Government Office for Science

Foresight

Studies of interactions between human traders and Algorithmic Trading Systems

The Future of Computer Trading in Financial Markets - Foresight Driver Review – DR 13

Contents

Studies of Interactions Between Human Traders and Algorithmic Trading Systems

Marco De Luca, Charlotte Szostek, John Cartlidge, & Dave Cliff [1]

This review has been commissioned as part of the UK Government's Foresight Project, The Future of Computer Trading in Financial Markets. The views expressed do not represent the policy of any Government or organisation.

[1] All authors are members of the Department of Computer Science, University of Bristol, Bristol BS8 1UB, UK. Emails: delucamarco@gmail.com; Charlotte.Szostek.09@bristol.ac.uk; john@john-cartlidge.co.uk; and dc@cs.bris.ac.uk. This document is Copyright © 2011 by M. De Luca, C. Szostek, J. Cartlidge, & D. Cliff; all rights reserved.

Abstract

In recent years there has been a very significant increase in the percentage of trades in the global financial markets that are initiated and executed by automated "robot" algorithmic trading software systems, autonomously performing trading roles that a decade or more ago would have been performed by human traders. The anonymity of many current electronic trading systems, operated by major exchanges and multilateral trading facilities, mean that an individual trader, whether human or robot, never knows[2] if the counterparty to a particular trade is a human or not. There are commonly-quoted estimates that the proportion of robot-executed trades is approaching 30%-70% on major European and US equity exchanges. In foreign-exchange markets, where there are no central exchanges, the proportion of spot (immediate-execution) transactions that are executed by robots is widely believed to be even higher. From this, it is clear that the current global financial markets involve a very significant degree of interaction between human and robot traders.

The interactions between human traders in electronic markets has long been studied in the field known as *Experimental Economics,* and more recently the interactions between software-agent traders in electronic markets has been the topic of various abstract research studies in so-called *Agent-based Computational Economics* (ACE). These two research fields are largely distinct: the first studies markets populated entirely by human traders; the second studies markets populated entirely by algorithmic software-agent traders. There is a surprising lack of studies of the interactions between human traders and robot traders. That is, there is very little scientific literature that explores *heterogeneous* markets, populated by *both* humans *and* robots.

In this document we review the very small amount of published literature that does describe scientific studies of interactions between human and robot traders under experimental conditions. We contend that the relative lack of such studies is a serious omission from the literature. We propose that De Luca's (2010) *Open Exchange* (OpEx) open-source design for studying human-robot interactions in electronic marketplaces should be used as a free *de facto* standard for future work in this area. We illustrate the use of OpEx by summarising recently published peer-reviewed accounts of early experiments with OpEx, and then present a detailed description and analysis of results from some new experiments, conducted specifically for this review document, where we relax some of the artificial experimental constraints that have been used in earlier work.

Experiments with the OpEx system indicate that the previously reported outperformance of the algorithmic trading systems over humans may well be related to the artificial nature of the experiment design that was employed in the earlier research: a design essentially unchanged since the first experimental economics results were published in the early 1960's. When the flow of orders in the market was trickled in gradually (rather than all orders being released simultaneously, which was an artificial constraint in the designs of the earlier experiments) the

[2] Anonymity is desirable for all Dealer-to-Dealer (D2D) electronic markets. Most financial products are traded electronically on D2D exchanges, including: equities, fixed income products, currency, commodities, and their vanilla derivatives such as futures and options. On the other hand, participants to Dealer-to-Customer (D2C) markets undergo a negotiation mechanism known as Request for Quote (RFQ), whereby two known counterparties privately negotiate a deal. The human component in D2C markets is predominant, as computers are only used for off-line calculations (as opposed to real-time): for this reason, D2C markets are out of the scope of our study.

performance of the software agents was no longer so clearly superior. Furthermore, when the software agents were slowed to operate on the same sort of timescales that human traders act on, the data we have thus far indicates that the market dynamics alter, but further experiments would be required to establish the significance of this with appropriate levels of certainty.

1. Introduction

In recent years there has been a very significant increase in the percentage of trades in the global financial markets that are initiated and executed by automated "robot" algorithmic trading software systems, autonomously performing trading roles that a decade or more ago would have been performed by human traders. Furthermore, the anonymity of many current electronic trading systems, operated by major exchanges and multilateral trading facilities (MTFs), mean that an individual trader, whether human or robot, never knows if the counterparty to a particular trade is a human or not. There are commonly-quoted estimates that the proportion of robot-executed trades is approaching 50% on major European equity exchanges; and is nearer 75% on major US equity exchanges. In foreign-exchange markets, where there are no centralised national exchanges, the proportion of spot transactions that are executed by robots is widely believed to be even higher. From this, it is clear that the current global financial markets involve a very significant degree of interaction between human and robot traders.

The interactions between human traders in electronic markets has long been studied in the field known as *Experimental Economics,* a field pioneered in the 1960's by Vernon Smith, for which he was awarded the 2002 Nobel Prize in Economics. More recently, the interactions between trading strategies embodied as "robot" software, in simulated electronic markets, has been the topic of various research studies in so-called *Agent-based Computational Economics* (ACE). Despite the existence of these two research fields, the one that studies homogeneously human markets and the other that studies homogeneously robot markets, there is a surprising lack of studies of the interactions between human traders and robot traders. That is, there is very little scientific literature that explores *heterogeneous* markets, populated by *both* humans *and* algorithmic systems.

In Section 2 of this report we provide as background a brief summary of work in experimental economics and ACE that are relevant here; in Section 3 we then go on to describe in more detail the very few papers that we have found that explicitly address studies of human traders interacting with robot traders under controlled experimental conditions.

We argue here that the relative lack of such studies is a serious omission from the literature. The reasons why the study of human-robot trading interactions has been so overlooked for so long is something that we can only speculate about. It seems to us that one likely issue is that the intellectual focus of mainstream economics has long been exclusively on the idealised *homo economicus* – it is only comparatively recently that economists have become interested in "behavioural economics" (i.e., how humans actually behave with respect to economic activity, rather than how they would behave if they were perfect). Similarly, the focus in ACE seems for almost all of its history to have been on using software agents or "robots" as proxies for human subjects (rather than as first-class autonomous trading entities in their own right). So, in this sense at least, the lack of relevant literature may simply be a straightforward reflection of the fact that the question of how robot and human traders interact has for a very long time simply not been of much interest at all to researchers either in experimental economics or in ACE.

Another issue that we think may have played a role is that for a long time it was believed that to perform appropriate studies required prohibitively costly investment in experimental facilities, i.e. that the cost of the apparatus for performing controlled experiments involving humans interacting on artificial electronic markets was just too high. This may have been true one or two decades ago, but the ongoing exponential reductions in the real costs of computer and networking equipment mean that nowadays it is possible to conduct such experiments using hardware that costs only a few thousand pounds. Of course, buying the hardware is only half the story: there is also a need for appropriate software. Probably for many potential experimenters, the absence of any appropriate software was the bigger problem: armed with a big enough budget, anyone can buy a roomful of computers and some network gear to connect them together, but knowing how to write software for what is, in essence, a real-time mini stock-exchange, is another matter altogether.

In 2010, recognising this need, one of us (De Luca) developed the necessary software, known as *Open Exchange* (OpEx), and full details of the design (including the source-code) is scheduled to be released as open-source in the near future.[3] The intent of releasing OpEx as open-source is that it should hopefully then become the free *de facto* standard for future work in this area. Section 4 of this document describes the design and architecture of OpEx in some detail.

In Section 5 we illustrate the use of OpEx by presenting a description and analysis of results from recent new experiments, conducted specifically for this review document, where we relax some of the artificial experimental constraints that have been used in earlier work. Results from our new experiments with OpEx (given in full in Appendix A) indicate that the previously reported outperformance of the algorithmic trading systems over humans are primarily speed-related, and may also be related to the artificial nature of the experiment design.

2. Background

2.1 Experimental economics

In 1962, a paper was published in the premier-league *Journal of Political Economy* by Vernon Smith, an academic economist at Purdue University (Smith, 1962). There, Smith described results from a series of laboratory experiments studying human traders interacting in a market, research that he had commenced in the late 1950's. Smith's 1962 paper was seminal in establishing the field now known as *experimental economics*, and in 2002 he was awarded the Nobel Prize in Economics[4] for his distinguished work in this field. It is useful to review the experiment methods that Smith introduced in his 1962 paper because they continue to have significant influence, half a century later.

Smith was interested in studying mechanisms by which buyers and sellers can come together to agree prices for transactions. In the economics literature, this mechanism is referred to as an *auction*, and there are many different types of auction. One type, seen in most high-street

[3] The costs of buying the OpEx hardware and developing the software were met from funds made available to Dave Cliff by the UK Engineering and Physical Sciences Research Council (EPSRC), grant number EP/I001603/1.

[4] Strictly speaking, there is no Nobel Prize in Economics. The Nobel Foundation has, since 1901, awarded Nobel Prizes in Physics, Chemistry, Medicine, Literature, and Peace. Since 1968, after a donation from Sweden's central bank, it has also awarded what is currently officially called the Sveriges Riksbank Prize in Economic Sciences in Memory of Alfred Nobel. This is commonly referred to as the Nobel Prize in Economics. Especially by economists.

stores, is the *posted offer auction*: sellers post an “offer price” or “asking price”, and the buyers either take it or leave it. Users of the eBay.com online auction site engage in either posted-offer auctions (known in eBay’s terminology as “Buy It Now”) for immediate execution, or they engage in an online process mediated by eBay that implements a *second-price open-bid auction* where an end-time for the auction is specified, and buyers can announce bid-prices at any time until the auction ends, with those bids being visible to all participants in the auction (“open-bid”) and the item being sold to the buyer who posted the highest bid-price, but the transaction price being set by the value of the second-highest bid (“second-price”). This is a slight (but significant) variation on the auction mechanism that is used in sales of fine art and antiques, among other things: there, most often under the direction of an auctioneer, the buyers announce increasing bid-prices until only one buyer remains, and that buyer then takes the deal: technically this is a *first-price open-bid auction*, but it is colloquially known as the *English auction* mechanism because, in the UK at least, the word “auction” was for many years synonymous with just this one mechanism. In contrast, in the Netherlands, the auction mechanism used in the markets for tulip and daffodil flower-bulbs (for centuries a major sector in the Dutch economy) is essentially the inverse of the English auction: in what is widely referred to as a *Dutch flower auction*, the buyers stay silent and the sellers announce decreasing offer-prices, until a buyer announces that they are willing to transact at the current price. The list of auction types is long, but there is one particular type of auction that the rest of this review will concentrate on, and indeed is the type of auction that Smith first studied. This is known as the *continuous double auction* (CDA).

The CDA can be thought of as the superposition of the English auction and Dutch flower auction, running simultaneously in the same space, and asynchronously -- without any auctioneer. In the CDA a buyer is free to announce a bid price at any time, and there is often an expectation (or even a rule) that the bid-prices should increase over time; and at the same time any seller is free to announce an offer-price (commonly referred to as an “ask”) at any time, with the expectation or rule that the offer-prices should decrease over time. Also, at any time, any buyer is free to accept any seller’s offer (but would typically only be interested in the current best offer) and any seller is free to accept any buyer’s bid, again with the primary focus being on the current best bid. There may be a clock running, and trading may start and end at specified times, but in general the “continuous” nature of the CDA means that buyers and sellers may join, act in, or leave the market at any time and that their actions are not coordinated by an auctioneer. The CDA is of great practical interest because it is the auction mechanism employed by almost every major financial market on the planet. Largely for that reason, Vernon Smith’s earliest experiments set out to explore the dynamics of CDA markets, populated by human traders, under experimental laboratory-style conditions.

To do this, Smith reduced the complexity of the real-world CDA to the simplest, most minimal instantiation that he could conceive of. He recruited a set of willing human subjects (typically undergraduate students, sometimes also students on postgraduate courses), and randomly assigned them to be either “buyers” or “sellers”. To each of the sellers he gave one item of “stock” to sell: the thing they were handed had no real value (for the sake of argument, think of it as a matchstick). To each of the buyers he gave an amount of money (again, the money could have been valueless play-money from a board-game like *Monopoly*). The amount of money given to each buyer was private, a secret, known only to Smith and that buyer; the buyer could not spend money that they had not been given, and so in this way Smith could control the *limit price* for each buyer, i.e. the price above which they could not purchase an item of stock. Similarly, he privately told each seller their secret limit price: the price below which they should not sell their item of stock. All of the experiment’s traders, buyers and sellers, were then told that they should try to make a deal, but that a clock would be running for a period that Smith referred to as an experimental “trading day” but which in practice would last for 5 to 10 minutes, or would be ended when no-one remaining in the experiment wanted to trade. The subjects were informed that for each of them if they had not entered into a transaction by the time the "day" ended, they would receive no reward; but if an individual trader had made a deal, their reward would be determined by the difference between the transaction price and that trader's limit price – their *utility* gained, i.e. the "profit" for a seller or the "saving" for a buyer. So, for example, if a buyer and a seller had limit prices of $1.50 and $0.20 respectively, and agreed to transact at a price of $1.00, the buyer's reward would be proportional to her $0.50 saving, and the seller's reward would be proportional to his $0.80 profit (so in this case the seller has greater utility, having done better out of the deal than the buyer). At the end of each "day", all unused assignments of stock and money would be cleared: nothing was carried over to the next period.

