Behavioural Investigations of Financial Trading Agents Using Exchange Portal (ExPo)

Steve Stotter, John Cartlidge(✉), and Dave Cliff

Department of Computer Science,
University of Bristol Merchant Venturers Building,
Woodland Road, Bristol BS8 1UB, UK
john@john-cartlidge.co.uk
http://www.cs.bris.ac.uk

Abstract. Some major financial markets are currently reporting that 50 % or more of all transactions are now executed by automated trading systems (ATS). To understand the impact of ATS proliferation on the global financial markets, academic studies often use standard reference strategies, such as "AA" and "ZIP", to model the behaviour of real trading systems. Disturbingly, we show that the reference algorithms presented in the literature are ambiguous, thus reducing the validity of strict comparative studies. As a remedy, we suggest disambiguated standard implementations of AA and ZIP. Using Exchange Portal (ExPo), an open-source financial exchange simulation platform designed for real-time behavioural economic experiments involving human traders and/or trader-agents, we study the effects of disambiguating AA and ZIP, before introducing a novel method of assignment-adaptation (ASAD). Experiments show that introducing ASAD agents into a market with shocks can produce counter-intuitive market dynamics.

Keywords: Software agents · Auctions · Agent-based computational economics · ACE · Agent-based modelling · ABM · Automated trading · Computational finance · ExPo · Exchange portal · Assignment adaptation

1 Introduction

In 2001, a team of researchers at IBM [9] reported on a series of experiments to test the efficiency of two adaptive trading-agent algorithms, MGD [16] and ZIP [8], when competing directly against human traders. Previous studies using homogeneous trader populations of all-humans or all-agents had indicated that, in both cases, trading interactions within the populations rapidly and robustly converged toward theoretically optimal, and stable, dynamic equilibria. IBM's results demonstrated for the first time that, in heterogeneous populations mixing human traders with trader-agents, both MGD and ZIP consistently out-performed the human traders, achieving greater efficiency by making more profitable transactions. The IBM authors concluded with a prescient statement, predicting:

N.T. Nguyen (Ed.): TCCI XVII 2014, LNCS 8790, pp. 22–45, 2014.
DOI: 10.1007/978-3-662-44994-3_2

"in many real marketplaces, agents of sufficient quality will be developed such that most agents beat most humans". Hindsight shows that they were correct: in many of the world's major financial markets, transactions that used to take place between human traders are now being fulfilled electronically, at super-human speeds, by *automated trading* (AT) and *high frequency trading* (HFT) systems. AT and HFT systems are typically highly autonomous and dynamically adapt to changes in the market's prevailing conditions: for any reasonable definition of *software agent*, it is clear that AT/HFT systems can be considered as software agents, even though practitioners in the finance industry typically do not make much use of the phrase.

However, as the number of AT and HFT systems has increased, and as the billions of dollars worth of daily transaction volumes that they control has steadily risen, a worrying gap has emerged between theory and practice. Commercial deployments of AT/HFT continue to proliferate (some major financial markets are currently reporting that 50 % or more of transactions are now executed by automated agents), yet *theoretical* understanding of the impact of trading agent technologies on the system-level dynamics of financial markets is dangerously deficient. To address this problem, in 2010 the UK Government's Office for Science (UKGoS) launched a two year "Foresight" project entitled *"The Future of Computer Trading in Financial Markets"*.[1]

One report [12] commissioned by that project and published by UKGoS attempted a replication of IBM's study, but with two extensions: firstly, trading agents used the Adaptive Aggressive (AA) strategy [26], which had previously been shown to outperform both MGD and ZIP [11]; secondly, to increase the experimental "realism", order assignments to trade were continuously replenished, thus producing a continuous "drip-feed" market that more closely approximates the real world, rather than a discrete, periodic market as had been used in almost all prior experimental studies. Results showed that, under these experimental conditions, agents were *less* efficient than human traders, with slower markets hindering agent performance but enhancing human performance [12].

In this paper, we perform two sets of experiments. Firstly, we replicate the continuous replenishment experiments of [12] using ExPo: *The Exchange Portal*, an open-source platform designed to facilitate financial trading experiments between humans, agents, or both [13]. However, unlike [12], we study agent-only markets. Perhaps surprisingly, we believe that this is the first time agent-only markets have been studied using continuous replenishment of order assignments. For our trading agents, we use two well-known "reference" algorithms from the trading-agent literature, AA [26] and ZIP [8].

In our second set of experiments, we introduce "market shocks" to the system and explore a novel extension to the reference algorithms (assignment-adaptive, or ASAD, agents), designed to enable agents to take advantage of such shocks. We demonstrate that if all agents in the market are ASAD, then the market is more efficient in the presence of market shocks than if all agents are non-ASAD.

[1] The final report from that investigation was published in Oct. 2012, and is available at: http://bit.ly/UvGE4Q.

However, somewhat counter-intuitively, when the market is a heterogeneous mixture of ASAD and non-ASAD, non-ASAD agents outperform ASAD agents by adapting to the new price signals generated by ASAD agents.

This paper is organised as follows.[2] In Sect. 2 we review the literature on financial trading agent experiments and the agent algorithms, AA and ZIP. In Sect. 3 we introduce ExPo, our experimental platform, and describe our experimental design. In Sect. 4 we present the results from our two sets of experiments. Finally, conclusions are drawn in Sect. 5.

2 Background

2.1 The Continuous Double Auction

An auction is a mechanism whereby sellers and buyers come together and agree on a transaction price. Many auction mechanisms exist, each governed by a different set of rules. In this paper, we focus on the *Continuous Double Auction* (CDA), the most widely used auction mechanism and the one used to control all the world's major financial exchanges. The CDA enables buyers and sellers to freely and independently exchange quotes at any time. Transactions occur when a seller accepts a buyer's "bid", or when a buyer accepts a seller's "ask". Although it is possible for any seller to accept any buyer's bid, and *vice-versa*, it is in both of their interests to get the best deal possible at any point in time. Thus, transactions execute with a counter party that offers the most competitive quote.

Vernon Smith explored the dynamics of CDA markets in a series of Nobel Prize winning experiments using small groups of human participants [20]. Splitting participants evenly into a group of buyers and a group of sellers, Smith handed out a single card (an *assignment*) to each buyer and seller with a single *limit price* written on each, known only to that individual. The limit price on the card for buyers (sellers) represented the maximum (minimum) price they were willing to pay (accept) for a fictitious commodity. Participants were given strict instructions to not bid (ask) a price higher (lower) than that shown on their card, and were encouraged to bid lower (ask higher) than this price, regarding any difference between the price on the card and the price achieved in the market as profit.

Experiments were split into a number of "trading days", each typically lasting a few minutes. At any point during the trading day, a buyer or seller could raise their hand and announce a quote. When a seller and a buyer agreed on a quote, a transaction was made. At the end of each trading day, all stock (sellers assignment cards) and money (buyer assignment cards) was recalled, and then reallocated anew at the start of the next trading day. By controlling the limit prices allocated to participants, Smith was able to control the market's supply and demand schedules. Smith found that, typically after a couple of trading days, human traders achieved very close to 100 % allocative efficiency; a measure of

[2] For an earlier version of the work presented here, we refer the reader to [23].

the percentage of profit in relation to the maximum theoretical profit available (see Sect. 2.2). This was a significant result: few people had believed that a very small number of inexperienced, self-interested participants could effectively self-equilibrate.

