was generated by using Eq.(22) but assuming an adaption rate α_f of 1.0. The fundamental price was then rescaled and translated to compensate for the actual adaption rate of the fundamentalists ($\alpha_f = 0.18$).

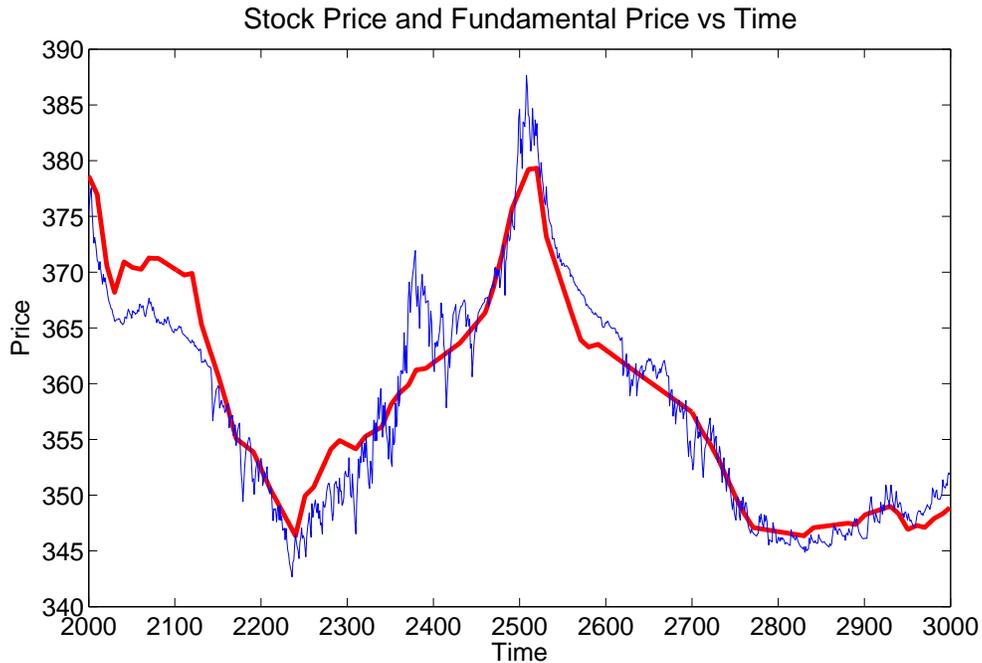


Figure 11: Fundamental price (thick line) and stock price (thin line) from a section of a single run of the integrated markets model. The fundamental price has been translated and rescaled to compensate for the adaptation rate of the fundamentalist traders.

This sequence shows several aspects of the artificial stock market. First, the stock price roughly reflects the underlying fundamental price. The price differential is due to the number of stocks held by the traders initially (in our case 120). Second, the stock price oscillates at a higher frequency than the underlying fundamental price. Despite this, fundamental price information is incorporated slowly, due to the adaption rate α_f less than 1.0. Large stock price changes lag behind similar changes in the fundamental price. Third, large changes in fundamental price lead to high volatility in the stock price. Fourth, the stock price tends to over- or under-shoot and then oscillate after a large change.

For this run, the proportion of fundamentalists was quite high ($N_f = 0.57$). It is interesting that, under our model, decreasing the proportion of fundamentalists tends to also decrease the kurtosis of the returns. In a market with only 20% fundamentalists, the returns look Gaussian. If the proportion of fundamentalists drops below 10%, the stock price collapses. The heterogeneity of the market traders is necessary to maintain market liquidity and trading volume.

If the fundamental price remains static over a long period of time, then the share price tends to decay in a deterministic way to the fundamental price. The variation in fundamental price due to the dynamics in the consumer market is an integral part of the stock returns in

our model.

5.4.4 Volatility and Trading Volume

There is a known positive correlation between volatility and trading volume in financial markets [Karpoff 1987]. That is, periods of high volatility are also those of high trading volume.

Our integrated model exhibits the same behavior. There is a correlation between volatility and trading volume. High volume and high volatility are interrelated, and each can significantly predict the other, although the effect of high volatility on trading volume is longer lasting. Figure 12 shows average cross-correlations and 95% confidence intervals for stocks from the 20 runs of the simulator, with parameters set as in Table 3.