Having made these preparations, and having given the subjects appropriate instructions, the clock was started and the subjects were told to interact under the rules of the CDA: the traders could announce a "quote" (a bid or an offer) at any time. Smith and his assistants observed what happened and noted the time of each quote, who made it, what its value was, and who, if anyone, accepted that quote. As each trader was only given enough stock or currency to enter into one transaction, and then could do nothing, each trading day did not last more than a few minutes. But Smith and his assistants would then re-allocate stock and currency to the traders, and declare the market to be open for another "day" of trading, which was monitored in the same way as the first. They would then repeat this process for several consecutive "days" – typically 10 or fewer.

As Smith had control over the limit prices given to the traders, he could control the shapes of the market's *supply and demand schedules*. The market's supply schedule is a specification of how the total quantity supplied by producers of some item varies as a function of the price of that item: most often, the higher the price that buyers are willing to pay, the greater the quantity that suppliers are willing to offer for sale. Similarly, the demand schedule specifies the relationship between the number of items demanded by buyers, again as a function of price; and, intuitively, most often as the price of an item increases, so there are fewer buyers willing to pay the higher price. On a graph showing price on the horizontal axis, and quantity on the vertical axis, the supply curve will slope upwards and the demand curve will slope downwards, as shown in Figure 2.1.

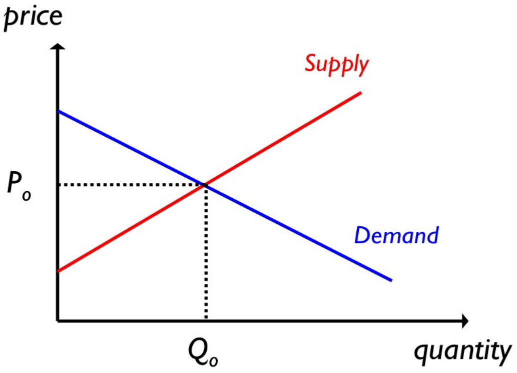


Figure 2.1: Supply and Demand curves (here illustrated as straight lines) relate the quantities supplied by sellers and demanded by buyers, respectively, to the price per item: as the price increases, the quantity supplied increases but the quantity demanded falls. The point at which the two curves intersect is the theoretical equilibrium point for this supply and demand schedule: *Q_0* is the equilibrium quantity and *P_0* is the equilibrium price.

As further illustration, consider a hypothetical example where, in one of Smith's experiments, there are five buyers, denoted by letters *A* to *E*, and they have been allocated limit prices of $0.50, $1.00, $,1.50, $2.00, and $2.50, respectively. As no buyer can pay more than $2.50, the quantity demanded is zero above $2.50. If the price is in the range $2.01 to $2.50, there is just one buyer (*A*) who is able to transact in that price range, so the quantity demanded is 1 over that range. At prices in the range $1.51 to $2.00, both *A* and *B* are able to transact, and so the quantity demanded is 2 over that range. As the price goes lower and lower, more buyers are able to transact and hence the quantity demanded in the market increases, up to the point where, for the price range $0.00 to $0.50, all five buyers are able to transact. We can plot the demand curve for this experimental market as shown in Figure 2.2.

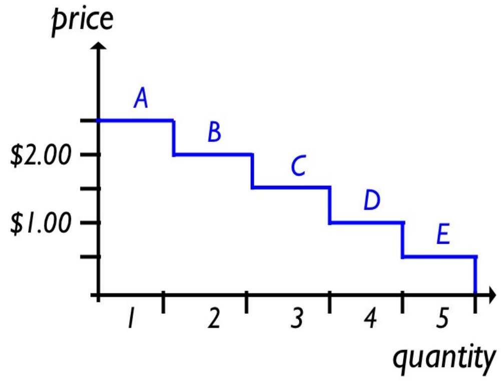


Figure 2.2. A "stepped" demand-curve for a market with just five buyers, each able to purchase only one unit. See text for discussion.

Referring back to Figure 2.1, it can be seen that the supply and demand curves intersect at a point marked (Q_0, P_0): this is known as the market's *equilibrium point*, and the corresponding values P_0 and Q_0 are the *equilibrium price* and *equilibrium quantity*, respectively. This point on the graph is of great significance, because if transactions occur at the equilibrium price, then the allocation of scarce stock from sellers to buyers can be e*fficient*, for a specific technical notion of efficiency. A common ideal of efficient allocation is the notion of *Pareto efficiency.* An allocation is Pareto efficient if no-one can be made better-off without someone else being made worse-off. Pareto efficient allocations can arise from free markets despite the fact that each trader in the market is competing, acting only to serve his or her self-interest: the traders appear to be led to an efficient allocation by an "invisible hand". Markets are not guaranteed to always achieve optimal allocations (conditions in which they fail are well known), but one of the primary reasons why market economies excite such interest is their ability to *self-equilibrate*. That is, traders interacting via an appropriate auction mechanism and acting only in their own rational self-interest can collectively discover the equilibrium point. Thus it is reasonable to talk of the equilibration behaviour of a collection of traders interacting in some market mechanism as an "emergent behaviour" of the market.

To illustrate this, consider a situation where transactions are consistently taking place at above-equilibrium prices. At these higher prices, the quantity demanded Q_D will be less than the equilibrium quantity Q_0, and yet the quantity supplied Q_S will be greater than Q_0. That is, $Q_D < Q_0 < Q_S$, and we can then expect competition among the sellers to result in reduction in their offer-prices; as the prices fall, so Q_S is reduced and Q_D increased, until eventually $Q_D = Q_0 = Q_S$. Similarly, if ever transactions were consistently taking place at prices below the equilibrium price, we would have $Q_D > Q_0 > Q_S$ and competition among buyers would push bid-prices up, reducing Q_D and increasing Q_S, and hence leading prices back to the equilibrium point where $Q_D = Q_0 = Q_S$. Because the equilibrium is achieved as a result of competition among buyers pushing prices up, and/or competition among sellers pushing prices down, the balance-point is a *competitive equilibrium*, which should be maintained until the market's supply or demand schedules change. Of course, this is an idealised and simplistic argument. In reality there may be delays and sources of noise or error within the market system, meaning that perfect equilibrium is never reached, and the system may spend considerable periods of time in off-equilibrium states. Whether a particular market system reaches equilibrium, and if so how, were the type of questions that Smith explored in his seminal 1962 paper.

Smith's paper explored the relationship between the nature of the supply and demand curves in his experimental markets, and the market's equilibration behaviour. Typically, in any one of his experiments, transaction prices in the first "day" were some way distant from the equilibrium price, but in subsequent trading periods the prices would show some degree of convergence to the equilibrium. To quantify the degree of convergence, Smith introduced a "coefficient of convergence", □, calculated for each trading period. This is defined as $\square=100(\square_o/P_0)$ where $\square_o$ is the root mean square (RMS) of the differences between the transaction prices in the period and the theoretical equilibrium price P_0 ($\square_o$ is manifestly analogous to the commonplace statistic the *standard deviation*, which is RMS deviation of samples from the mean sample-value); hence □ can be thought of as measure of price variability about the equilibrium price, expressed as a percentage of that equilibrium price. Smith also monitored how efficient the allocations were in his experimental markets, by calculating a measure called the *allocative efficienc*y of the market, defined as the total utility actually earned by all the traders in a period, expressed as a percentage of the total combined maximum utility that could in principle be earned by all the traders: typically after one or two periods, human traders achieve allocative efficiency scores very close to 100%.

In the first eight experiments reported by Smith (1962), each trader was allowed to buy or sell only one unit of stock,[5] although in later experiments that constraint was relaxed. Smith also experimented with changing the supply and demand curves during the experiment (i.e., after the subjects had been given a few days to get accustomed to one supply/demand schedule, at the start of the next day's trading a new set of limit prices, representing a new schedule, would be distributed to them), and with having the buyers remain silent while only the sellers could quote offers, i.e. a switch from a CDA mechanism to a posted-offer auction.

Smith's results were, at the time, bordering on the revolutionary. They clearly established that markets with very small numbers of inexperienced traders could, with very little opportunity for learning, equilibrate to the underlying theoretical equilibrium point predicted from classical economic theory. He also showed that aspects of the supply and demand schedule could determine whether equilibrium was approached from above or below, and whether the predicted P_0 was reached or the traders stabilised at some off-equilibrium value. For further discussion of the significance of the results in Smith's first paper, see Cliff (1997, pp.17-21). Figure 2.3, reproduced from Cliff (1997), shows the data from Smith's first reported experiment, using the visualisation method that Smith introduced and which has since become something of a *de facto* standard in the literature. On the left-hand side of the chart is a representation of the supply and demand curves, with dotted lines indicating the values of Q_0 (6) and P_0 ($2.00). On the right-hand side is a trace of the transaction prices recorded each trading period (or

[5] Technically, the "stock" in experiments such as these is a unit of an arbitrary abstract *commodity*, because the items being traded have no qualities other than price by which they can be distinguished in the market.

"day" in Smith's terminology), with the numbers running along the bottom showing the number of transactions in each period.

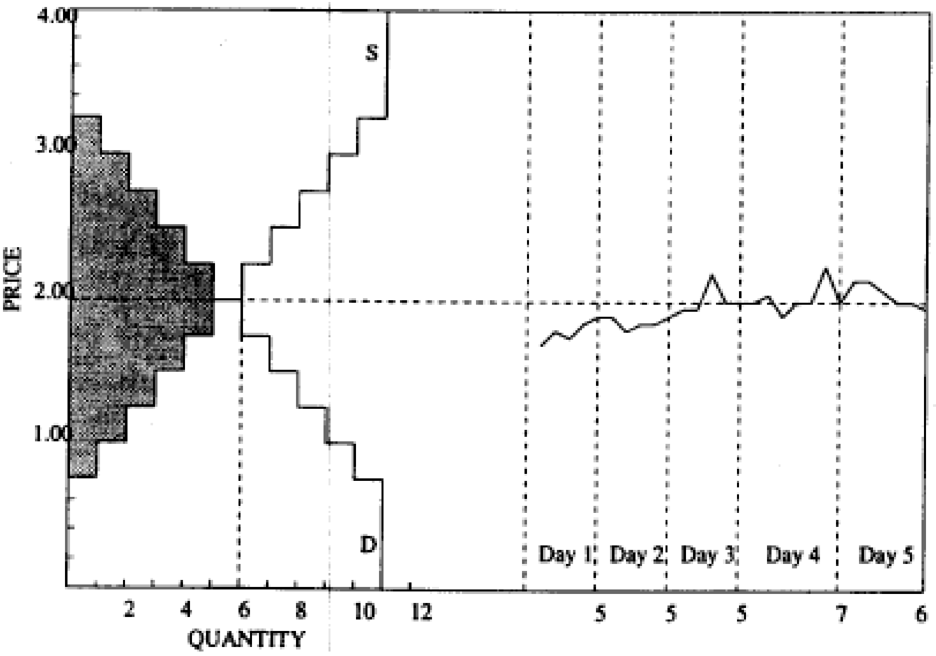


Figure 2.3: redrawn from Smith's (1962) Chart 1: results from the first-ever experimental economics study of traders interacting in the continuous double auction mechanism. This image is reproduced, with the permission of the author, from Figure 6 of (Cliff, 1997). See text for explanation.

Following the publication of Smith's 1962 paper, an entire research field grew around his central idea of conducting laboratory experiments on human traders to better understand the microstructure of market dynamics. Smith continued to introduce innovations in experimental economics throughout his career. Most notably, growing tired of the slow and error-prone nature of having human assistants manually record the sequence of quotes and trades in each market experiment, he devised and implemented a network of computer terminals that the experiment's subjects would be asked to sit at, one terminal per trader, allowing the traders to enter their quotes and accept counterparties for transactions, and via which Smith and his colleagues could control the amount and type of information about current market activity that the traders were presented with. He did this many years before electronic screen-based trading was commonplace in real-world financial markets. Summaries of past and current work in experimental economics are to be found in the following publications: Davis & Holt, 1992; Kagel & Roth, 1995; Guala, 2005; Smith, 2005 & 2006; Plott & Smith, 2008; Bardsley *et al.*, 2009; and Durlauf & Blume, 2009.

When Smith commenced his research, the only economic agents that it was possible to study via experiments were human ones. But, in the last twenty years, autonomous software trader-agents have been developed as proxies for, or replacements of, human traders, and experimental study of markets populated by such "robot" traders is now a relatively mature field, as we shall see in the next section.

2.2 Agent-Based Systems

There are two communities of researchers who have been active in the study of markets populated by autonomous software trader-agents: economists, and computer scientists. It is beyond the scope of this paper to attempt a detailed historical analysis of how research interests in these two fields converged and then came to overlap, intersect, and cross-fertilise over the course of the last twenty years. The story we tell here will, of necessity, be something of a caricature, but we believe it to be accurate in spirit even if it does skim over an awful lot of detail.