2.2 Measuring Market Performance

An "ideal" market can be perfectly described by the aggregate quantity supplied by sellers and the aggregate quantity demanded by buyers at every price-point (i.e., the market's supply and demand schedules, Fig. 1). As prices increase, in general there is a tendency for supply to increase, with increased potential revenues from sales encouraging more sellers to enter the market; while, at the same time, there is a tendency for demand to decrease as buyers look to spend their money elsewhere. At some price-point, the quantity demanded will equal the quantity supplied. This is the theoretical market equilibrium. An idealised theoretical market (and many real ones) has a *market equilibrium* price and quantity (P_0, Q_0) determined by the intersection between the supply and demand schedules. The dynamics of competition in the market will tend to drive transactions toward this equilibrium point. For all prices above P_0, supply will exceed demand, forcing suppliers to reduce their prices to make a trade; whereas for all prices below P_0, demand exceeds supply, forcing buyers to increase their price to make a trade. Any quantity demanded or supplied below Q_0 is called

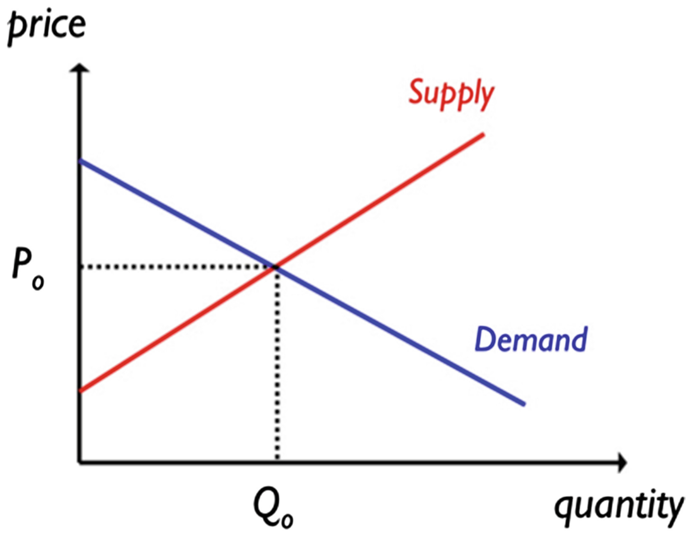

Fig. 1. Supply and Demand curves (here illustrated as straight lines) show the quantities supplied by sellers and demanded by buyers at every price-point. In general, as price increases, the quantity supplied increases and the quantity demanded falls. The point at which the two curves intersect is the theoretical equilibrium point; where Q_0 is the equilibrium quantity and P_0 is the equilibrium price.

"intra-marginal"; all quantity demanded or supplied in excess of Q_0, is called "extra-marginal". In an ideal market, all intra-marginal units and no extra-marginal units are expected to trade.

In the real world, markets are not ideal. They will always trade away from equilibrium at least some of the time. We can use metrics to calculate the "performance" of a market by how far from ideal equilibrium it trades, allowing us to compare between markets. In this report, we make use of the following metrics:

Smith's Alpha. Following Vernon Smith [20], we measure the equilibration (equilibrium-finding) behaviour of markets using the coefficient of convergence, α, defined as the root mean square difference between each of n transaction prices, p_i (for $i = 1 \ldots n$) over some period, and the P_0 value for that period, expressed as a percentage of the equilibrium price:

$$\alpha = \frac{100}{P_0} \sqrt{\frac{1}{n} \sum_{i=1}^{n} (p_i - P_0)^2}. \tag{1}$$

In essence, α captures the standard deviation of trade prices about the theoretical equilibrium. A low value of α is desirable, indicating trading close to P_0.

Allocative Efficiency. For each trader, i, the maximum theoretical profit available, π_i^*, is the difference between the price they are prepared to pay (their "limit price") and the theoretical market equilibrium price, P_0. Efficiency, E, is used to calculate the performance of a group of n traders as the mean ratio of realised profit, π_i, to theoretical profit, π_i^*:

$$E = \frac{1}{n} \sum_{i=1}^{n} \frac{\pi_i}{\pi_i^*}. \tag{2}$$

As profit values cannot go below zero (traders in these experiments are not allowed to enter into loss-making deals), a value of 1.0 indicates that the group has earned the maximum theoretical profit available, π_i^*, on all trades. A value below 1.0 indicates that some opportunities have been missed. Finally, a value above 1.0 means that additional profit has been made by taking advantage of a trading counterparty's willingness to trade away from P_0. So, for example, a group of sellers might record an allocative efficiency of 1.2 if their counterparties (a group of buyers) consistently enter into transactions at prices greater than P_0; in such a situation, the buyers' allocative efficiency would not be more than 0.8.

Profit Dispersion. Profit dispersion is a measure of the extent to which the profit/utility generated by a group of traders in the market differs from the profit that would be expected of them if all transactions took place at the equilibrium price, P_0. For a group of n traders, profit dispersion is calculated as the root mean square difference between the profits achieved, π_i, by each trader, i, and the maximum theoretical profit available, π_i^*:

$$\pi_{disp} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\pi_i - \pi_i^*)^2}. \quad (3)$$

Low values of π_{disp} indicate that traders are extracting actual profits close to profits available when all trades take place at the equilibrium price P_0. In contrast, higher values of π_{disp} indicate that traders' profits differ from those expected at equilibrium. Since zero-sum effects between buyers and sellers do not mask profit dispersion, this statistic is attractive [17].

2.3 Algorithmic Traders

Zero-Intelligence Plus (ZIP) agents were developed by Dave Cliff [8] to overcome the provable shortcomings of Gode & Sunder's ZI-C agents [17]. ZIP agents are profit-driven traders that adapt using a simple learning mechanism: adjust profit margins based on the price of other bids and offers in the market, and decide whether to make a transaction or not. When a decision to raise or lower a ZIP trader's profit margin, $\mu_i(t)$, is taken, ZIP modifies the value using market data and an adaptation rule based on the Widrow-Hoff "delta rule" [28]:

$$\Delta_i(t) = \beta_i(\tau_i(t) - p_i(t)), \quad (4)$$

where β_i is the *learning rate*, p_i is the quote price and τ_i is the *target price* (based on the price of the last quote in the market). At time t, an update to the profit margin, μ_i, takes the form:

$$\mu_i(t+1) = \frac{p_i(t) + \Gamma_i(t+1)}{l_i - 1}, \quad (5)$$

and

$$\Gamma_i(t+1) = \gamma_i(t) + (1 - \gamma_i)\Delta_i(t), \quad (6)$$

where $\Gamma_i(t+1)$ is the amount of change on the transition from t to $t+1$, and γ_i is the *momentum* coefficient. Given the limit price, l_i, of the current assignment, ZIP then updates its profit margin, $\mu_i(t)$, based on these trading rules, where the final quote price, p_i, is given as:

$$p_i = l_i(1 + \mu(t)). \quad (7)$$

The ZIP strategy has become a popular benchmark for CDA experiments. In their IBM study, [9] concluded that ZIP was a dominant strategy, beating humans in experimental trials and matching the performance of their own modified GD [16] algorithmic trader. More recently, ZIP has again been shown to outperform humans [10,11]. However, it is no longer considered the dominant agent strategy (having been shown to be beaten by AA; see Sect. 2.3). ZIP has also been tested against humans in a continuous "drip-feed" market, where ZIP was shown to be *less* efficient than humans (a result that surprised the authors) [7,12]. However, we believe that De Luca's implementation of ZIP [18] that was used in those experiments may have played some part in this result.

The original implementation of ZIP [8] was designed to handle only one limit price, had no explicit notion of time and no persistent orders. So, when the IBM team used ZIP to conduct human vs. agent experiments, they adapted ZIP for their platform [9]. In order to handle persistent orders, a "sleep time" was introduced into ZIP, such that if no trade took place within a given time period, then the ZIP agent would automatically initiate a competitive price movement, i.e., a price movement towards the best value on the other side of the order book [ibid]. Perhaps more importantly, ZIP was further modified to have a vector of internal price variables, allowing profit to be made at different values for different assignments. These modifications were similar to an alternative implementation that had been independently proposed in a previous study [19]. Other versions of ZIP also appear in the literature. In [26], ZIP (and presumably, also AA) algorithms were forced to update only the *most profitable* bid (for buyers) or ask (for sellers) at any one time. This approach was replicated in De Luca's open-source implementation of ZIP and AA [18]. Finally, ZIP has also been adapted to enable arbitrage, by allowing an individual agent to both buy *and* sell. Initially introduced by [25], and recently adapted by [2], ZIP "arbitrageurs" contain two profit margins (buy and sell) and the price adjustment mechanism adjusts two prices each time the agent receives new market information. For this reason, ZIP arbitrageurs can be considered equivalent to two standard ZIP agents (one buyer and one seller) working as a team.