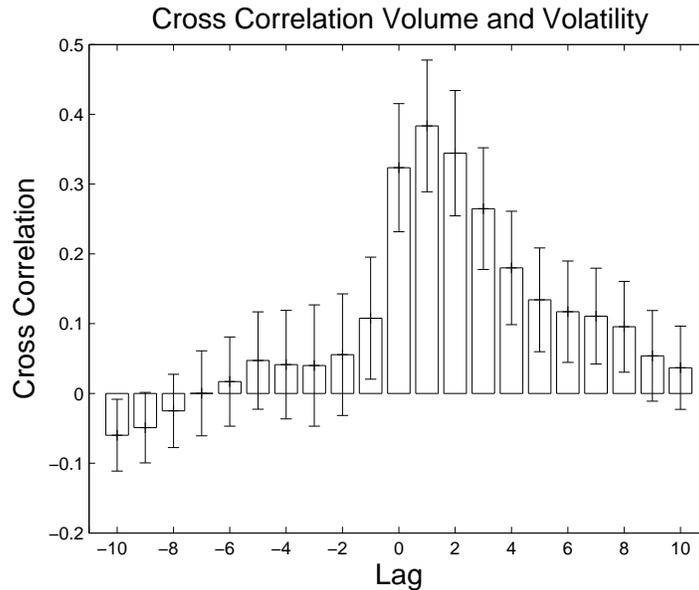


Figure 12: Cross correlation between trading volume and absolute returns. The figure was generated by averaging 45-day periods of volume and absolute returns for 40 stocks (20 runs, 2 firms per run). Cross correlations were measured for each stock. The plot shows mean cross correlations and 95% confidence intervals. The plot shows that volatility and trading volume are interrelated, with each being a significant predictor of the other, although the effect of volatility on trading volume is longer-lasting.

6 Conclusions

We have described an integrated model consisting of three agent types: production firms, consumers and financial traders. The agents operate in two coupled markets: a consumer market and a financial market. The model builds on previous work by simplifying and integrating previous models of consumers, firms and traders. We have found that for a particular reasonable setting of the parameters, a large number of stylized facts can be reproduced simultaneously in the two markets. We have also indicated in which parameter regimes the model does not perform well with respect to different stylized facts. We have shown that it is possible to incorporate a profit signal from a competitive consumer market endogenous to the model itself. This endogenous profit signal provides some of the low-frequency and large-scale variability seen in the financial market model.

We have introduced a new model validation technique based on Markov chain Monte Carlo sampling, and used the new technique to investigate under which model parameter

regimes the model exhibits realistic behaviors. We have shown that this technique can highlight interesting correlations between model parameters and offer insights into the mechanisms underlying the behavior of the model. We feel that this technique has wide applicability to other agent-based models, and is an important contribution of this paper.

We have demonstrated that the combined model is more than just the sum of its parts. The behavior of each of the markets is substantially influenced by the dynamics of the other market. In particular, firm performance in the consumer market is significantly affected by how the firm estimates future performance. Firms operate best given a mixture of performance-based and stock-based pay. Similarly, the statistical properties of the stock market are best for intermediate values of firm parameters. We are currently using the integrated model to investigate inter-market stylized facts that are beyond the reach of individual models. These include managerial compensation schemes; product hype in the financial market; consumer social networks and their effect on the financial market; and brand-recognition-based trading strategies.

Acknowledgements

This work was funded by the Austrian Science Fund (FWF) under grant SFB#010: “Adaptive Information Systems and Modeling in Economics and Management Science”. The Austrian Research Institute for Artificial Intelligence is supported by the Austrian Federal Ministry of Education, Science and Culture.

References

- Argote, L. (1999). *Organizational Learning: Creating, Retaining and Transferring Knowledge*. Kluwer Academic Publishers.
- Arthur, W. B., J. Holland, B. LeBaron, R. Palmer, and P. Tayler (1997). *The Economy as an Evolving Complex System II*, Chapter Asset pricing under endogenous expectations in an artificial stock market, pp. 15–44. Reading, MA: Addison-Wesley.
- Arthur, W. B., J. H. Holland, B. LeBaron, R. Palmer, and P. Taylor (1997). Asset pricing under endogenous expectations in an artificial stock market. In W. B. Arthur, S. N. Durlauf, and D. A. Lane (Eds.), *The Economy as an Evolving Complex System II*, pp. 15–44. Reading, MA: Addison-Wesley.
- Baier, T. and J. Mazanec (1999). The SIMSEG project: A simulation environment for market segmentation and positioning strategies. Technical report, SFB Adaptive Information Systems and Modelling in Economics and Management Science.
- Beardah, C. and M. Baxter (1996). Matlab routines for kernel density estimation and the graphical presentation of archaeological data. In H. Kammermans and K. Fennema (Eds.), *Interfacing the Past: Computer Applications and Quantitative Methods in Archaeology 1995*, *Analecta Prehistorica Leidensia* 28(1), Leiden.
- Bellman, R. E. (1957). *Dynamic Programming*. Princeton, NJ: Princeton University Press.
- Bertsekas, D. P. and J. N. Tsitsiklis (1996). *Neuro-Dynamic Programming*. Belmont, MA: Athena Scientific.
- Brock, W. and C. Hommes (1997). A rational route to randomness. *Econometrica* 65, 1059–1095.
- Brock, W. and C. Hommes (1998). Heterogeneous beliefs and routes to chaos in a simple asset pricing model. *Journal of Economic Dynamics and Control* 22, 1235–1274.
- Chiarella, C. and X. He (2001). Asset pricing and wealth dynamics under heterogeneous expectations. *Quantitative Finance* 1, 509–526.
- Chiarella, C. and X. He (2002). Heterogeneous beliefs, risk and learning in a simple asset pricing model. *Computational Economics* 19, 95–132.
- Dangl, T., E. Dockner, A. Gaunersdorfer, A. Pfister, A. Soegner, and G. Strobl (2001). Adaptive erwartungsbildung und finanzmarktdynamik. *Zeitschrift für betriebswirtschaftliche Forschung* 53, 339–365.