In brief, we will note here simply that in computer science, in the late 1980's and early 1990's, researchers interested in engineering artificial intelligence (AI) systems started to talk in terms of working on *autonomous agents*: self-contained entities that could sense their environment and somehow combine those sensor readings with their internal state (possibly via learning and/or reasoning mechanisms) to take actions that are appropriate to that environment, in the pursuit of the agent's "aims" or "goals", without the need for external control. This definition fits a wide range of entities: animals are autonomous agents; and a mobile robot can be an autonomous agent too, if it is sufficiently independent of human control. Both animals and robots typically require sensors (eyes, ears, cameras, microphones) to detect aspects of the physical environment they find themselves in, and actuators (muscles, motors) to take appropriate actions with. To ensure coherent coordination between the sensors and the actuators (also referred to as sensory-motor co-ordination), physical agents typically have a specific embodiment "frame" (skeleton, chassis).

But not all environments of interest are physical ones: some are entirely virtual, and although the autonomous agents that inhabit virtual environments are disembodied entities they still need to engage in coordinated sensing and acting. As is clear in the present-day financial markets, an autonomous software trader (commonly referred to as a "robot trader") can exist in disembodied form, "sensing" various dynamic streams of market information, integrating that changing sensory data with its internal state (which may involve historical data), and taking appropriate actions such as issuing buy or sell orders into the market, or cancelling existing orders. For overviews of autonomous agent research in AI, see Maes (1990, 1994). Wooldridge & Jennings (1995), Jennings *et al.* (1998), Jennings *et al.* (2001), and Sterling & Taveter (2009).

Much relevant work in the AI community has been devoted to the *Trading Agent Competition* (TAC) established in 2000 by Michael Wellman, a professor at the University of Michigan, who devised the original TAC (see e.g. Wellman *et al.*, 2002) and ran it for several years before handing its organisation and administration over to the Swedish Institute for Computer Science (SICS).[6] The original TAC, now discontinued, involved agents acting as personal travel arrangers: the agents were required to negotiate with multiple suppliers for air fares, ground transport, hotels, etc, to assemble a travel package for a client. A subsequent addition to the TAC involved a separate contest framed as a supply-chain management problem, and is known as TAC-SCM. In 2011, TAC involves three contests: TAC-SCM, a TAC for selling online advertising, and a TAC for Market Design (known as "CAT") where the competition is to develop new market mechanisms rather than new trading agents. In all the TAC contests, the focus is very much on software agents interacting with other software agents, rather than on exploring the interaction dynamics of human and automated traders. The dynamics and microstructure of markets involving both human and software-agent traders interacting with one another is, as far as we know, something that has never been studied by the TAC research community. Conspicuous by its absence from the list of TAC contests is any mention of a TAC suited for, or modelled on, trading in real-world financial markets. We do not know why TAC

[6] The web-page for the TAC at SICS is http://www.sics.se/tac/.

has never involved contests directly relevant to current financial markets (as opposed to possible future ones); it seems to us to be an error of omission.

Nevertheless, other AI researchers have addressed issues directly relevant to financial markets. Most notably, Prof. Michael Kearns at the University of Pennsylvania developed a significant simulator, in partnership with Lehman Brothers, called the *Penn-Lehman Automated Trading* (PLAT) Project, which ran 2003-2006 (Kearns & Ortiz, 2003). The PLAT website[7] states: "PLAT is a broad investigation of algorithms and strategies for automated trading in financial markets and related environments. The project makes use of the Penn Exchange Simulator (PXS), a simulator for automated trading that uses real-world, real-time stock market data available over modern Electronic Crossing Networks (ECNs)". Again, the focus here was on research that developed better software-agent traders, and because of the use of real-world market price data, those traders were price-takers only: they could not influence the prices via their own actions. PLAT was not intended for the study of human-agent interactions in market scenarios.

More recently, in 2010 Prof. Philip Treleaven and Dr. Dan Brown at University College London initiated the UCL Algorithmic Trading Competition,[8] a contest in which automated trading systems can compete against one another in an effort to make the most profit while trading (with either real or virtual money) against real-time market data-feeds. The emphasis in these contests is on the participants fine-tuning individual trading algorithms: because each algorithm is a "price-taker", buying or selling at whatever price is shown on the data-feed, and because an individual algorithm's transaction cannot significantly shift the price of the instrument being traded (i.e., there is no "market impact" effect of high-volume transactions), each algorithm can be thought of as trading *in vacuo*: interactions with other traders, be they human or algorithmic, are simply not an issue.

Largely independent of the development of autonomous-agent research in AI, but at roughly the same time, other computer scientists and systems engineers were turning their attention to the problem of controlling or managing large-scale systems in a decentralised fashion. Decentralised control is attractive because it is resilient: there is no single central point of vulnerability, as there is in a traditional centralised control system where loss of the "command center" results in loss of control of the entire system. Traditional centralised control systems also frequently suffer problems as the scale of the system being controlled increases: above some threshold system-size or number of components, centralised control systems often suffer crippling slow-downs or collapses because of the sheer volume of data that needs to be assimilated and digested before appropriate actions can be determined. In 1988, Mark Miller and Eric Drexler published two landmark papers (1988a, 1988b) where they argued for using artificial computational microeconomic systems for decentralised dynamic balancing of supply and demand of scarce resources in a large-scale distributed systems, an approach that subsequently became known as *market-based control* (MBC: see, e.g., Clearwater, 1996). In an MBC system, scarce resources are bought and sold by groups of autonomous software agents acting as buyers (for resource-consumers) or sellers (for resource-providers). Probably the most famous MBC system was developed at Xerox PARC research labs by Clearwater *et al.* (1996), where PARC's traditional air-conditioning control system serving 53 offices was replaced by an MBC system involving software agents buying and selling air in various conditions, and the MBC system was shown to give better distribution of temperatures and use fewer resources. MBC requires the development of trading-agent software (indeed, it was the

[7] http://www.cis.upenn.edu/~mkearns/projects/plat.html.

[8] See http://www.financialcomputing.org/phd-programme/conferences-events.

initial motivation for the ZIP algorithm discussed at length later in this document) but because the aim in MBC is for highly automated, autonomous, resource-allocation and control systems, there is no tradition within the MBC literature of studying the interactions of human and robot traders in electronic marketplaces.

At pretty much the same time as computer scientists were starting to work on autonomous-agent AI systems and on MBC, but almost entirely independently, various academic economists were turning their attention to the use of computer simulations to explore the kind of issues that Smith and other experimental economists had been studying by running lab sessions populated by groups of human traders. At least in part, the motivation for the economists was the fact that running experiments like those that Smith had pioneered was a costly process. Brian Arthur, an economist working at the Santa Fe Institute, proposed in 1993 that "software automata" could be engineered to behave like human economic agents, and explored the use of a simple machine learning algorithm to develop artificial autonomous agents that could be calibrated against human learning data from psychology experiments (Arthur, 1993). Subsequent to this, he and various colleagues collaborated on the *Santa Fe Stock Market,* a simulated CDA-based stock market in which the market dynamics of different interacting automata (i.e., agents) could be studied: see Palmer *et al.* (1994), LeBaron *et al. (*1999), and also Ehrentreich (2007). An international contest with a $10,000 prize was organized at the Santa Fe Institute, where researchers from around the world could submit automata that would then be pitted against each other in a series of trading contests in an attempt to identify the best trading strategy (Rust, Miller, & Palmer, 1993). In much the same way that a similar contest organized years earlier by Robert Axelrod identified a surprisingly simple winning strategy for the iterated *Prisoner's Dilemma* game (Axelrod, 1984 & 1997), the surprise winner of the Santa Fe contest was a surprisingly simple strategy, submitted by Todd Kaplan, which is now widely known as "Kaplan's Sniper".

Kaplan's Sniper is a surprisingly robust and effective trader algorithm which outperformed all the competition at the Santa Fe contest, including more complex trading algorithms that used traditional optimization approaches, statistical predictions of future transaction prices, and/or machine learning algorithms. The Sniper strategy is remarkably easy to explain: it sits quietly on the sidelines doing nothing at all, merely observing the market, until the difference between the best bid and the best offer (the *bid-offer spread*) drops to a sufficiently small value, or until the best offer is less than the smallest transaction price in the previous period, or until there is not much time until the market closes. When any of these conditions are met, the Sniper jumps in and "steals the deal" by hitting the best bid/offer so long as that deal makes the Sniper a profit greater than its pre-set minimum profitability threshold.

The Sniper is, perhaps, too simple. Close inspection of the description of the strategy reveals that it doesn't adapt to market activity, and therefore doesn't engage in the self-equilibrating price-discovery process that is one of the major reasons for interest in CDA markets. A Sniper trader is essentially parasitic, benefiting from the price-discovery work of other strategies present in the market; if the other strategies present in the market are no good at price discovery, Kaplan's Sniper won't contribute anything to the process: it will "steal" any deal, even deals at prices a long way from equilibrium. The clearest demonstration of the shortcomings of Kaplan's Sniper comes when all traders in a CDA market are each playing the Sniper strategy: each of them sits there, doing nothing, waiting for someone else to act, and so there is a total lack of market activity until the "not much time left" triggers a sudden burst of activity, despite which no equilibration occurs. The real lesson of Kaplan's contribution is that it demonstrates that a very simple strategy can do surprisingly well in CDA markets so long as it can free-ride on the price-discovery activities of other traders in the market.

In the same year that the paper describing Kaplan's victory in the Santa Fe CDA-trading competition was published, a landmark paper involving a mix of traditional human experimental economics and software-agent market studies was published in the *Journal of Political Economy* by economists Dhananjay Gode and Shyam Sunder (Gode & Sunder, 1993). Gode & Sunder were interested in understanding how much of the efficiency of a CDA is due to the intelligence of the traders, and how much of it is due to the organisation of the market. To explore this, they introduced the idea of running *zero-intelligence* (ZI) trading agents in computer-simulated CDA market experiments, the structure of which were much the same as those first introduced by Smith in 1962. Gode & Sunder ran five sets of computer-based CDA experiments with humans as a control, and then replaced the human traders with various types of ZI trader. The "zero intelligence" name can be interpreted literally: the degree of intelligence that ZI traders have in their strategies is zero, or very-near-to-zero. Gode & Sunder's most minimal ZI traders simply generated random prices for bids and offers, drawn from a uniform distribution over the range from zero to some arbitrary system-maximum. As these ZI traders had no economic constraints at all, they were named ZI-U (for *Unconstrained*). If a ZI buyer's randomly-generated bid price was greater than the most recent ZI seller's randomly-generated offer price, then the fact that the prices "cross" meant that the two traders entered into a transaction (and *vice versa* for when a seller's bid crosses below the most recent best bid).

The next step up from ZI-U, reasoned Gode & Sunder, was a trader that still generated random prices for bids or offers, but instead used a uniform distribution bounded from above (for buyers) or from below (for sellers) by the trader's limit price. That is, the ZI-C (for *Constrained*) trader was allowed to generate random bid or offer prices subject to the one constraint that the random-prices should not create the possibility of the trader entering into loss-making deals.

Gode & Sunder ran experimental-economics lab-tests with ZI-U, ZI-C, and human traders in five different markets, and monitored the allocative efficiency of the transaction-price time series in each. The results from the ZI-U traders were, it is simplest to say, plain useless: each of their transaction-price time-series showed no sign of equilibration; it looked like random noise, which is exactly what it was. The surprise though was that the time series from the ZI-C experiments were surprisingly human-like, showing clear signs of convergence toward the theoretical equilibrium price within each trading "day", and the allocative efficiency scores for the ZI-C market experiments were extremely close to those of the human ones. The conclusion drawn from this was that much of the "intelligence" in the systems lies in the dynamics, the emergent behavior, of the CDA market mechanism, and not in the traders. A second issue demonstrated by Gode & Sunder's 1993 results was that while high allocative efficiency scores were clearly no guarantee that the traders in the market possessed significant intelligence (because human and ZI traders each scored roughly the same on allocative efficiency), the results from the two types of trader could be distinguished by their *profit dispersion* scores. Profit dispersion is a measure of the extent to which the profit/utility generated by each individual trader in the market experiment differs from the profit that would be expected of them if all transactions took place at the equilibrium price.

In a second paper published shortly afterwards (Gode & Sunder, 1994), Gode & Sunder presented results from experiments where, first of all, human traders interacted in a series of CDA experiments; next, those humans were then each invited to design an AI program trader (i.e., a "robot") to compete in a second round of robot-vs-robot CDA experiments; and then finally the AI robots were replaced by ZI-C traders for a third round of experiments. Although Gode & Sunder's 1994 paper includes a description of the market data made available to the AI robots, no specific details of the individual AI robots' algorithms were given. Gode & Sunder report that, on the average, the ZI traders required more bids and offers to be quoted into the market per transaction than the human traders did, but the AI traders required even more than the ZI's. As with their 1993 paper, in 1994 Gode & Sunder found that there were no significant differences in the allocative efficiency figures for the human, AI, and ZI traders, but that they could be told apart by their profit dispersion statistics.