Here, we test to see whether a ZIP implementation with multiple profit margins, ZIP_M, is more efficient than a ZIP trader with a single profit margin, ZIP_S. As far as we are aware, this comparison has not been directly tested before. We use ZIP_S to describe the implementation in [26], where only the most profitable order is updated on every wakeup; and ZIP_M to denote an implementation of ZIP similar to that used in [9,19,24], such that ZIP_M is capable of updating all profit margins for all orders simultaneously. Every unique limit price received is given a new μ and γ (the values of μ and γ are decided at random when the agent is started) and all ZIP parameters are the same as those used in [8].

Adaptive-Aggressive (AA) agents were developed by Vytelingum [26] to explicitly model "aggressiveness"—trading the opportunity of extra profit for the certainty of transacting. Aggressive agents enter competitive bids (or asks) for a quick trade, while passive agents forgo the chance of a quick trade in order to hold out for greater profit. To control the level of aggressiveness, AA uses the Widrow-Hoff delta rule [28] that is also used in ZIP (Eq. 4). However, whereas ZIP uses learning to update profit margin, AA updates an aggression parameter based on previous market information. At time, t, AA estimates the competitive equilibrium price, p^*, based on a moving window of historic market transaction prices; p^* is then used in AA's long-term adaptation component, which updates θ, a property of the aggressiveness model. In this long-term adaptation component, an internal estimate of Smith's α (Eq. 1) is calculated, enabling the agent to detect and react to price volatility. AA was developed to perform well in dynamic markets. Short-term learning is used to react to the current state of the market, while long-term learning is used to react to market trends. AA has been shown to dominate other agent strategies in the literature [11,26], however, unlike ZIP, which has been independently re-implemented by many different researchers, we believe the only replication of AA in the literature prior to this study is De Luca's OpEx implementation [18].

In Vytelingum's original AA implementation [26], it is unclear how an agent should quote when the market first opens and is empty. In De Luca's version [18], AA uses the maximum bid or ask price allowed in the market, $P_{max} = 400$, to determine an agent's initial quote price, $p_{t=0}$, such that $p_{t=0}$ is a random variable from a uniform distribution with range $[0.15P_{max}, 0.85P_{max}]$. In the absence of any "real" market data, the value $p_{t=0}$ acts as a proxy for the initial estimate of market equilibrium. But, since $p_{t=0}$ is artificially constrained by the arbitrary market value P_{max}, we believe that this method of generating $p_{t=0}$ is not domain independent and may present AA with an unfair "equilibrium finding" advantage when compared with other agent strategies, such as ZIP, which do not have access to this parameter. Moreover, for their first quote price, De Luca's OpEx agents [18] do not make use of the limit prices of their internal assignments (other than to maximally bound the quote at the bid limit and minimally bound at the ask limit). We believe this to be unrealistic. At the beginning of the market the only information agents have available for price discovery are their own personal assignments. Therefore, it is intuitive that agents should try to benefit from any information contained therein. For this reason, we introduce a modification to AA whereby agents set their own internal estimation of P_{max} such that P_{max} equals twice the maximum assignment limit price an agent holds.[3] Readers should note that agents could only submit a quote once they had received an assignment to trade.

In March 2012, an unexpected "max spread rule" in De Luca's AA code of OpEx version 1 was exposed [5]. This rule states that an agent should automatically execute against the best quote on the other side of the book if the relative spread (the difference between best quotes on either side of the book) is within a threshold, $maxSpread$ (and within limit price range).[4] Although this rule is not described in the definition of AA, we believe that it is a vestigial morph of a spread rule appearing in Risk-Based (RB) agents [27], a previous trader agent that Vytelingum eventually developed into AA [26]. The max spread rule encourages De Luca's AA agents to "jump the spread" for a quick transaction. However, in OpEx version 1, $maxSpread$ was hard-coded to a value of 15 %. Following [5], we believe that this value is unrealistically large and therefore casts a question of doubt on the validity of previous experimental results gathered using these agents.[5] In this

[3] We do not suggest that two is the optimum multiplier for this equation; rather we aim to investigate the effect of introducing this modification and select two as a simple heuristic estimate.

[4] For a lengthy discussion on the consequences of the max spread rule, see [5].

[5] Since this issue was raised by [5], the spread jumping rule has subsequently been classified as a bug and removed from De Luca's OpEx AA agents (http://sourceforge.net/p/open-exchange/tickets/1/).

paper, we explore the effect of the spread-jumping rule. Unless otherwise stated, we remove the *maxSpread* condition (i.e., set *maxSpread* = 0 % for our AA agents). All other AA parameters are set to those suggested by [26]. Following the literature, we also use the rule of updating only the *most profitable* bid (for buyers) or ask (for sellers) at any one time (similar to ZIP_S).

3 Methodology

3.1 ExPo: Exchange Portal Platform

Exchange Portal [13] is a real-time online financial trading exchange platform designed to run controlled scientific trading experiments between human traders and automated trader robots (see Fig. 2). ExPo was developed at the University of Bristol as both a teaching and research platform and has been open-sourced as a gift to the wider research community. ExPo can be run across a network (e.g., the internet), with human and/or automated trader agents messaging the exchange via HTTP. Alternatively, ExPo can be run on a single machine, with all clients running locally. For all experiments detailed in this paper, we run ExPo and the agent traders on the same physical machine. Prior to running experiments, ExPo was stress-tested through a rigorous series of agent-only experiments (see [22]).

Figures 3 and 4 show a typical set up for an auction using the admin GUI (Fig. 3) and an example of ExPo in operation (Fig. 4). The assignment sequences

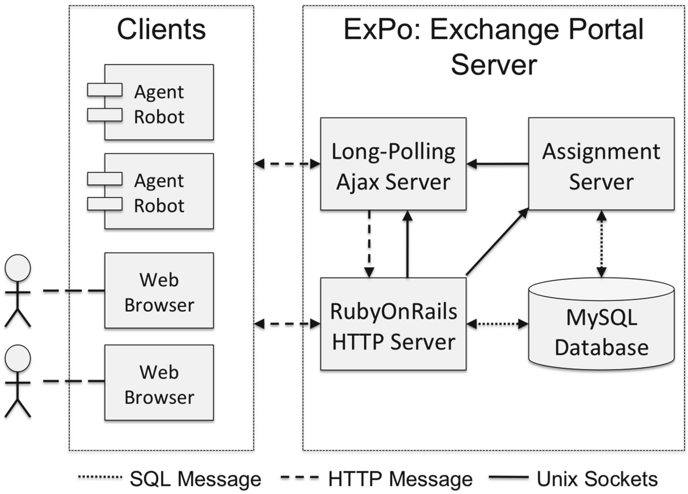

Fig. 2. ExPo architecture. The ExPo exchange is a Ruby on Rails web server application with RESTful architecture, using a MySQL database for storage. Clients (automated trader agents, or human traders using a web browser) connect and message the server using HTTP messaging. ExPo internal servers communicate via unix sockets.