- Gaunersdorfer, A. (2000). Adaptive beliefs and the volatility of asset prices. Technical report, SFB Adaptive Information Systems and Modelling in Economics and Management Science.
- Grossman, S. (1989). *The Informational Role of Prices*. Cambridge, MA: MIT Press.
- Jaakkola, T. S., S. P. Singh, and M. I. Jordan (1995). Reinforcement learning algorithm for partially observable Markov decision problems. In G. Tesauro, D. S. Touretzky, and T. K. Leen (Eds.), *Advances in Neural Information Processing Systems*, Volume 7, pp. 345–352. The MIT Press, Cambridge.
- Karpoff, J. (1987). The relationship between price changes and trading volume: A survey. *Journal of Financial and Quantitative Analysis* 22, 109–126.
- Kim, M. and P. Koveos (1994). Cross-country analysis of the price-earnings ratio. *Journal of Multinational Financial Management* 4(3/4), 117–127.
- LeBaron, B., W. B. Arthur, and R. Palmer (1999). Time series properties of an artificial stock market. *Journal of Economic Dynamics and Control* 23(9–10), 1487–1516.
- Levy, M. and H. Levy (1996). The danger of assuming homogeneous expectations. *Financial Analysts Journal* 52(3), 65–70.
- Metropolis, N., A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller (1953). Equation of state calculations by fast computing machines. *Journal of Chemical Physics* 21, 1087–1092.
- Muth, J. (1961). Rational expectations and the theory of price movements. *Econometrica* 29, 315–335.
- Natter, M., A. Mild, M. Feurstein, G. Dorffner, and A. Taudes (2001). The effect of incentive schemes and organizational arrangements on the new product development process. *Management Science*. to appear.
- Raberto, M., S. Cincotti, S. Focardi, and M. Marchesi (2001). Agent-based simulation of a financial market. *Physica A* 299(1-2), 320–328.
- Rummery, G. A. and M. Niranjan (1994). On-line Q-learning using connectionist systems. Technical Report CUED/F-INFENG/TR 166, Engineering Department, Cambridge University.
- Shiller, R. (1981). Do stock prices move too much to be justified by subsequent changes in dividends? *The American Economic Review* 71, 421–436.
- Simon, H. A. (1982). *Models of Bounded Rationality, Vol 2: Behavioral Economics and Business Organization*. Cambridge, MA: The MIT Press.
- Steiglitz, K., M. Honig, and L. Cohen (1995). A computational market model based on individual action. In S. Clearwater (Ed.), *Market-Based Control: A Paradigm for Distributed Resource Allocation*. Hong Kong: World Scientific.
- Sutton, R. S. (1996). Generalization in reinforcement learning: Successful examples using sparse coarse coding. In D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo (Eds.), *Advances in Neural Information Processing Systems*, Volume 8, pp. 1038–1044. The MIT Press, Cambridge.
- Sutton, R. S. and A. G. Barto (1998). *Reinforcement Learning: An Introduction*. Cambridge, MA: The MIT Press.
- Tesauro, G. (1999). Pricing in agent economies using neural networks and multi-agent Q-learning. In *IJCAI-99*.
- Tesfatsion, L. (2002). Agent-based computational economics: Growing economies from the bottom up. *Artificial Life* 8(1), 55–82.
- Watkins, C. J. C. H. (1989). *Learning from Delayed Rewards*. Cambridge, UK: Cambridge University. Ph.D. thesis.
- Watkins, C. J. C. H. and P. Dayan (1992). Q-learning. *Machine Learning* 8, 279–292.