Gode & Sunder's results were striking, and rapidly attracted considerable attention, yet from the perspective of scientific methodology there was a curious omission from their work: both in their own narrative, and when others cited their papers, there was a lack of a detailed causal mechanistic explanation for why the ZI-markets managed to produce such remarkably human-like price dynamics. The ZI markets are nothing more than a bunch of stochastic processes (the individual ZI traders) engaging in nonlinear interactions with each other (via the CDA market mechanism), and so *prima facie* one might expect some mathematical analysis of the nonlinear stochastic system to offer an explanation for the results. However, the explanation offered in the contemporary economics literature for why ZI markets behave as they did seemed instead to rest on vague hand-waving appeals to the idea that the "intelligence" in the system lies in the CDA market mechanism rather than the heads of the traders.

In 1997, one of us (Cliff, 1997) presented the first detailed mathematical analysis and replication of Gode & Sunder's ZI results. The mathematical analysis was not particularly complicated, but it demonstrated that the ability of ZI-C traders to converge on the equilibrium price in the CDA market was very heavily dependent on the shape of the market's supply and demand curves. Put simply, the nature of the specific supply and demand curves used by Gode & Sunder in their 1993 paper was such that a purely theoretical analysis (i.e., deriving a set of equations) could predict that the expected value of the transaction-prices in those experiments would be equal or near to the theoretical equilibrium price given by the intersection of those supply and demand curves. (Put more bluntly: they didn't really need to run their CDA experiments; the mathematical analysis lets us know the results before the experiments have taken place). Furthermore, from this analysis, it was possible to demonstrate that if the supply and demand curves in Gode & Sunder's 1993 paper were shaped differently, the ZI-markets would have converged on transaction prices that were very different from the theoretical equilibrium price, and hence would not have scored at all well on allocative efficiency. That is, the ZI traders operating in differently-shaped CDA markets would have got it very wrong; they would have failed to equilibrate.

Cliff demonstrated that these differently-shaped supply and demand curves did not have to be at all unusual: supply/demand profiles familiar from the experimental economics literature all the way back to Smith's 1962 paper were sufficient to cause the ZI markets to fail to equilibrate.[9] Cliff reinforced his theoretical analysis by writing computer programs that implemented an independent replication of Gode & Sunder's ZI-market experiments, and the failures that were predicted from his theoretical analysis were demonstrated to occur in the 'real' system too. Cliff's results demonstrated that, when it comes to the degree of intelligence required of the traders in a CDA market, zero is certainly not enough (Cliff & Bruten, 1998a).

After Cliff had demonstrated that ZI-C traders failed to equilibrate in CDA markets with commonplace supply and demand schedules, he then developed a new trading algorithm that did not exhibit the same failures that ZI-C traders did. Inspired by the minimalism of Gode & Sunder's approach, Cliff's named his algorithm Zero-Intelligence-Plus, or ZIP. Unlike ZI-C, ZIP traders do not issue randomly-generated prices for their bids and offers. Instead, each ZIP

[9] Technically, ZI-C markets will fail to equilibrate when there is a sufficiently large difference in the absolute values of the gradients of the supply and demand curves, and/or in so-called "box-shaped" supply/demand schedules where both supply and demand are perfectly elastic (that is, the curves are flat) and there is an excess of either demand or supply.

trader quotes a price that it generates by combining its (private, fixed) limit-price with its *margin coefficient*. For example, a seller with a limit price of $2.00 and a margin of 10% will quote $2.20 (i.e., $2.00 plus 10%), while a buyer with the same limit and margin will quote $1.80 (i.e., $2.00 minus $10%). A ZIP trader's margin is, like its limit price, private to that trader; but each trader can alter its margin over time, in response to events in the market. Cliff devised a minimally simple "decision tree", a set of simple if-then heuristics, to determine when a ZIP trader should lower its margin, and when it should raise it, and he then introduced some long-established machine-learning mathematics, which he rearranged for use in determining by how much the ZIP trader's margin should be changed (see Cliff & Bruten 1998b for a brief description; or Cliff 1997 for full details and illustrative program-code).

Thus, significantly, ZIP was not only autonomous (i.e., capable of generating its own orders without human intervention, much like ZI-C) it was also *adaptive.* ZIP's adaptivity meant that it "learned" an appropriate margin on the basis of its "experience" in the marketplace, and if there was a change in the market (such as a shift in supply and/or demand, possibly altering the equilibrium point) then ZIP traders could rapidly adapt to the new market conditions. Cliff (1997) demonstrated that ZIP traders did not suffer the failures that affected Gode & Sunder's ZI-C in CDA markets, and also that ZIP traders could give reliable behaviour in auction market mechanisms other than the CDA. The primary debt that Cliff's ZIP owed to Gode & Sunder's ZI traders was an emphasis on minimalism, attempting to identify the simplest possible trader-algorithm that could reliably give human-like market dynamics. In essence, ZIP was written to make the academic point that while Gode & Sunder-style zero intelligence was not enough, it only required a little bit more than zero intelligence to get interestingly human-like CDA market dynamics. There was no intention when developing ZIP of creating something that could outperform human traders, nor of making anything that could be of use in the real-world financial markets. However, in the years after publication of the ZIP algorithm, it first became well known for outperforming human traders (discussed below in Section 3.1); and then its minimal simplicity – and hence comparatively high speed of execution – proved to be of interest when technology developments in the global financial markets led to intense competition in the area of low-latency trading (see e.g. Cliff, Brown, & Treleaven 2011).

Cliff's (1997) ZIP algorithm was not the only adaptive autonomous trading algorithm that was developed in the latter half of the 1990s. At pretty much exactly the same time, but entirely independently, Steve Gjerstad was working on his PhD in economics with his advisor John Dickhaut, and the two of them co-authored a paper in 1998 that summarised a core aspect of Gjerstad's PhD research, which sought to offer a more convincing explanation than Gode & Sunder's (1993) of how interaction among adaptive agents could lead to price-discovery in CDA markets (Gjerstad & Dickhaut, 1998). Their paper did not give a name to the trading algorithm that they developed and explored, but it has since become known in the literature simply as the Gjerstad-Dickhaut, or GD, algorithm.

GD is a relatively sophisticated algorithm that requires each GD trader to compute its own private "belief function" using data from recent market activity; this function indicates the probability, for every possible bid or offer price, that a bid or offer would be accepted at that price. It uses a standard mathematical approach (cubic spline interpolation) to compute values of the belief function for prices that do not occur in the history list.[10] When a GD trader comes to determine a price to quote in the market, it chooses a price which maximizes its expected gain: for each possible price, this is simply calculated by multiplying the utility gain from a

[10] Our opinion is that GD's specification of cubic-spline interpolation is something of a red herring: any of many smooth interpolation processes would probably yield very similar results.

transaction at that price by the belief-function's probability of the quote being accepted at that price. Gjerstad and Dickhaut demonstrated that CDA markets populated by GD traders exhibited the kind of transaction-price dynamics that were familiar from human CDA experiments.

The GD algorithm was later found to require a minor modification to reduce price volatility: the modification was simply that the definition of the belief function should be altered so that it returned a zero-probability of acceptance for bids lower than the previous lowest transaction price and also a zero for offers higher than the previous highest transaction price. This modified GD, or MGD, algorithm has been a commonly-used benchmark in the literature ever since.

So, by the late 1990's, there were several autonomous "robot" trading algorithms for CDA markets that had attracted attention in the research community: Kaplan's Sniper had won the Santa Fe CDA competition; subsequently Gode & Sunder's ZI-C traders had initially appeared to give surprisingly human-like CDA market dynamics, but Cliff then demonstrated that in fact ZI-C was not intelligent enough, and so he developed ZIP in response; and finally Gjerstad & Dickhaut had published the GD algorithm as a tool for explaining issues in the economics of CDA markets. While Kaplan's Sniper had been shown to work well in *strategically heterogeneou*s markets (i.e., markets with a mix of different trader strategies or robot algorithms), ZI-C, ZIP, and GD had been evaluated only in strategically homogeneous markets (i.e., markets where all traders were trading the same strategy, but with different limit prices and internal parameters for each trader).

The issue of how the ZI-C, ZIP, and GD/MGD robot algorithms fare when they compete against each other in strategically heterogeneous markets was explored in a series of papers by a team of researchers at IBM's T.J.Watson Research Labs in New York State, USA.

In 2001, Gerald Tesauro and Raj Das reported on results from experiments they conducted at IBM where they tested populations of trading-agents using pairs of algorithms chosen from ZI-C, Kaplan, ZIP, GD, and MGD, in real-time market experiments where traders had to trade multiple units with different limit prices. It was this more realistic environment that prompted them to develop the MGD modification of GD described above. A similar minor modification was required for ZIP but Tesauro & Das did not alter the name of that algorithm. Tesauro & Das first performed strategically homogeneous population tests for validation by comparison to the previously-published results for each algorithm. They then performed "balanced-group" tests in which:

> "…buyers and sellers are evenly split between two types of [trader] agent [algorithm], and every agent of one type has a counterpart of the other type with identical limit prices. …we believe [this] test to be the fairest way to test two different algorithms against each other." (Tesauro & Das, 2001).

The results from these tests indicated that, in robot-vs-robot experiments at least, MGD consistently outperformed the other algorithms. In the language of game-theory, MGD "dominated" the other strategies.

In subsequent work, Tesauro and Das worked with IBM colleagues William Walsh and Jeffrey Kephart (Walsh, *et al.*, 2002) to explore the interaction dynamics not merely between pairs of robot strategies (such as MGD-vs-ZIP) but instead between three strategies in simultaneous competition. Walsh *et al.* explored the "population dynamics" of CDA markets with various mixtures of ZIP, GD, and Kaplan-Sniper traders, and characterised the dynamics for all possible mixture ratios. Again, the results indicated that in general MGD was dominant.

IBM's studies of interactions among different robot strategies have been replicated and extended in PhD theses by Iain Toft (2007) and Perukrishnen Vytelingum (2008). Both theses offer support for the results claimed by IBM, although Toft reports that the difference between the performance advantage of MGD over ZIP did not appear to be particularly significant, and that in some of his MGD-vs.-ZIP experiments it was his implementation of ZIP that outperformed MGD (Toft, 2007, p.154). In a novel extension, Toft (2007, pp.182-184) describes the results from a six-way competition involving ZI-C, Kaplan's Sniper, MGD, ZIP, and two other CDA trading algorithms, the "A-FL" algorithm of He *et al.* (2003), and the minimal "SI" algorithm devised by Toft himself as an alternative benchmark to ZI-C. All six of these algorithms are tested in the same market experiments together, competing against each other in the same CDA market (with each algorithm being played by two buyers and two sellers): in these experiments, ZIP and GD were again clearly dominant, with ZIP having the upper hand.[11]

IBM was not the only large computer company to have invested in a sizeable research team working on automated trading agents. In 1996, Steve Gjerstad, inventor of the GD algorithm, had left academia to join Hewlett-Packard Labs in Palo Alto, California, although in 1999 he moved to IBM's T.J Watson Research Labs to join team of researchers whose work was described above. In 1998, Dave Cliff, inventor of ZIP, had also left academia to join Hewlett-Packard Labs' Agent Technology Group, a similar-sized team to IBM's, based at HP's European research labs in Bristol, England. At both HP and IBM, research was focused on studies of interacting software agents, but both companies had also invested in experimental economics lab facilities (Gjerstad had set up the HP Labs facility, before moving on to do the same for IBM). At HP, the experimental economics lab was used in a largely conventional manner: to study the interaction of human traders in electronically-mediated markets, in much the same way that Smith had established years before (for an example of work done in HP's experimental economics facility, see Cliff & Preist, 2001). The IBM team, however, put their experimental economics laboratory to a truly novel use: they conducted the first ever experiments where human traders and autonomous "robot" software agents interacted *within the same market*, allowing for systematic comparison of human-robot interactions in electronically mediated marketplaces. The revolutionary nature of IBM's work, and the surprise outcome of their experiments, generated global media coverage when their first results were published at the 2001 International Joint Conference on Artificial Intelligence (IJCAI), the premier peer-reviewed conference in the field. We explore the topic of human-robot interactions in detail in the next section.

[11] Toft (2007, p.185) notes that his results contradict those claimed by He *et al.* (2003) for the A-FL algorithm, and states: "[Toft's] early results with A-FL were consistent with results in the originating paper, therefore the disparity is attributed to difficulty in implementing and configuring the agent. An A-FL agent requires extensive and meticulous configuration. … there are a total of 34 parameters, a considerable number for a [*sic*] agent that reasons with their limit price, the outstanding bid and offer, and the median price only. Without reference implementations of agents for a particular market structure it is difficult to reproduce previously published results when the agent is complex or the originating literature lacks implementation details. By comparison, and in stark contrast to A-FL, [ZIP] and [MGD] are comparatively simple agents with comprehensively documented implementations and configurations."