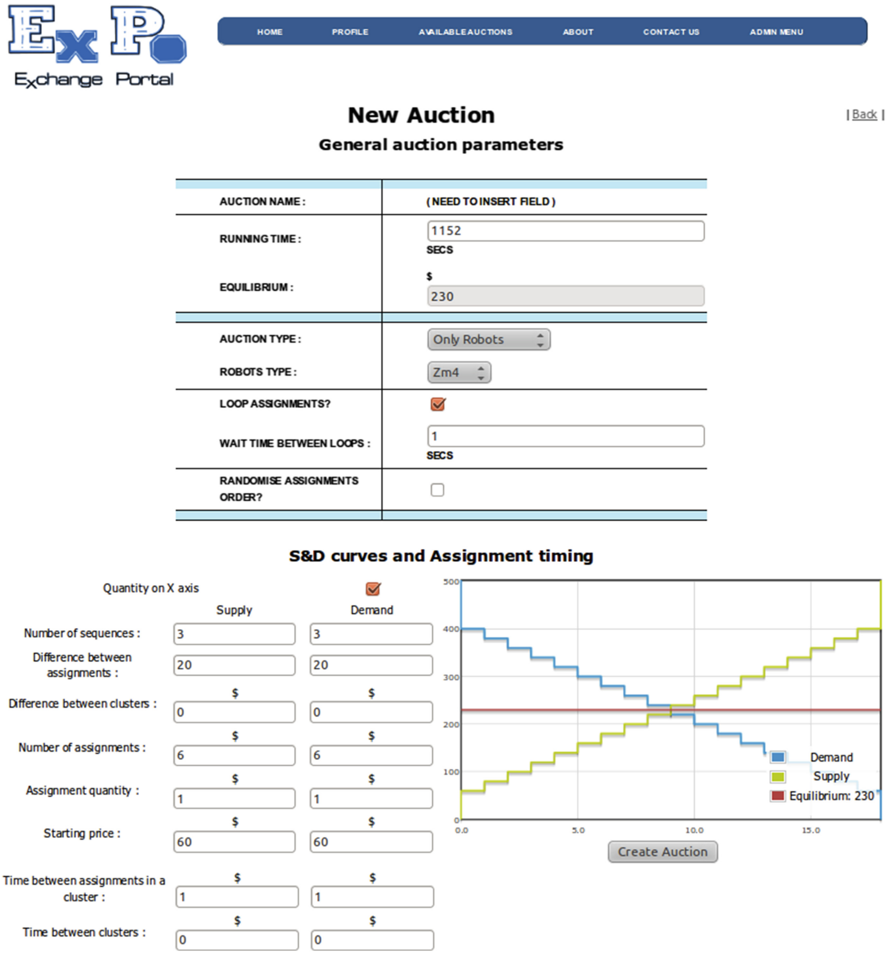

Fig. 3. Screenshot of ExPo's auction configuration GUI, used to initialise a financial trading experiment. Top: the auction parameters table is used to name a market experiment, define the market running time, set the market equilibrium price, link to the trader agent algorithm code, and select whether or not human users are able to participate. Bottom: the assignment sequences for participants are configured using the text boxes on the left, and illustrated dynamically by the graph on the right, with the blue line indicating aggregate market demand and the yellow line indicating aggregate market supply.

for participants are looped until the end of the auction. When competitors are added to an auction through the automation scripts, they are put on the same assignment sequences as already exist in the market. This is designed to avoid accidentally introducing an asymmetrical advantage for any one group.

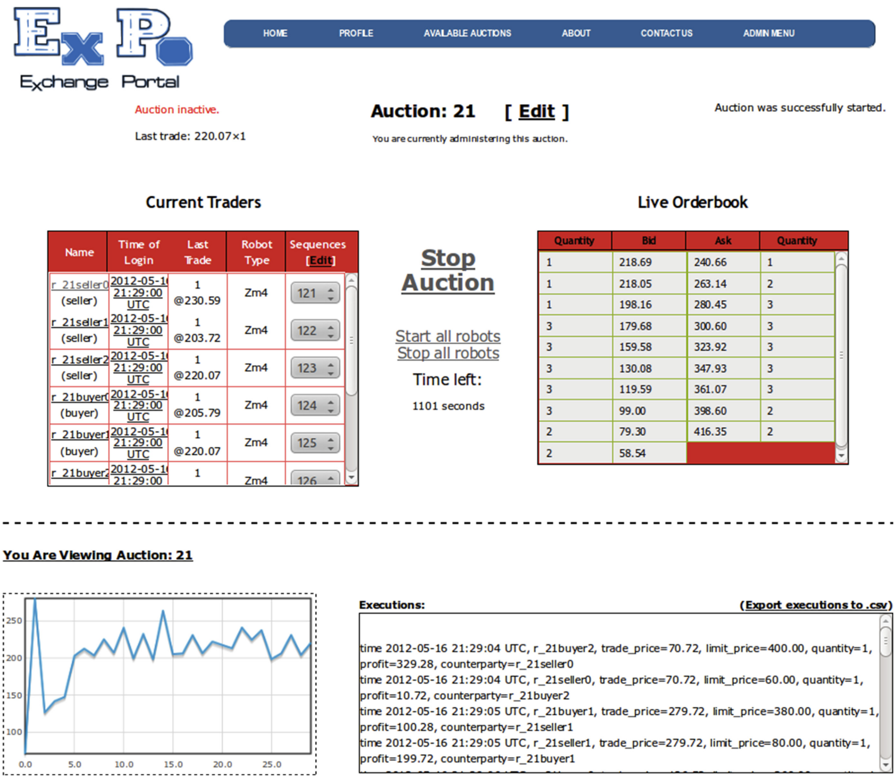

Fig. 4. ExPo screenshot of the admin screen (not available to ordinary market participants) during an open market period. Top-left: table showing details of all traders (human and robot) participating in the market. Top-right: the public order book displays current prices and volumes quoted in the market. Bottom-left: execution prices of trades are plotted dynamically. Bottom-right: an exportable list of all market transactions are detailed.

3.2 Experiment Design

Market environments used in previous experiments typically follow the "trading day" model of Smith's original experiments (notable exceptions include [5–7, 12]). The problem with this is that it assumes traders only get new assignments at the start of each trading day—typically only one assignment each. Platforms like ExPo help to model markets in a more realistic way. By modelling a market as a continuous replenishment auction, we are able to model in *real time*, allowing assignments to drip feed into the market like they would if you were a sales trader on a financial trading desk, receiving assignments from clients throughout the day.

Each agent strategy in the market was grouped into 3 buyers and 3 sellers. The running time for each auction was 1152 s (64 assignment "loops" of 18 s each), similar to the 20 min length of time that was used in [12]. Assignments

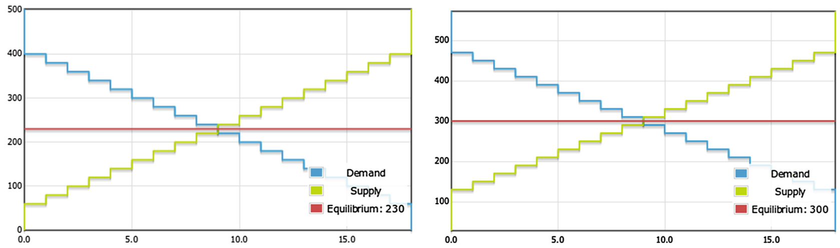

Fig. 5. Supply and demand assignment sequences used for experiments. Left: equilibrium price, $P_0 = 230$. Right: equilibrium price $P_0 = 300$. Each agent (3 buyers and 3 sellers) receives 6 assignments per assignment "loop", hence the total volume demanded and supplied per loop is 18 and $Q_0/loop = 9$. Assignments are allocated in pairs (to one seller and one buyer) every second, with each agent receiving a new assignment on average every 3 s. Assignment loops are repeated 64 times, producing a total experiment running time of 1152 s, and an equilibrium quantity $Q_0 = 64 \times 9 = 576$.

were sequentially allocated in pairs (to one buyer and one seller) every second, thus for each agent the mean time between assignments received was 3 s. Each assignment "loop" (see Fig. 5), agents each received 6 assignments with different limit prices. As assignments belonging to an agent are grouped by limit price, when an agent receives a new assignment the assignment quantity for that limit price was incremented. All agents treat current holdings of assignments as a single entity, increasing or decreasing their quote price as a group. However, one or multiple assignments may be traded from a group at any time if only a certain number are able to transact on the order book. No retraction of assignments was permitted, and once assignments were distributed, their limit prices could not be modified. For all experiments, equilibrium was set at 230 (Fig. 5, left), and raised to 300 (Fig. 5, right) when a "market shock" occurred. We do *not* use the NYSE spread-improvement rule, thus enabling traders to submit quotes at any price.

When a new assignment is provided to an agent, that agent has the ability to put it straight on the order book. Although agents can create new orders immediately, each agent can only update their orders once a sleep-time, s, has expired. While the agent is asleep (we can think of this as a "thinking" period), it is still actively able to calculate a new order price using shouts and transactions in the marketplace. Once sleep-time has elapsed, an agent is able to update their order price. The ability to put new assignments on the order book as soon as they are received is an important difference to previous implementations of sleep-time. An order placed immediately on the book is more advantageous than delaying a trade by waiting. The sleep-time of each agent was set randomly within a boundary of ±(0-25) % of the sleep-time provided. This is the same "jitter" setting implemented in [9]. For all experiments reported here, we set sleep-time $s = 4$ s. While it is not strictly necessary to enforce a period of sleep time in agents (on the scale of human reaction times) when the market contains no humans, we do this to replicate the experimental method of [7,12]. This enables us to directly compare results, and hence challenge or confirm any of their conclusions.

All experiments were repeated 5 times and results analysed using the non-parametric Robust Rank-Order (RRO) statistical test [14,15]. The number of trials was necessarily restricted due to the real-time nature of experiments, with each run taking approximately 20 min.

4 Results

4.1 AA Modifications

Here, we present results from a series of experiments between the "reference" AA agents from the literature, and the modifications we suggested in Sect. 2.3.

The Effect of P_{max} on AA. In De Luca's implementation of AA [18], agents use the OpEx system parameter $P_{max} = 400$. For the majority of OpEx experiments, markets were engineered to have an equilibrium value of $P_0 = 200$, exactly *half* the value of P_{max}, e.g., [7,12]. We believe that the use of this system parameter by AA agents may produce artifactual dynamics and favourably bias AA agents (when compared with other agents, such as ZIP, that do *not* make use of this system parameter). Here, we test three implementations of AA to observe the effect P_{max} has on AA dynamics: AA_L, with *low* value $P_{max} = 500$; AA_H with *high* value $P_{max} = 2000$; and AA_D, with *dynamic* $P_{max} = 2 \times \max(limitPrice)$. The value used for AA_L was purposely set to be approximately twice equilibrium (set to $P_0 = 230$ in all experiments) to enable comparison with OpEx results. Note that, since limit price is exogenously assigned to agents via the supply and demand permit schedules, P_{max} will vary between AA_D agents. For example, if an agent, a, receives 2 sell assignments with limit prices 250 and 350, then $P_{max} = 700$ for that agent, a. For buy assignments, quote prices are implicitly bounded by zero.

Figure 6 displays mean Smith's α across 5 runs of homogeneous AA_L, AA_H and AA_D markets. We see that a lower value of P_{max} encourages better market equilibration by constraining the "exploration" of initial equilibrium values. This suggests that P_{max} introduces an artificial system bias. In heterogeneous markets (containing 3 AA_L and 3 AA_H on *each* side) AA_L agents gained greater efficiency in 4 of the 5 experiments. However, using Robust Rank Order (RRO) [15] this result was not statistically significant at the 10.3 % level.

Table 1 summarises the performance of homogeneous AA_L, AA_H and AA_D markets. We see that P_{max} has virtually no effect on efficiency, but has a large effect on Smith's α and profit dispersion. There is no significant difference between the efficiencies or α values of homogeneous AA_D and AA_H markets. We believe the reason AA_D did not outperform AA_H on these metrics is due to the assignment distribution pattern. In all experiments, assignments are distributed in descending order, such that buy assignments with the highest limit

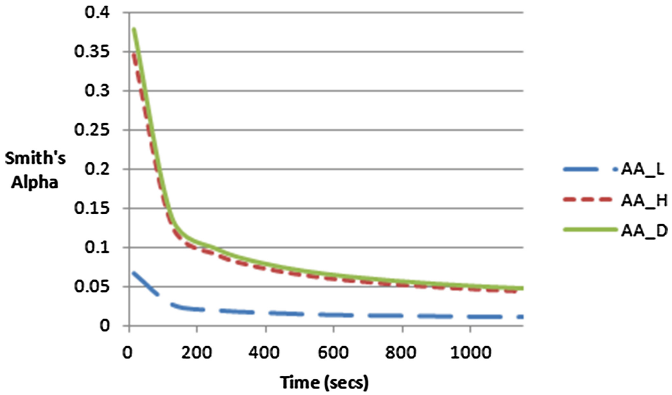

Fig. 6. Smith's α over time for each homogeneous AA market. AA_L produces lower α than AA_H, demonstrating that lower values of P_{max} artificially encourage equilibration. AA_D performs similarly to AA_H, but does not rely on the market dependent P_{max} value and hence is more robust.

Table 1. Performance of AA with varying values of P_{max}. While efficiency varies little between the three settings, AA_L produces significantly lower Smith's α and profit dispersion, verifying that the spurious variable P_{max} affects market dynamics.

Strategy	Efficiency	Alpha	Profit dispersion
AA_L	0.999372	0.0114	97.3
AA_H	0.999365	0.0436	204.4
AA_D	0.999323	0.0469	253.4

prices are always allocated first. Therefore, initial values of P_{max} for AA_D agents are higher than they would be otherwise.

Having shown that AA agents are sensitive to the system value P_{max}, we propose that AA agents should be modified to dynamically adapt their own *internal* value of P_{max}. For the remainder of this paper, unless stated otherwise, we use the dynamic AA_D version of AA.

The Effect of *maxSpread* on AA. In OpEx version 1 [18], AA agents had a fixed parameter value $maxSpread = 15\,\%$. These agents were used in [7,12]. Here, we test the effect of this parameter by comparing homogeneous and heterogeneous markets containing two AA versions: AA_D with no *maxSpread* condition; and AA_D^{MS} with $maxSpread = 15\,\%$.

Figure 7 displays the time series of trade prices from one example run of a homogeneous AA_D^{MS} market (left) and homogeneous AA_D market (right). As we would expect, AA_D^{MS} markets have greater price volatility and less equilibration to P_0, with AA_D^{MS} happy to "jump" a spread of 15 %. Conversely, AA_D agents will post quotes closer to equilibrium and wait to be "hit". Table 2

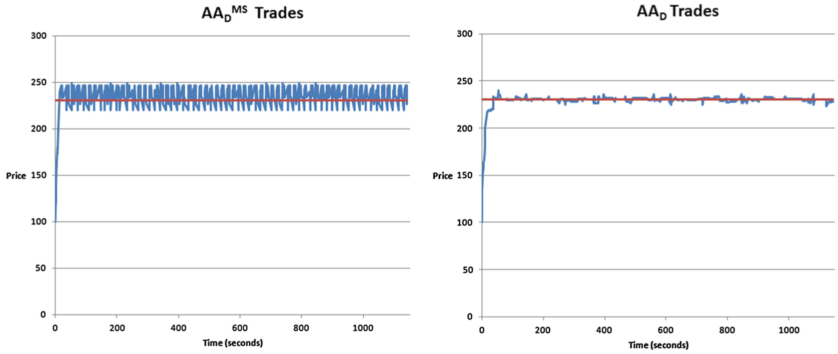

Fig. 7. Trade prices executed in homogeneous markets of AA agents with maxSpread rule (left) and no maxSpread rule (right). Left: AA_D^{MS} agents ($maxSpread = 15\,\%$) produce volatile trading dynamics, with execution prices rapidly fluctuating above and below equilibrium price $P_0 = 230$, within a region approximately bounded by $P_0 \pm 7.5\,\%$. Right: AA_D agents ($maxSpread = 0\,\%$) produce much more stable dynamics, with executions clustered closely around P_0. Since AA_D^{MS} agents are happy to accept prices away from equilibrium (within the maxSpread limit), maxSpread markets (left) are more *liquid* (produce more trade executions) than non-maxSpread markets (right).

Table 2. Mean results summary (5 runs) of *fast* homogeneous markets, allocating assignments every 3 s. ZIP_M performs significantly better than ZIP_S across all measures. AA_D outperforms AA_D^{MS}, and significantly dominates overall.

Agent	Trials	Efficiency	Smith's α	Profit disp.	Total shouts	Total trades
ZIP_S	5	0.974	0.0664	678.6	4245	582
ZIP_M	5	0.995	0.0529	308.6	7479	594
AA_D^{MS}	5	0.988	0.0658	530.5	4036	639
AA_D	5	0.999	0.0469	253.4	4104	577

summarises mean results (5 runs) across all homogeneous markets. Comparing AA_D^{MS} with AA_D, we see that the "spread jumping" behaviour of AA_D^{MS} results in lower efficiency, higher α (less equilibration) and greater profit dispersion. AA_D^{MS} markets also execute roughly $10\,\%$ more trades than AA_D, producing the most *liquid* markets of all strategies tested. However, it should be noted that although AA_D^{MS} made more trades, they were not more profitable. In heterogeneous markets containing 2 agent types (with 3 agents of each type on each side), AA_D gained significantly higher efficiency than AA_D^{MS} (RRO, $p \leq 0.004$).