3. Humans-vs-robots

The 2001 IJCAI paper published by members of the IBM team (Raj Das, James Hanson, Jeff Kephart, & Gerald Tesauro) is described in some detail in Section 3.1. Given the groundbreaking nature of that paper, it seems reasonable to expect that in the decade since there would have been a significant number of additional papers published by authors around the world, replicating and extending the work of the IBM team. Surprisingly, this is not the case. To the best of our knowledge (and this is something that we have invested considerable effort[12] in establishing), the total number of papers that either replicate or extend the IBM robots-vs.-humans work can be counted on the fingers of one hand: there are two papers by Jens Grossklags and Carsten Schmidt (2003, 2006) which we describe in Section 3.2; and there are two 2011 papers by Marco De Luca and Dave Cliff (two of the authors of this review) which we summarize in Section 3.3. We conclude with a speculative discussion, in Section 3.4, of possible reasons for this surprising lack of published research that has built on IBM's seminal study.

3.1 Das, Hanson, Kephart, & Tesauro (2001)

Das *et al.* were all IBM researchers working at the IBM T.J.Watson Research Labs. Their 2001 paper describes how they pitted human traders against trading agents in an experimental economics lab: they ran some control experiments where human traders sat at trader interfaces running on desktop PCs and communicated with a central "exchange" server PC, as was (by then) routine practice in experimental economics labs; they had also, as we saw in Section 2.2, previously run experiments where the only traders communicating with the server were "robot" software agents, as is common in ACE experiments and in MBC research and development.

And then they ran a series of experiments in which they explored the dynamics of markets in which some of the traders were humans, and others were robots. Somewhat astonishingly, this had never been done before: the paper by the IBM team is the first study of human-robot interactions in the CDA.

The algorithms they tested against humans included versions of ZIP and MGD, modified to work with the *order-book* that their experimental-economics facility provided: this is a step closer to the reality of real-world electronic exchanges, which routinely show a list or queue of the *n* current best bids and offers, ordered best to worst (we illustrate an order book from OpEx later, in Figure **4.2**). In Smith's early experiments, and in the ZI/ZIP/GD trading-agent research that Smith's work subsequently inspired, traders could only interact with each other by taking the current best bid or the current best offer, or by making a quote that improved on the best

[12] Using Google Scholar (scholar.google.com) one of us (Szostek) identified more than 150 papers known to Google that cite the original IBM IJCAI paper (Das, *et al.*, 2001) and then read each of those papers in sufficient detail to determine whether the reason for citing IBM's work was because the citing paper was reporting on results from human-vs-robot experiments, or whether the IBM paper was merely referred to in passing. The only papers found that actually reported on new human-robot experiments were the two (2003, 2006) by Grossklags & Schmidt, which we describe in Section 3.2. It is plausible that somewhere there exists one or more peer-reviewed papers that do report on human-vs-robot work but don't cite the Das *et al.* paper, or ones that do cite it but which do not exist in a machine-readable form on the Internet and hence have not been detected by Google: we cannot rule out either of these possibilities, but we think them both unlikely. As further corroboration of our claim that the papers discussed here are the only papers that report on human-vs-robot experiments, we note that in two major survey/review papers (Duffy, 2006; Grossklags & Hall, 2007) the only human-robot auction-market experiments discussed are the one we describe in Sections 3.1 & 3.2.

bid or offer (that is, the markets showed an order book with *n*=1). Because Gjerstad's GD algorithm required some significant modification for it to work with an order book of length n>1 and to avoid groups of GD traders generating pathological bursts of market activity, Das *et al.* referred to it as the Modified GD, or MGD, algorithm.

The computational simplicity of both ZIP and MGD mean that software-agent traders running these algorithms can execute at superhuman speeds. To ensure that there was a reasonable chance of the human participants in their experiments interacting with the software agents in the experimental markets, Das *et al.* imposed a "sleep-wake" cycle on the software agents in their experiments: while awake, each robot would interact with the market, adjusting its internal variables in response to market events and possibly also issuing a quote into the market, and would then go into a "sleep" state, rendering itself inactive until it received a wake-up signal. Each robot's sleep-time was a fixed period of *s* seconds, with some small random variation added of up to ±25% so that the robots did not all have identical sleep times. Das *et al.* used two variations of the sleep cycle: in their "fast" experiments, *s*=1 and robots were woken up whenever a new quote was made or trade occurred; in their "slow" experiments, *s*=5 and robots were only issued with a wakeup when a trade occurred (that is, they slept through quotes that did not result in transactions).

Das *et al.* (2001) conducted six experiments: one ZIP-slow, one ZIP-fast, and four MGD-fast. In each experiment there was an equal number of humans and robots, and all the robots would be running the same strategy (i.e., either all ZIP or all MGD). Das *et al.* presented detailed plots of market activity from two experiments (one involving humans-vs-MGD, the other involving humans-vs-ZIP), and summarised the results from all six experiments in a data-table showing the "surplus" (actual profit extracted) and efficiency (actual surplus expressed as a percentage of the surplus that would result if all transactions took place at the market's theoretical equilibrium price) for humans and for robots.

Das *et al.* made two major qualitative observations. The first was that in their human-robot CDA markets, transaction prices were consistently and significantly off-equilibrium, which is strange given the CDA's widely-reported attractiveness as a mechanism for discovery of the underlying equilibrium price when populated entirely by humans, and also when populated entirely by ZIP or MGD robots: somehow, the mix of humans and robots interacting with each other impaired the CDA's equilibration dynamic. The second was that both ZIP and MGD agents consistently and significantly beat the human traders: in all six of Das *et al.*'s experiments, human-trader efficiency scores are less than those of robot-traders (over the six experiments, average human surplus was 7358 against average robot surplus of 10381; average human efficiency was 87.6% against average robot efficiency of 102.6%). Of the two robot strategies, MGD scores best, but also scores worst, and close inspection of the data-table given by Das *et al.* shows that the mean values of both the surplus and the market efficiency were higher for ZIP than for MGD: average surplus and efficiency for MGD were 10130 and 102.3%, respectively, but for ZIP they were 10883 and 103.0%. Das *et al.* did not discuss this; it seems likely that they did not consider the difference between ZIP and MGD to be statistically significant, and in any case were probably not interested in establishing which of the two robot algorithms was superior: Das *et al.* did not perform any MGD–slow experiments, and did not perform equal number of ZIP and MGD experiments, so drawing meaningful comparisons between ZIP and MGD is not possible from their data.

Das *et al.,* stated in the introduction to their paper that: "…the successful demonstration of machine superiority in the CDA and other common auctions could have a much more direct and powerful impact – one that might be measured in billions of dollars annually." This view, coming from the research labs of IBM, gathered a lot of attention. The clear implication was that, in the forseeable future, employing humans at the point of execution in the financial markets might soon cease to make economic sense: faster, cheaper, software technology could replace the rather expensive and rather slow human traders in the financial markets. This view of the near future, and its wider implications, was explored in depth in a subsequent report co-produced by IBM Business Consulting Services and The Economist Intelligence Unit, titled *The Trader is Dead, Long Live The Trader!* (IBM, 2006).

3.2 Grossklags & Schmidt (2003, 2006)

In their 2003 paper, Grossklags & Schmidt studied the extent to which the market behaviour of human traders in the CDA was altered by their knowledge of whether robot traders were present in the same market or not, using an electronic CDA where each human trader interacted with the other market participants via a trader-interface running on a PC. They ran a control experiment where only human traders were present in the CDA market, to establish a baseline for "normal" human behavior in their markets. They then ran experiments where a number of robot traders, each operating a simple automated arbitrage strategy (i.e., not one of the trading algorithms explored by Das *et al.*, 2001), were also present alongside the human traders. In some of their humans-vs-robots experiments, the human traders were not told about the presence of the robots (and hence the traders assumed that all the counterparties they interacted with via the trading interfaces were fellow humans); in other human-vs-robot experiments, the humans were told that the market included some robot traders. Thus, there were three distinct "treatments": the baseline control market with no robots; the market with robots but uninformed humans; and the market with robots and informed humans. Grossklags & Schmidt performed six independent experiments for each treatment, giving a total of 18 sessions. They found that when human traders were told that the market included robot traders, the market was more efficient; and (surprisingly) that when the humans were not told about the presence of robots, the human-vs-robot market was less efficient than the human-only market. In the conclusions of their 2003 paper, Grossklags & Schmidt make some valuable methodological observations:

> "On a methodological level we are concerned with the rather high variability of individual session averages for efficiency and behavioral variables observed in this and other market experiments.
>
> "We feel confident that our design and the statistical analysis … provide a good description of the underlying effects. Evidence on CDA markets relying on a single independent observation for each treatment should be treated carefully and may require further repetitions." (Grossklags & Schmidt, 2003).

Grossklags & Schmidt's 2006 paper is an expanded version of their 2003 paper. It includes more in-depth discussion of related work, more detailed analysis of the results, and a verbatim transcript of the instructions issued to the human traders, but does not include any substantive new results or insights.

Grossklags & Schmidt owed an obvious debt of inspiration to Das *et al.* (2001), and they duly cited that paper, but it is clear that Grossklags & Schmidt were using a different robot algorithm to any of the ones employed by Das *et al.*, and more significantly they were exploring a different issue: the effect of knowledge/ignorance of the presence of robot traders on the behaviour of human traders. So, while Grossklags & Schmidt might reasonably be described as having been inspired by the IBM work; they had certainly not replicated it.

Without an independent replication, any experimental claim is always under some doubt: it is (with the greatest of respect to the experimenters involved) always possible that there was some experimental error or confounding factor that the original experimenters were not aware of, which led them to draw an incorrect conclusion. For any experimental work, as soon as it has been independently replicated, the likelihood of it being in error is greatly reduced (but not eliminated: there is always the possibility that the experimenters who performed the replication happened to make the same mistake as the originators of the experiment). To the best of our knowledge, the first and only replication of the results presented by Das *et al.* (2001) came a decade later, when De Luca & Cliff published a paper in early 2011, discussed next.

3.3 De Luca & Cliff (2011a, 2011b)

As we discussed in Section 3.1, although Das *et al.*'s (2001) work had demonstrated machine superiority in the CDA, the difference between the performance of IBM's MGD algorithm and ZIP was so small that it was unlikely to be statistically significant. Put another way, IBM had shown that algorithmic trading systems could beat humans, but not that IBM's algorithm was the undisputed champion. Researchers at IBM continued to work on improving the performance of the MGD algorithm, and a year later IBMers Tesauro & Bredin (2002) published results from an extended version of MGD, which they named GDX. In that paper, Tesauro & Bredin demonstrated that GDX could consistently outperform ZIP in agent-vs-agent experiments, and in their paper they wrote: "We suggest that this algorithm [GDX] may offer the best performance of any published CDA bidding strategy."[13]

This claim seems reasonable enough, given Tesauro & Bredin's published results, but it is important to note that Tesauro & Bredin did not test GDX against human traders, which is perhaps why they expressed some caution in the phrasing of their claim: they *suggested* that GDX *may* offer the best performance. The only way to tell for sure is to test GDX against humans, in much the same way that Das et al. had tested MGD and ZIP against humans. Noting that no-one had done this, and indeed that apparently no-one had ever published a replication of any of Das *et al.*'s result, De Luca & Cliff published a paper in early 2011 in which they wrote:

> "To the best of our knowledge, GDX has never been tested against human traders under experimental conditions. In this paper, we report on the first such test: we present detailed analysis of the results from our own replications of IBM's human vs. ZIP experiments and from our world-first experiments that test humans vs. GDX. Our overall findings are that, both when competing against ZIP in pure agent vs. agent experiments and when competing against human traders, GDX's performance is significantly better than the performance of ZIP." (De Luca & Cliff, 2011a)

Thereby replicating Das *et al.*'s results for the first time in a decade, and vindicating Tesauro & Bredin's claim for supremacy of GDX made nine years earlier.

However, in that nine-year interval, Perukrishnen Vytelingum's PhD research at the university of Southampton, UK, had led to the development of a strategy loosely based on ZIP, with significant novel extensions, which he named the *Adaptive Aggressive* (AA) strategy, and in his PhD thesis he had demonstrated that AA could consistently out-perform GDX in robot-vs-robot

[13] For completeness, we note here that Toft (2007, p.171) states that he was unable to replicate the performance claimed for GDX by Tesauro & Bredin (2002).

experiments (Vytelingum, 2006; Vytelingum, *et al.* 2008).

Having experimentally verified GDX's dominance over all other traders (human or robot) in their 2011a paper, the obvious next question for De Luca & Cliff to ask was whether Vytelingum's AA could also outperform GDX in human-vs-robot experiments. If it could, AA would be the new 'undisputed champion' trader-agent strategy. This was a question that De Luca & Cliff answered in their second 2011 paper (De Luca & Cliff, 2011b), presented at the 2011 IJCAI conference, the same conference at which Das *et al.* had published their first human-vs.-robot results a decade previously.