4.2 ZIP Modifications

Single vs. Multiple Profit Margins. We tested multi-profit margin, ZIP_M, and single-profit margin, ZIP_S, in a series of homogeneous markets. Table 2 summarises mean results (5 runs). ZIP_M is significantly more efficient than ZIP_S

in *fast* continuous replenishment markets, with 3 s between assignments (RRO, $p \leq 0.004$). However, this superiority diminishes as the market slows. With 6 s between assignments, ZIP_M still has significantly greater efficiency (RRO, $0.004 \leq p \leq 0.008$), but with 12 and 24 s between assignments, ZIP_M are no longer more efficient. This suggests that holding a vector of simultaneously adjustable profit margins is more effective in markets where a quick response is necessary.

Overall, AA_D is the dominant strategy of the four tested (see Table 2), with significantly higher allocative efficiency and significantly lower Smith's α than both ZIP_M and ZIP_S across all market speeds (RRO, $p < 0.048$). This confirms the dominance of AA over ZIP reported in the literature (for the full set of detailed results, see [22]).

4.3 Market Shocks

Thus far, we have assessed the performance of agents in *static* markets with a fixed theoretical equilibrium, P_0. Here, we test the performance of agents in *dynamic* markets that experience a market "shock", such that P_0 changes value mid-way though an experiment. For brevity, we only present results for shocks where the market equilibrium, P_0, increases. However, the reader should note that shocks where P_0 decreases are equally likely and lead to symmetrically similar results (see Fig. 8). As such, where buyers benefit from a shock in one direction, sellers will equally benefit from a shock in the other. When a market shock occurs, new assignments entering the market are perturbed by the same value as the shock. For example, if a market shock moves P_0 from 230 to 300, all new assignment allocations are given an increased limit price 70 units higher than they were before the shock. Real-world financial markets are inherently dynamic, experiencing continual supply and demand fluctuations. By exploring dynamic markets we aim to better understand the dynamics of agent traders in real-world markets.

When a market shock occurs, assignments that have already been allocated into the market are *not* recalled. Thus, the *actual* market equilibrium P_0' does not immediately move to the new theoretical market equilibrium P_0. Rather, P_0' asymptotically tends toward P_0, only reaching P_0 when all assignments allocated *before* the market shock have executed. We use this model of assignment *persistency* since we assume agents are acting as sales traders—assigned by a client to buy or sell on their behalf. Figure 8 illustrates example markets containing, from left to right, ZIP_S, ZIP_M and AA_D agents. In each case, we see transaction prices gradually tend toward the new equilibrium after a market shock. These results are different to those seen in discrete *trading day* experiments presented in the literature, where markets tend to re-equilibrate much quicker. However, we believe the setup we use here to be a more accurate model of real markets.

Table 3 summarises the mean profits of traders across 5 experiments with *positive* market shocks; i.e., shocks in which P_0 *increases*. Results for *negative* market shocks are symmetrically similar. For brevity, we do not present results for negative shocks, since all conclusions drawn are the same as those for positive

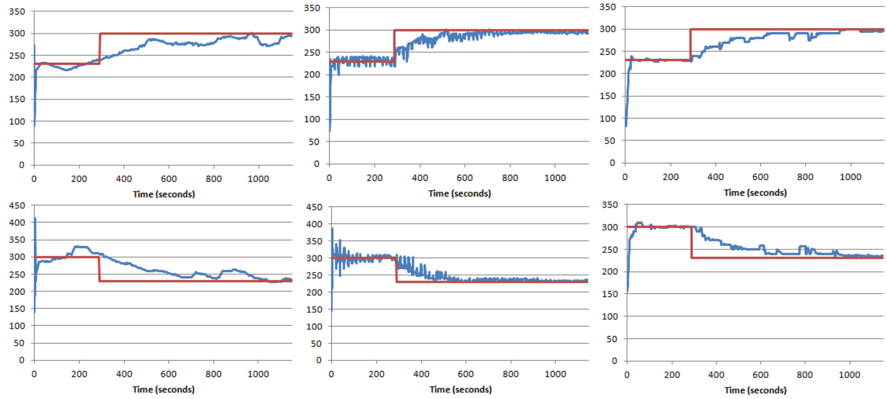

Fig. 8. Illustrative example of a market shock. Top row: a *positive* shock from $P_0^t = 230$ to $P_0^{t+1} = 300$. Bottom row: a *negative* shock from $P_0^t = 300$ to $P_0^{t+1} = 230$. Markets containing only ZIP_M agents (centre) re-equilibrate after a market shock more quickly than ZIP_S (left) and AA_D (right). Market dynamics are symmetrically similar for positive (top) and negative (bottom) shocks.

Table 3. Mean profit in *positively* shocked homogeneous markets.

	Average profit per trade		
Strategy	Buyers	Sellers	% difference
ZIP_S	97.08	71.65	35.50 %
ZIP_M	90.62	72.50	24.99 %
AA_D	98.28	69.46	41.49 %

shocks. We see that, in all cases, positive shocks benefit buyers (similarly, negative shocks benefit sellers). This is because, for the period that P_0' is below P_0, buyers have the opportunity to trade at a "cheap" price. In Fig. 8, top row, the area between the new equilibrium line (in red) and the transaction time-series (in blue) is additional profit that buyers are making and that sellers miss out on (similarly, for negative shocks, bottom row, this is additional profit for sellers). We can quantify this by the percentage difference in the average profit per trade of buyers and sellers (Table 3). We see that ZIP_M markets have significantly lower profit spread (RRO, $0.071 < p < 0.089$), indicating quicker re-equilibration after market shock. There is no significant difference in profit spread between ZIP_S and AA_D markets. We believe shocked homogeneous markets containing ZIP_M agents are able to re-equilibrate more quickly due to ZIP_M agents' ability to update multiple orders each time they "wake". Thus, if we ran further experiments using AA_D agents with multiple profit margins, we would similarly expect a decrease in re-equilibration time.

However, while both AA and ZIP agents are able to re-equilibrate *after* market shocks, neither algorithm is specifically designed to *anticipate* price movements following a shock. In the following section, we explore the effects of adding such a novel mechanism.

4.4 Assignment-Adaptive Agents

If an agent is capable of analysing their own assignments to see if there is an inherent rise (or fall) in value, then it may be possible to infer that a market shock has occurred, thus enabling the agent to anticipate a rise (fall) in transaction prices. By adjusting profit margins accordingly, the agent may be able to secure greater profit. Here, we introduce a preliminary method for agents to adapt their profit margins using information contained in their own assignment orders. We call these agents *Assignment Adaptive* (ASAD). This is exploratory work and is not intended to be a definitive solution. Rather, we are more interested in the dynamics of markets that contain such agents. For all experiments, we use ZIP_M agents, previously shown to most quickly re-equilibrate after market shocks. Once again, we present results for positive market shocks only. However, results for negative shocks are symmetrically similar and the same conclusions can be drawn for shocks in both directions.

ASAD agents store assignment limit prices in a rolling memory window containing the last 20 prices, ordered oldest to youngest. Agents only begin acting on these prices once the window is filled (i.e., once an agent has received and stored 20 assignment prices). ASAD agents then calculate the gradient of change in assignment prices using Ordinary Least Squares (OLS) regression [21], such that gradient, ∇, is:

$$\nabla = \frac{\sum x_i y_i - \overline{y} \sum x_i}{\sum x_i^2 - \overline{x} \sum x_i}, \tag{8}$$

where x_i is the index position of assignment limit price y_i in the assignment price window. Figure 9 provides a visual example of how this gradient calculation can help to detect a change in prices. This gradient value, ∇, is then transformed using a simple logarithm function, in order to return a value greater than 1 for positive gradients and a value less than 1 for negative gradients:

$$\phi = \begin{cases} -\ln(1 - \nabla) & \text{if } \nabla < 0 \\ \ln(\nabla + 1) & \text{otherwise.} \end{cases} \tag{9}$$

We call this value the *shock indicator*, ϕ. Values of $\phi > 1$ indicate prices in the market may increase; values of $\phi < -1$ indicate prices in the market may fall.

ASAD agents use ϕ to alter profit margin according to the following two rules:

```
if (seller & phi > 1) increase profit margin,
if (buyer & phi < -1) increase profit margin.
```

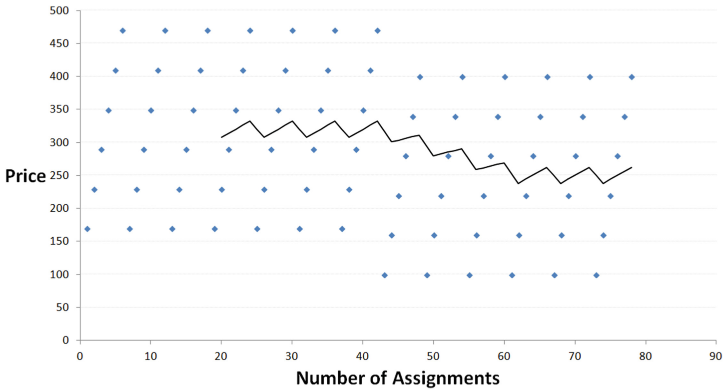

Fig. 9. Illustrative example of an agent's assignment sequence, subject to a *negative* market shock after assignment 42. Blue dots plot the limit price of each assignment. The black line plots a moving average over 20 assignments (the ASAD agent memory window). Ordinary Least Squares (OLS) regression is used to calculate the grandient of change in assignment prices (i.e., the gradient of the moving average), which between assignments 43 and 61 is significantly negative, indicating a negative market shock (Color figure online).

While $\phi > 1$ for sellers (or $\phi < -1$ for buyers), agent calculated quotes are increased, or *inflated*, by 20 %. To prevent ASAD agents from returning to market clearing price (P_0') too early after a shock is detected, the cumulative value of ϕ is used to "wind-down" ASAD price inflation from 20 % to 0 % over time. This decline in percentage over time is proportional to the size of the cumulative value of ϕ, reduced (increased) by 0.5 every time the ASAD agent can update its orders (subject to no current shock occurring), until cumulative ϕ, and therefore percentage, equals zero.

Results from one homogeneous market containing ASAD agents is shown in Fig. 10. We see that immediately following a positive market shock prices begin to rise. Prices then overshoot the new equilibrium value, before returning to near-equilibrium value. This suggests that ASAD agents are sensitive to market shocks, but require tuning. In homogeneous markets with all ASAD agents, sellers benefit from a positive market shock, being able to either match or beat buyers' average profit. This is in stark contrast to ZIP_M markets, where sellers consistently lose out by a margin of $\approx 25\,\%$. Further, very little profit is lost in the market itself, suggesting that assignment adaptation can equalise profit between buyers and sellers during a market shock.

However, when testing ASAD (adapted ZIP_M) agents in positive shock markets containing naïve ZIP_M agents, results were somewhat surprising:

- In heterogeneous markets containing six ASAD and six ZIP_M agents, ASAD sellers performed significantly worse than ZIP_M sellers. Surprisingly, ZIP_M sellers also outperformed all buyers.

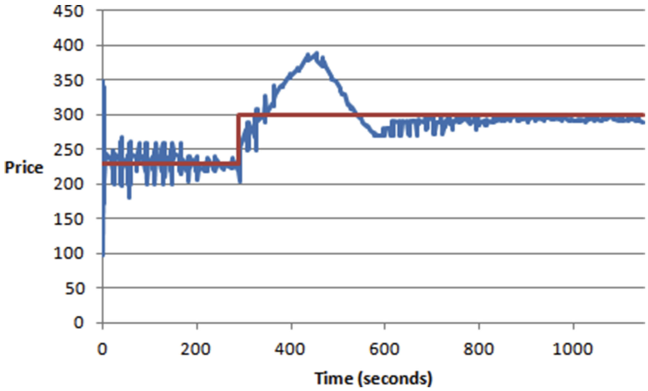

Fig. 10. Example of a *positive* market shock in a homogeneous ASAD market. The market quickly reacts to the shock, but initially overshoots the new equilibrium.

- In heterogeneous markets containing eleven ZIP_M agents and only one ASAD seller, once again the profits of every ZIP_M seller was increased, while the ASAD agent significantly under-performed.
- The profit spread between buyers and sellers of homogeneous markets containing twelve ZIP_M agents was significantly higher than in markets containing at least one ASAD agent; although in every case the ASAD agent(s) suffered.

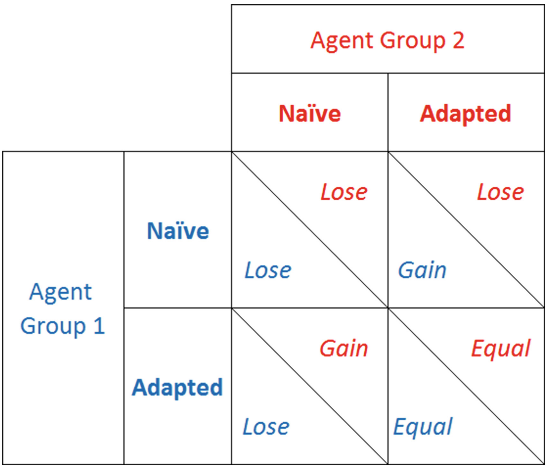

Fig. 11. Normal form matrix of results between competing ASAD (adapted) and non-ASAD (naïve) agents. Homogeneous markets of adapted (ASAD) agents perform better than homogeneous markets of naïve (ZIP_M) agents. However, in heterogeneous markets, naïve (ZIP_M) agents gain while adapted (ASAD) agents lose.

These findings suggest that ASAD agents generate a new price signal to which price sensitive ZIP_M agents can react and benefit. However, ASAD agents themselves suffer from the resulting behaviour of ZIP_M agents. If we consider longer-term market evolution, a population of ASAD agents can be easily invaded by ZIP_M. If the entire market is ASAD then everyone benefits, but if any non-ASAD agent enters the market, it parasitically benefits from the behaviour of ASAD and will flourish, eventually exterminating the ASAD agents from the marketplace. We summarise these outcomes in Fig. 11. Although these results may appear counter-intuitive, such dynamics are not unusual in co-adaptive systems of competing populations (for example, see [1,3,4]).

Potentially, these findings could be due to the simple ASAD strategy implemented here. For example, ASAD agents are not designed to consider the rate of change of prices in the market. Perhaps a more suitable approach would be to implement an adaptive learning rule, such as the Widrow-Hoff delta rule [28], which is the basis of the adaptation mechanism in ZIP [8] and AA [26]. We reserve this extension for future work.

5 Conclusion

We have used the Exchange Portal (ExPo) platform to perform a series of agent-based computational economics experiments between populations of financial trading agents, using continuous replenishment of order assignments.

In the first set of experiments, we exposed several idiosyncrasies and ambiguities in AA and ZIP, two of the standard "reference" algorithms from the literature. First, we showed that ZIP performs better in fast markets when agents contain a vector of profit margins that they can update simultaneously. Then, for AA agents, we demonstrated how P_{max} provides unfair information about the market and how the algorithm can use readily available information to overcome this. Finally, we demonstrated how "spread jumping" in AA negatively affects market dynamics and performance.

In the second set of experiments, we introduced market "shocks" and presented a novel exploratory Assignment Adaptation (ASAD) modification to ZIP. Results showed that homogeneous populations of ASAD agents perform better than homogeneous populations of ZIP agents. However, in heterogeneous ASAD-ZIP populations, ZIP agents perform better while ASAD agents perform worse. This suggests that ASAD agents provide a novel price signal that benefits ZIP, to the detriment of ASAD agents themselves.