The results presented by De Luca & Cliff (2011b) at IJCAI confirmed Vytelingum's claim (by independent replication) that AA outperforms ZIP, MGD, and GDX in agent-vs.-agent experiments, and then provided the first-ever demonstration that when AA robots compete against human traders in human-vs.-robot CDA experiments, AA's performance against humans is superior to that of ZIP, MGD, and GDX. In a direct paraphrase of the words of Tesauro & Bredin (2002), De Luca & Cliff wrote: "We therefore claim that, on the basis of the available evidence, AA may offer the best performance of any published bidding strategy." At the time of writing, the results presented by De Luca & Cliff (2011b) have not yet been independently replicated, and should therefore be treated accordingly. However, in Section 4 of this review, we describe the details of the OpEx system that was used by De Luca & Cliff for their experiments, and De Luca's plan is to publish the full OpEx source-code under an appropriate Creative Commons open-source copyright-release scheme in due course: once OpEx is available as a free, common standard for running human-vs.-agent experiments, the hope is that there will then be a significant increase in the frequency with which human-agent experiments are replicated and extended. The curious lack of any reasonable history of replication and extension is something that we discuss in the next section.

3.4 Discussion

In concluding our review of the literature that we presented in Sections 2 and 3, we think there are three significant issues that deserve to be highlighted:

- The histories of experimental economics and autonomous agent research are, from the perspective of this review, essentially non-intersecting. That is, apart from the few papers that we discussed here, neither field appears to have devoted any significant attention to understanding how humans and autonomous agents interact in experimental auction-market conditions. To the best of our knowledge, all research in the experimental economics literature has been devoted to understanding how humans interact with one another in the context of various market mechanisms; and in agent-based computational economics (ACE) economists have worked at developing autonomous trading agents, but their agents are always proxies for human subjects, in the manner discussed by Arthur (1993) and exemplified by Gjerstad & Dickhaut (1998). Similarly, while a sizeable proportion of excellent work by computer-science/AI researchers interested in developing autonomous agents has been heavily inspired by classical economics, by operations research, and by game theory, seemingly all of that has been directed toward a vision of the future where groups of autonomous agents interact with one another (i.e., so-called multi-agent systems) with minimal involvement from humans: in so-called agent-mediated e-commerce, for instance, the motivation seems to be largely that agent shall speak unto agent, freeing humans from the tiresome tasks of negotiating or executing commercial transactions. The fact that global financial markets are, right now, populated by interacting human and robot traders seems to be a fact that has passed by (or, at least, has not been judged to be particularly relevant by) both academic economists and AI autonomous-agent researchers.

- As might be expected from a Nobel Laureate, Smith's work has had a huge and lasting influence. But the extent to which the methods that Smith first described in his 1962 paper are still in use, essentially unchanged, in experimental work half a century later is truly striking: the experiments reported by Gode & Sunder (1993, 1994), by Cliff (1997), by Cliff & Bruten (1998a, 1998b), by Gjerstad & Dickhaut (1998), by the IBM team (Tesauro & Das, 2001; Das, Hanson, Kephart, & Tesauro, 2001; and Tesauro & Bredin, 2002), by Grossklags & Schmidt (2003, 2006), by Vytelingum *et al.* (2006, 2008), and most recently by De Luca & Cliff (2011a, 2011b) are all essentially based on *exactly* the experiment design that Smith introduced in 1962: the market only trades one type of abstract, commodity, financial instrument, and its trading is broken into discrete periods ("days") with simultaneous replenishment of all stock or money for all traders at the start of each day. Perhaps after 50 years it is time to be a little more adventurous in formulating experiments.

- There is a startling, near-zero, lack of replication and extension of IBM's pioneering agent-human studies, work that was first published in 2001. In the decade since then, Grossklags and Schmidt (2003, 2006) performed one experiment that was related to, but did not replicate, the IBM work. The only replication of IBM's work that we know of is the one published by De Luca & Cliff (2011a) and the only extension is by the same authors (2011b). Given that in most well-established sciences the veracity of an experimental result is (for the best-willed of reasons) not well trusted until it has been replicated, this seems like something of a systemic failure: why, exactly, is there such a poor tradition of replication?

It seems plausible that the final point, concerning the lack of replication, is due to a perception that the costs of performing experiments that study human-agent interactions is so high that only major corporate research labs (such as IBM's or HP's) can afford to invest in the necessary facilities. While this may have been true a decade or more ago, the continuing Moore's Law reductions in the cost of computing equipment mean that it is now possible to purchase all the necessary equipment for a few thousand pounds, and it will all comfortably fit inside a large suitcase. This is significant: it means that the entire "laboratory" is sufficiently mobile that it can be taken to venues where experiment subjects are, rather than requiring the subjects to come to the Laboratory: De Luca & Cliff (2011a) refer to this as the "lab-in-a-box" approach. But of course there is little point buying the hardware if one does not have appropriate software to run on it. In the next section, we give a description of *Open Exchange* (OpEx), an experimental economics software system designed and implemented by one of us (De Luca), which is intended to be released as free open-source software sometime in the near future, as a resource for use by the research community; and then in Section 5 we present as-yet-unpublished results from new experiments with OpEx where (with respect to the second point, above) we explore a more adventurous, and more realistic, experiment design.

Our hope is that the release of OpEx will result in a significant increase in the number of experimental studies of human-robot trading interactions; we would be highly gratified if it is somewhat less than ten years before the results presented in Section 5 are independently replicated, or refuted even.

4. Open Exchange (OpEx)

Open Exchange is the experimental economics market simulator designed and developed by De Luca over 2009-2010. The first results from experiments with OpEx were published in (De Luca & Cliff, 2011a, 2011b), summarized previously in Section 3.3. In Section 5 we present new results, not yet published elsewhere, from additional sets of OpEx experiments. Before that, in this section, we describe the design of OpEx itself.

Figure 4.1 illustrates the interaction between the core components in a simple configuration. The connections between the components on the left hand side show the flow of order data. Orders represent the traders' instructions to buy or sell a specific quantity of a given product at a particular price condition. Human traders enter their orders in the Trading GUI, a graphical application that allows users to view the market order book (i.e. the descending-ordered list of currently outstanding bids, and the ascending-ordered list of currently outstanding offers), their “blotter” (personal history of orders and trades), and, in case of a Sales Trading simulation, their assignments. Agent traders, on the other hand, produce orders automatically, without the need of human intervention, on the basis of the market conditions that they observe. Once generated, orders are sent to the Order Manager, which routes them to the appropriate order processor (in this example, the single Exchange) depending on the destination specified by the sender. Once received by the Exchange, orders are processed according to the specific order matching logic implemented and order completion data is passed back to the Order Manager, which dispatches it to the appropriate sender. It is worth noting that order data are private, as only the originator of an order receives the order completion data relative to that specific order, which will let her know its progress.[14] Conversely, market data are published on the Market Data Bus and can be seen by every market participant.

The order matching logic that we will cover in detail here is the *price-time priority matching logic*, which constitutes the basis for the CDA. The Exchange assigns an individual container to all the orders for each product (or instrument), and all the orders for the instrument are grouped into the particular container; the container is called the order book, which is empty when the trading day starts and changes when orders for that specific instrument are entered and executed. All buy orders (*bids*) are first sorted in descending order of price, and in ascending order of time for any equal prices. This means that bid orders from traders that are willing to pay the highest price have priority over orders at lower prices, and when two orders have the same bid price, the order that was entered earlier has priority over any subsequent orders at that same price. Similarly, all sell orders (*asks*, or *offers*) are first sorted in ascending order of offer price, and in ascending order of time for equal prices. Thus, offer orders where traders are willing to accept the lowest price have the highest priority, and the order entered earlier gains a higher priority when two orders have equal offer price. A hypothetical order book for Vodafone's stock is shown in Figure 4.2. The leftmost two and the rightmost two columns represent the buy side (bids) and the sell side (asks) of the order book, respectively. Each side is made up of a number of rows, or *levels*, each of which describes the quantity available on the market at the specified price. Levels are sorted by descending price in the buy side, and by ascending price in the sell side, so that the topmost row contains the best bid (highest buy) and best ask (lowest sell) price of an instrument. The maximum number of levels in an order book is known as its *depth*. The Trading GUI subscribes to the market data bus and displays the order book in real-time, so that the traders can adjust their orders to match the market conditions. Market

[14] This is in line with real-world electronic trading scenarios, where anonymity is often an indispensable requirement.

data is yet more crucial for robot-trader agents that, lacking any human intervention, base their decisions entirely on the data in the order book.[15]

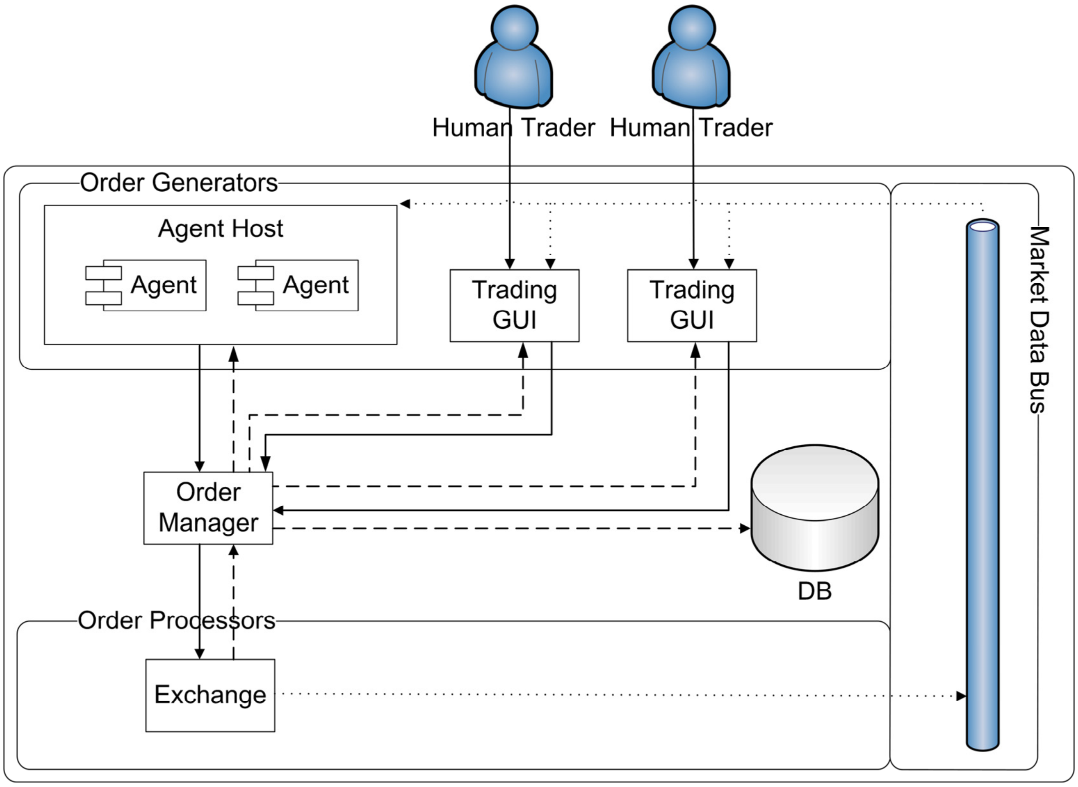
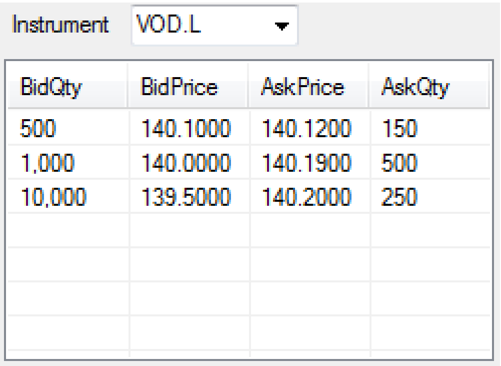


Figure 4.1: an instance of Open Exchange. The solid lines and the dotted lines represent the flow of order data, respectively the requests and the replies. The sparsely dotted lines indicate the market data flow, from the Exchange back to the order generators through the Market Data Bus.

The order matching logic that we will cover in detail here is the *price-time priority matching logic*, which constitutes the basis for the CDA. The Exchange assigns an individual container to all the orders for each product (or instrument), and all the orders for the instrument are grouped into the particular container; the container is called the order book, which is empty when the trading day starts and changes when orders for that specific instrument are entered and executed. All buy orders (*bids*) are first sorted in descending order of price, and in ascending order of time for any equal prices. This means that bid orders from traders that are willing to pay the highest price have priority over orders at lower prices, and when two orders have the same bid price, the order that was entered earlier has priority over any subsequent orders at that same price. Similarly, all sell orders (*asks*, or *offers*) are first sorted in ascending order of offer price, and in ascending order of time for equal prices. Thus, offer orders where traders are willing to accept the lowest price have the highest priority, and the order entered earlier gains a higher priority when two orders have equal offer price. A hypothetical order book for Vodafone's stock is shown in Figure 4.2. The leftmost two and the rightmost two columns represent the buy side (bids) and the sell side (asks) of the order book, respectively. Each side is made up of a number of rows, or *levels*, each of which describes the quantity available on the market at the specified price. Levels are sorted by descending price in the buy side, and by ascending price

[15] The order book is actually one of the market indicators that are essential to the activity of real-world stock traders and algorithmic trading systems.

in the sell side, so that the topmost row contains the best bid (highest buy) and best ask (lowest sell) price of an instrument. The maximum number of levels in an order book is known as its *depth*. The Trading GUI subscribes to the market data bus and displays the order book in real-time, so that the traders can adjust their orders to match the market conditions. Market data is yet more crucial for robot-trader agents that, lacking any human intervention, base their decisions entirely on the data in the order book.[16]



Instrument: VOD.L

BidQty	BidPrice	AskPrice	AskQty
500	140.1000	140.1200	150
1,000	140.0000	140.1900	500
10,000	139.5000	140.2000	250

Figure 4.2: A hypothetical order book for Vodafone's stock, as displayed by OpEx Trading GUI. This book has a depth of three.