This work naturally suggests further extensions. Firstly, to expose the benefits of dynamically selecting a value of P_{max}, we set $P_{max} = 2 \times \max(limitPrice)$. The multiplier value, 2, was arbitrarily selected and should be optimised for performance. Secondly, it is likely that the introduction of an adaptive learning algorithm (similar to that used by ZIP) could improve the performance of ASAD. Thirdly, unlike ZIP agents, AA agents have never been adapted to contain a vector of profit margins that they can update simultaneously (i.e., an AA_M). We reserve these avenues of research for further work.

Perhaps more interestingly, we also reserve more general open questions for future exploration. Firstly, in the work presented here all market shocks are exogenous. It would be very interesting to see how results are affected when shocks are endogenous to the market. However, to answer this, it is first necessary to have agents acting as "proprietary" ("prop") traders—buying and selling on their own behalf for profit—rather than "sales" traders (trading on behalf of a client). This is a more difficult challenge, but one that is pertinent if we are to further our understanding of the global financial markets. Secondly, since real financial markets include human traders and "robot" automated trading agent systems, we hope to explore the dynamic interactions between these groups by introducing human participants into our experiments. ExPo has been specifically designed to enable human participation; and further, since ExPo participants (whether human, or robot) connect to the exchange using HTTP messaging across a network, ExPo allows geographically dis-located human participants to sign in via a web browser and then leave or return at will. Theoretically, this enables us to run experiments with large numbers of participants, over long time periods of days, weeks, or even months. As far as we are aware, this has never been done before and has the potential to provide valuable insight into real world financial markets.

Acknowledgments. The authors would like to thank Tomas Gražys for significant development of the ExPo platform. John Cartlidge is supported by EPSRC grant, number EP/H042644/1; primary financial support for Dave Cliff's research comes from EPSRC grant, number EP/F001096/1.

References

1. Cartlidge, J., Ait-Boudaoud, D.: Autonomous virulence adaptation improves coevolutionary optimisation. IEEE Trans. Evol. Comput. **15**(2), 215–229 (2011)
2. Cartlidge, J.: Trading experiments using financial agents in a simulated cloud computing commodity market. In: Duval, B., van den Herik, J., Loiseau, S., Filipe, J. (eds.) 6th International Conference on Agents and Artificial Intelligent, Agents (ICAART-2014), vol. 2, pp. 311–317. SciTePress, March 2014
3. Cartlidge, J., Bullock, S.: Caring versus sharing: how to maintain engagement and diversity in coevolving populations. In: Banzhaf, W., Ziegler, J., Christaller, T., Dittrich, P., Kim, J.T. (eds.) ECAL 2003. LNCS (LNAI), vol. 2801, pp. 299–308. Springer, Heidelberg (2003)
4. Cartlidge, J., Bullock, S.: Unpicking tartan CIAO plots: understanding irregular coevolutionary cycling. Adapt. Behav. **12**(2), 69–92 (2004)
5. Cartlidge, J., Cliff, D.: Exploring the "robot phase transition" in experimental human-algorithmic markets. The Future of Computer Trading in Financial Markets-Foresight Driver Review-DR25, Crown Copyright, Oct 2012. http://bitly.com/SvqohP
6. Cartlidge, J., Cliff, D.: Evidencing the "robot phase transition" in experimental human-algorithmic markets. In: Filipe, J., Fred, A. (eds.) 5th International Conference on Agents and Artificial Intelligent, Agents (ICAART-2013), vol. 1, pp. 345–352. SciTePress, Feb 2013

7. Cartlidge, J., Szostek, C., De Luca, M., Cliff, D.: Too fast too furious: faster financial-market trading agents can give less efficient markets. In: Filipe, J., Fred, A. (eds.) 4th International Conference on Agents and Artificial Intelligent, Agents (ICAART-2012), vol. 2, pp. 126–135. SciTePress, Feb 2012
8. Cliff, D.: Minimal-intelligence agents for bargaining behaviors in market-based environments. Technical report, HPL-97-91, Hewlett-Packard Labs, Aug 1997. http://bit.ly/18uC9vM
9. Das, R., Hanson, J., Kephart, J., Tesauro, G.: Agent-human interactions in the continuous double auction. In: Nebel, B. (ed.) 17th International Joint Conference on Artificial Intelligent (IJCAI-01), pp. 1169–1176. Morgan Kaufmann, Aug 2001
10. De Luca, M., Cliff, D.: Agent-human interactions in the continuous double auction, redux: using the OpEx lab-in-a-box to explore ZIP and GDX. In: Filipe, J., Fred, A. (eds.) 3rd International Conference on Agents and Artificial Intelligent (ICAART-2011), pp. 351–358. SciTePress, Jan 2011
11. De Luca, M., Cliff, D.: Human-agent auction interactions: adaptive-aggressive agents dominate. In: Walsh, T. (ed.) 22nd International Joint Conference on Artificial Intelligent (IJCAI-11), pp. 178–185. AAAI Press, Jul 2011
12. De Luca, M., Szostek, C., Cartlidge, J., Cliff, D.: Studies of interactions between human traders and algorithmic trading systems. The Future of Computer Trading in Financial Markets-Foresight Driver Review-DR13, Crown Copyright, Sep 2011. http://bitly.com/RoifIu
13. ExPo: The Exchange Portal, Mar 2012. http://sourceforge.net/projects/exchangeportal/
14. Feltovich, N.: Nonparametric tests of differences in medians: comparison of the wilcoxon-mann-whitney and robust rank-order tests. Exp. Econ. **6**, 273–297 (2003)
15. Feltovich, N.: Critical values for the robust rank-order test. Commun. Stat. Simul. Comput. **34**(3), 525–547 (2005)
16. Gjerstad, S., Dickhaut, J.: Price formation in double auctions. Games Econ. Behav. **22**(1), 1–29 (1998)
17. Gode, D., Sunder, S.: Allocative efficiency of markets with zero-intelligence traders: markets as a partial substitute for individual rationality. J. Polit. Econ. **101**(1), 119–137 (1993)
18. OpEx: Open Exchange software, Mar 2012. https://sourceforge.net/projects/open-exchange/
19. Preist, C., van Tol, M.: Adaptive agents in a persistent shout double auction. In: 1st International Conference on Information and Computation Economies, pp. 11–18. ACM Press (1998)
20. Smith, V.: An experimental study of comparative market behavior. J. Polit. Econ. **70**, 111–137 (1962)
21. Stock, J.H., Watson, M.M.: Introduction to Econometrics, Chap. 4, 3rd edn. Pearson, Upper Saddle River (2012)
22. Stotter, S.: Improving the strategies of algorithmic traders and investigating further realism in their market environment. Master's thesis, Department of Computer Science: University of Bristol, UK, July 2012
23. Stotter, S., Cartlidge, J., Cliff, D.: Exploring assignment-adaptive (ASAD) trading agents in financial market experiments. In: Filipe, J., Fred, A.L.N. (eds.) 5th International Conference on Agents and Artificial Intelligent, Agents (ICAART-2013), vol. 1, pp. 77–88. SciTePress, Feb 2013
24. Tesauro, G., Das, R.: High-performance bidding agents for the continuous double auction. In: ACM Conference on Electronic Commerce, pp. 206–209. ACM Press (2001)

25. van Montfort, G.P.R., Bruten, J., Rothkrantz, L.: Arbitrageurs in segmented markets. Technical report, HPL-97-120, Hewlett-Packard Labs, Oct 1997. http://www.hpl.hp.com/techreports/97/HPL-97-120.pdf
26. Vytelingum, P.: The structure and behaviour of the continuous double auction. Ph.D. thesis, School of Electronics and Computer Science, University of Southampton, UK (2006)
27. Vytelingum, P., Dash, R.K., David, E., Jennings, N.R.: A risk-based bidding strategy for continuous double auctions. In: López de Mánataras, R., Saitta, L. (eds.) 16th European Conference on Artificial Intelligence (ECAI-2004), pp. 79–83. IOS Press (2004)
28. Widrow, B., Hoff, Jr., M.E.: Adaptive switching circuits. In: Institute of Radio Engineers, Western Electron, Show and Convention (IRE WESCON), Convention Record, Part 4, pp. 96–104, Aug 1960