One instance of OpEx is identified by the unique instance of the Configuration Server, which all the components of that instance refer to. The Configuration Server provides the components with their configuration, so that each component can connect to the specific services it needs. A minimal functional instance of OpEx would include one Order Manager, one Exchange, and at least two order generators (each either an Agent or a Trading GUI). However, thanks to its distributed architecture, OpEx can scale easily to virtually any number of order processors and order generators; in particular, the order generators in an instance can be any combination of Agents and Trading GUIs, both proprietary and sales. Simulating sales trading scenarios requires the Assignment Server, the OpEx component that dispatches each instruction (or assignment) to the appropriate agent entitled to execute it. The assignments are stored on the database as well as the configuration of the components. OpEx also stores in the database several data that change dynamically in the course of trading simulations, including order progress data, trades, quotes, and supply and demand curves. Post-experiment data analysis can thus benefit from all the advantages of Data Base Management Systems (DBMSs), particularly facilitating the creation of reports and charts.

In the Open Exchange framework, automated trading agents are implemented as individual plugins running on an instance of the *AgentHost*. In line with the distributed architecture of OpEx, there can be multiple instances of the Agent Host, each one running a particular set of Agents. Every Agent implements one specific algorithm and has its own configuration settings, loaded at startup. One instance of the Agent Host is capable of running multiple instances of the same Agent, so that more than one automated trader following a specific strategy can participate in the market simultaneously. The behaviour of an OpEx Agent consists of cyclically listening to stimuli and reacting to them sequentially by performing one or more actions. Agents are idle as they wait for the next stimulus, whereas they perform calculations and can send a signal to the market when they are active. Each stimulus is produced by a specific source

[16] The order book is actually one of the market indicators that are essential to the activity of real-world stock traders and algorithmic trading systems.

asynchronously, and it is conveyed to a common stimulus collector together with all the other unprocessed stimuli. Sorted chronologically, the resulting stimuli queue is then processed sequentially by the appropriate processor. The strategy of an Agent is univocally determined by defining the reaction to four different classes of stimuli: *Timer*, *Quote*, *Session*, and *New Assignment*. The internal alarm mechanism of an Agent produces a *Timer* stimulus periodically, with the frequency specified in that Agent's preferences. When the alarm rings, a *Primary Timer* stimulus is generated. However, agents also allow Secondary *Timer* stimuli, which are periodically generated between two consecutive Primary Timer stimuli in order to achieve timed events at finer granularity.[17] *Quote* stimuli are produced by the market data listener of an Agent every time a market participant sends a new order or amends an existing one. Unlike Timer stimuli, Quote stimuli bring information about the timestamp of the event: the price of the order; the quantity; the direction (buy or sell) and whether the order has been accepted (i.e. the order has crossed an existing order on the contra-side of the book, leading to a trade) or rejected (i.e. the order has not matched any orders on the opposite side, therefore it ended in the appropriate price queue in the book). *Session* stimuli are also captured from the market data listener, and indicate whether the market has just opened, or it has just closed. In addition, a Session stimulus indicates how long the next session is going to last, and when it is going to start. Finally, *New Assignment* stimuli are produced by the assignment listener of an Agent every time there is a new assignment ready to be processed for that specific Agent. These stimuli carry complete assignment information, including quote direction, instrument, quantity and limit price. Agents process assignments sequentially, in the same order they receive it: this way, an assignment cannot be traded until the previous one has been completed. Every time the market closes, the assignments for every Agent are reset, and new assignments are sent at the start of the next trading session, right after the market-open Session stimulus.

Our choice of timing mechanism is consistent with the previous IBM work (Das *et al.*, 2001), where similar timing rules were used to regulate the activity of the agents. However, the results presented by Das *et al.* are from experiments run using two different timer periods ("fast", 1 second; and "slow", 5 seconds) for the different algorithms; this is an issue that we discuss in Section 5. In our work reported here, the "fast" configuration corresponds to the agents' primary and secondary timer period set to 1 second, while the "slow" configuration timing settings are 10 seconds for the primary timer, and 2.5 seconds for the secondary timer.

Architecture

The most common use-case of OpEx consists of several human traders using individual physical machines to trade either with each other, or with automated traders: the need for a distributed architecture was established in the early stages of the development process. Each instance of the Trading GUI denotes a client of the Configuration Server, to which the GUI connects to on start-up, in order to notify the system of its presence and to retrieve its configuration settings. The connection with the Configuration Server is held by the GUI, and kept alive by sending heartbeat messages to the server, until the GUI is shut down, only after the system has been logically notified of the exit of that instance from the pool of running applications. This client/server paradigm applies to all other OpEx components: Order Manager, Exchange, Agent Host and Assignment Server. The advantages of using this approach are dual: first, it constitutes a centralised mechanism that regulates the running status of the components; second, it reduces the amount of local static configuration of the components to the bare minimum, that is the network location of the Configuration Server (after

[17] Secondary Timer stimuli may awake an agent in order to perform calculations, but, unlike Primary Timer stimuli, they never lead to perform an action on the market (i.e. send a new order to buy or sell, or cancel an order).

Name	Vendor	Free	Languages and APIs supported
MQ Series	IBM	No	Java, .C/C++, .NET
JMS	Sun	Yes	Java
MarketView PSH	ION Trading	No	Java, C/C++, .NET
MSMQ	Microsoft	Yes (comes with Windows)	.NET
Rendezvous	TIBCO	No	Java, C/C++, .NET, Perl

a component has connected, it can use the more sophisticated mechanisms described later in this section to retrieve its configuration).One of the fundamental functionalities the Trading GUI and the Sales GUI implement is order entry: human traders participate in the market by entering and sending order requests. The order data generated by the GUIs include all the information required by the order matching logic. However, orders are not sent directly to the Exchange but through the Order Manager. The reasons why we created this component are multiple. First, because the number of order requests (including creation, amendment and cancellation) in a high frequency platform is significantly high, the requests need to be labeled carefully in order to be managed: the Order Manager acts as an on-demand centralised unique ID generator, and provides the order generator applications with a token that univocally identifies their requests across the whole instance. Secondly, the Order Manager isolates the routing functionality from the clients and promotes decoupling between order generators and order processors. Custom routing rules can be set up for individual components or groups of components in a centralised fashion, making it easier to configure networks involving many components. A further system functionality factored out by the Order Manager is dynamic data persistency: as requests, responses and updates from and to all the components flow through it, the Order Manager stores these data into the database. Finally, the Order Manager also reduces the number of connections required to attach N order generators to M order processors: the N*M links required to link each source with each destination is reduced down to N+M.

In commercial algorithmic trading systems, orders are ultimately matched by electronic exchanges such as London Stock Exchange (LSE), Chicago Board of Trade (CBOT) and Tokyo Commodity Exchange (TOCOM). These market exchanges are separate entities from the financial institutions that act as market players, and their systems are interconnected by a high-speed WAN link. Usually, hedge funds and investment banks connect to the exchanges through legacy software or in-house solutions, often referred to as *market gateways*. Market gateways do not implement any matching logic: they act as logical routers of trading data traffic between market exchanges and their subscribers, decoupling the infrastructure of the exchange from that of the institution. OpEx, on the other hand, does not offer market connectivity to real world exchanges, therefore it lacks a separate market gateway component and implements the order matching logic in the Exchange component. Depending on global routing rules and the routing instructions attached to them, the Order Manager forwards the orders received from the GUIs and the Agent Hosts to the appropriate Exchanges. In a simple configuration, all the orders produced by the order generators in an instance are routed to one Exchange, which processes them according to the rules of the CDA. When it starts, the Exchange spawns as many order book processors as the instruments configured in its application settings. Each order book processor runs in a separate thread, so that multiple instruments can be traded simultaneously across one instance. In line with commercial market exchanges, this allows the development and testing of pair trading strategies (i.e. two instruments traded at the same time) and portfolio trading (i.e. a basket of any number of instruments traded simultaneously). Having one thread per order book processor also allows fine tuning for each instrument, which is useful if some instruments require more processing power than others. This can be achieved by assigning more CPUs and/or a higher priority to the more liquid instruments and allocate the illiquid ones to one CPU. Within an order book processor, order requests are handled sequentially in a FIFO fashion: no order requests can be processed until the current request has finished processing. This transactional mechanism guarantees the order book is consistent before and after every order allocation. When a request is processed, the order book is usually altered and the updated copy is both stored internally and published onto the Market Data Bus, so that all the market participants are informed of the changes. The status of the order is also updated according to the order state machine and the updated copy order is sent back to the Order Manager, which in turn saves it and forwards it back to the originator. Unlike commercial systems, where the only public data available is the order book, the Exchange also publishes *quote* data: every time a creation or amendment request is processed, the market participants are informed about the order price, the quantity, and whether the order was accepted (partially or completely filled) or rejected (no trades were made and the order was added to the order book). This mechanism was put in place for compatibility with the automated agents we analysed, which use quote data as a vital part of their calculations.[18] Finally, if a trade happens, the Exchange sends the relevant trade data to the Order Manager, to be stored in the database: for both counterparties, trade data include trade price, quantity traded, and counterparty name.

OpEx uses a database as a centralized repository of data, both to retrieve the configuration settings of the components and to store the data produced during the simulations. This choice was made for several reasons. First, the system benefits from most of the built-in features of DBMSs: concurrency, necessary when multiple components need to write data simultaneously; atomicity, which guarantees that data are consistent before and after each transaction; backup and recovery, crucial to take snapshots of the data at particular times and to import and export data across different installations of the system; indexing and query, indispensable for data manipulation and to generate reports. Second, the data in the database are well-structured and strongly typed: this saves a significant amount of time when the data need to be analysed, since there is no need for pre-processing. Third, both configuration data and results for all components are retrieved from and stored into one location: setting up the parameters and collecting results for each component from its physical network location would constitute a much more time-consuming and error-prone process.

Component	Order	Market	Trade	Quote
GUI	✓S×L	×S✓L	×S×L	×S×L
Order Manager	✓S✓L	×S×L	×S✓L	×S✓L
Exchange	×S✓L	✓S×L	✓S×L	✓S×L
Agent Host	✓S×L	×S✓L	×S×L	×S✓L

Table 2: The distribution of data services (S) and listeners (L) plug-ins across producers and consumers of dynamic data. Tick signs and crosses represent the presence or the absence of the particular plug-in, respectively.

The requirements outlined here imply intensive communication among the components, mostly in the form of messaging: sending and receiving pieces of structured data across remote network locations. Since both *push* and *pull* communication are employed according to the

[18] It can be easily proved that only partial quote data can be inferred from market data. Suppose that there are only two price levels in the order book: b on the buy side, and a on the sell side, both for one unit only. If after a market data update only one of the two levels is left, for example a, then we are sure that the order at price b was traded (in absence of cancellations), but we cannot know the price at which the aggressive order was sent: any sell order at a price less than or equal to b would satisfy the conditions to generate a trade.

Component	Order	Market	Trade	Quote
GUI	✓S×L	×S✓L	×S×L	×S×L
Order Manager	✓S✓L	×S×L	×S✓L	×S✓L
Exchange	×S✓L	✓S×L	✓S×L	✓S×L
Agent Host	✓S×L	×S✓L	×S×L	×S✓L

Table 3: The distribution of data services (S) and

specific task, we adopted a hybrid messaging paradigm between *publish/subscribe* (pub/sub) and *Remote Procedure Call* (RPC). Commercial high-frequency trading systems rely on different messaging solutions: some of them use custom libraries developed in-house; some others employ third-party libraries. We excluded the option to develop from scratch a messaging library built on top of sockets for timing reasons, and instead reviewed a few available libraries, summarised in Table 1. All the libraries satisfy the messaging requirements of OpEx, however we chose MSMQ because it is distributed with Microsoft Windows and it is part of the .NET framework, which we preferred to other technologies for the richness and ease of use of its IDE, graphical libraries and tools. The MSMQ API exposes basic methods to manipulate message queues and exchange messages between remote network locations, through which we built the hybrid communication mechanism mentioned above. Dynamic data belonging to a particular category is sent by a specific *data service*, and received by the corresponding *data listener*. Thus OpEx employs services and listeners for order data, market data, quote data and trade data. Both services and listeners run as in-process-plug-ins of the different applications, according to the specific application needs, as shown in Table 2, and comprise a remote outgoing queue, to which messages are sent, and a local incoming queue, from which messages are received. The order data service embedded in GUIs and Agent Hosts points to the Order Manager's listener, so that an order corresponds to a message sent by the order generator and received by the Order Manager. Once the message is received, the listener acknowledges the reception to the data service plug-in by sending an order message back to the originator. It is worth noting that, while the network location of the Order Manager has to be known by the order generators (in fact, it is found in the application settings retrieved at start-up), the Order Manager learns the location of GUIs and Agent Hosts as they start sending orders, dynamically building a data structure that associates a particular instance of an application to its network location. The Order Manager also embeds an order data service, as it propagates the order messages to the appropriate destination(s) according to both the global routing rules and the particular location specified in the order message. Messages are sent by the data service to the incoming queue of the order data listener plug-in built in the Exchange(s), which responds by sending messages back to the incoming queue of the order data service of the Order Manager. In this case, the location of the Exchange(s) is loaded at start-up by the Order Manager as part of its configuration settings. The way market data plug-ins are attached to components follows a similar pattern: the Exchange, originator of market data traffic, implements a market data service, while GUIs and Agent Hosts subscribe to the data by employing a market data listener. However, unlike order data that are transmitted from GUI to Order Manager and from Order Manager to Exchange, market data updates are announced publicly using *multicast*.[19] We preferred this solution to the iterative point-to-point

[19] A multicast is a single stream of data (i.e., a set of packets) that is transmitted simultaneously to selected multiple hosts who have joined the appropriate multicast group. In contrast to broadcasts (which are used on some LANs), multicast clients receive the data stream only if they have previously elected to do so (i.e., by joining the specific multicast group address). Multicasting is suitable for the situation in which there is a large amount of information to be transmitted to various (but usually not all) hosts on

alternative because market data update more frequently than any other dynamic data category in the system and because they are of interest for GUIs and Agent Hosts, the two applications that populate an instance of OpEx the most. Thus, transmitting updates only once avoids data replication and network packet collisions, reduces bandwidth occupancy by a factor equal to the number of listeners, and ultimately favours scalability. Trade data service and listener are implemented in Exchange and Order Manager respectively, so that information can flow from the former to the latter in order to be stored in the database. Quote data listener and service are equally embedded in Exchange and Order Manager for the same reason; Agent Hosts also include a listener for quote data, which is used intensively in most of the calculations performed by the algorithms covered in this paper.

a network. It is commonly used when distributing real time audio and video to the set of hosts that have joined a distributed conference. It is also used by video servers for sending out networked television channels.

5. New experiments

In this section we present results from a new series of artificial trading experiments between humans and agents, using the OpEx framework, where we begin to explore the performance of well-known robot trading algorithms in experimental settings that are closer to the reality of the global financial markets: we move away from the artificial constraint of regular simultaneous replenishments of currency and stock, using instead a continuous drip-feed; we also explore the extent to which the outperformance of robots over humans is an effect of the robots' greater speed.

5.1 Methods

OpEx was used by two of us (Szostek & Cartlidge) as an experimental platform to conduct a series of experiments between humans and robots trading in an artificial market, over April-May 2011. Following the structure used previously (De Luca & Cliff, 2011a, 2011b), each experiment involved six human traders (three buyers and three sellers) and six robot traders (three buyers and three sellers). Human participants were seated around a rectangular table with buyers on one side and sellers opposite. Before starting each experiment, participants were given a brief introduction to the rules of the market and allowed some time to familiarize themselves with the Sales Trading GUI (this briefing and the short tutorial typically took less than 10 minutes). When training was complete the market was reset to ensure the removal of any residual orders in the system. Participants were then notified that the experiment was about to begin and reminded that their aim was to maximize profit. Each experiment lasted 20 minutes,20 during which the market was continuously open for trading.

Having the market be continuously open throughout the experiment is a significant difference from the usual "periodic" experiment design where the market repeatedly opens, trades for a period (or "day"), closes, and then re-opens with fresh replenishment for all traders for each trading period; i.e., the design that was used in Smith's (1962) original experiments. The move away from periodic experiment designs to continuous trading is a step nearer to the situation in the real-world financial markets where buyers and sellers continuously and asynchronously enter and leave the market at all times, and where trading activity can "follow the sun" around the planet, with at least one major trading venue open somewhere in the world at any time of day or night. The design of the continuous-market experiments that we use here is taken from earlier experimental economics work performed at Hewlett-Packard Labs by Cliff & Preist in 1998 (reported in Cliff & Preist, 2001), the design of which was in turn directly inspired by Smith & Williams (1983).

At the start of each experiment, the market was empty and traders had no order assignments. New assignments were periodically sent to traders following the permit schedule shown in Table 4, with each trader receiving 6 assignments every 170 seconds. From Table 4, we see the limit price of each assignment and the time step that the assignment is sent to each trader: after 10s, for example, human Buyer 2 and robot Buyer 2 each receive a buy order assignment with limit price 340; and human Seller 2 and robot Seller 2 each receive a sell order assignment with limit price 60. After 170s the permit schedule repeats, producing 7 full permit cycles during the 20 minutes experiment (no assignments were sent in the final 10s).

[20] One experiment (UoB1-AA-slow) was actually run for the longer duration of 30 minutes, but for consistency with the other experiments described here, we only analyze and discuss the results for the first 20 minutes of that experiment.

	1	2	3	4	5	6
Buyer1	350 (0)	250 (4)	220 (7)	190 (9)	150 (14)	140 (16)
Buyer2	340 (1)	270 (3)	210 (8)	180 (10)	170 (12)	130 (17)
Buyer3	330 (2)	260 (4)	230 (6)	170 (11)	160 (13)	150 (15)
Seller1	50 (0)	150 (4)	180 (7)	210 (9)	250 (14)	260 (16)
Seller2	60 (1)	130 (3)	190 (8)	220 (10)	230 (12)	270 (17)
Seller3	70 (2)	140 (4)	170 (6)	230 (11)	240 (13)	250 (15)

Table 4: Order permits for traders showing limit prices of each order assignment. Traders receive six assignment types during each cycle of 170s. Numbers in brackets show the time step that each order assignment is sent, in 10-second multiples – so (4) means 40s.

For each permit cycle, the sequence of limit prices for order assignments were arranged in an arithmetic progression (hence, assignments early in the cycle were easiest to execute, having either high buy, or low sell, limit prices). Received assignments were queued in a traders' personal assignment queue until the trader decided to execute the order. Traders were able to work orders from their assignments in any order and at any time, thus enabling them to have multiple simultaneous orders on the exchange order book. Unable to make a loss, traders were forced to submit buy (sell) orders at less than (greater than) or equal to the limit price. Profit was then calculated as the difference between execution price and limit price for each order. For fairness, the maximum theoretical profit available to each player was deliberately kept equal. Figure 5.1 Figure shows the demand and supply schedules generated by the permit schedule in Table 4. It can be seen that demand and supply is symmetric and unbiased, with a theoretical market equilibrium price P_0=200.

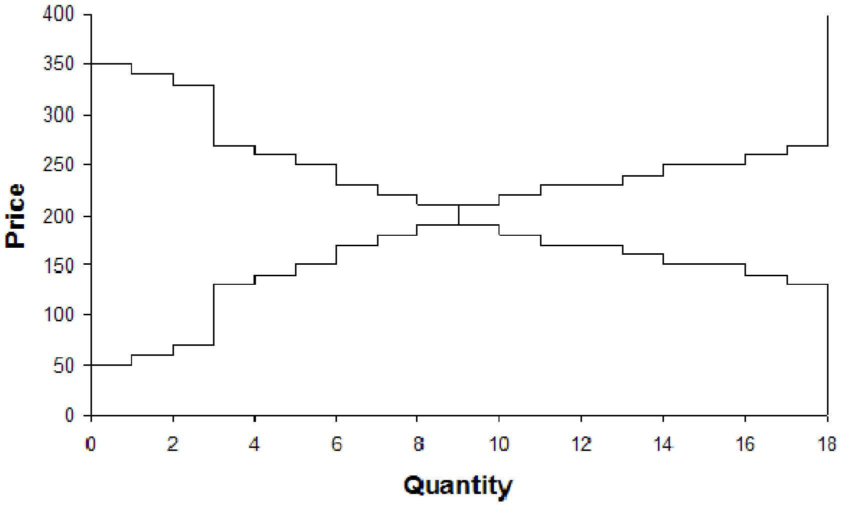
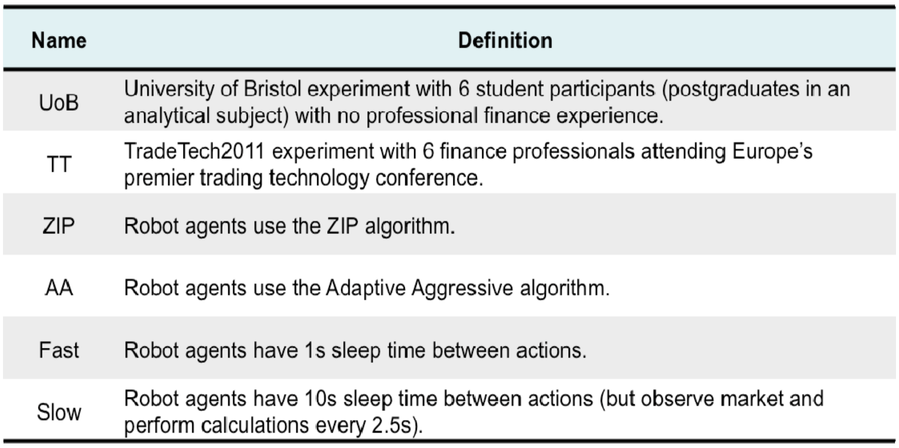


Figure 5.1: Demand and supply schedules for traders in the market. The theoretical market equilibrium price is P_0=200.

Experiments were run under several conditions (refer to Table 5 for a summary). For each run, robot agents were all of the same type, either AA or ZIP, and were set at the same speed, either fast or slow. Under the slow condition, robot agents were set to wake from sleep and act every 10s, and were additionally enabled to perform internal calculations (but no actions) every 2.5s. Under the fast condition, robots were able to both calculate and act every 1.0s. Human participants were separated into two groups, *experienced* finance professionals and *inexperienced* non-financial postgraduate students. Experiments using experienced participants were run at *TradeTech2011*, Europe's premier trading technology conference for industry professionals (held at the Excel Centre, London, April 2011). Participants were all registered conference delegates who responded to adverts and public announcements of the experiment during the conference. To encourage participation and subsequent competition, the one participant with the greatest efficiency (profit as a proportion of maximum theoretical profit) at the end of the experiment was rewarded with an iPad2 tablet computer (valued at £400), and the five remaining participants received no reward at all. Experiments using inexperienced participants were run at the University of Bristol during April and May 2011 using postgraduate students in non-financial but analytical subjects (i.e., these students had skills suitable for a professional career in finance, but did not work in the industry, and had no professional financial experience). For inexperienced experiments, in each experiment all six participants received £20 in cash for participating, with a further £40 cash bonus awarded to the winner (most efficient), and £20 cash bonus for second place.[21]



Name	Definition
UoB	University of Bristol experiment with 6 student participants (postgraduates in an analytical subject) with no professional finance experience.
TT	TradeTech2011 experiment with 6 finance professionals attending Europe's premier trading technology conference.
ZIP	Robot agents use the ZIP algorithm.
AA	Robot agents use the Adaptive Aggressive algorithm.
Fast	Robot agents have 1s sleep time between actions.
Slow	Robot agents have 10s sleep time between actions (but observe market and perform calculations every 2.5s).

Table 5: Summary of experiment conditions.

5.2 Results

We present here the results from a series of experiments performed during April and May 2011, grouped into five conditions. Table 4 lists our acronyms for the various conditions that our experiments involved: the robots were either AA or ZIP; inspired by the IBM 2001 study, their sleep-wake cycle was either "Fast" (1.0s sleep) or "Slow" (10.0s sleep); and the human subjects were either postgraduate students from the University of Bristol ("UoB"), or finance professionals who attended the *TradeTech2011* trading-technology conference in London ("TT"). The specific combinations we tested were: UoB-AA-Fast; UoB-AA-Slow; UoB-ZIP-Fast; TT-AA-Fast; and TT-ZIP-Fast. Experiments were run twice for all conditions.

Figure 5.2 shows the time-series of quotes (buy and sell orders submitted to the exchange) for experiment TT1-AA-Fast, with human quotes in red and agents in blue. This data set is selected as a representative example of market activity; for full results from all experiments, refer to Appendix A. Filled and open-faced markers show accepted and rejected offers respectively, with triangles representing bids and squares representing offers. Vertical lines denote the start and end of each permit cycle (170s) and the dashed horizontal line shows the theoretical market equilibrium price, P_0*=200*.

[21] Financial assistance in funding the prizes used to incentivize the participants in our experiments was provided by Syritta Algorithmics Ltd: we are very grateful to Syritta for their support of our research.

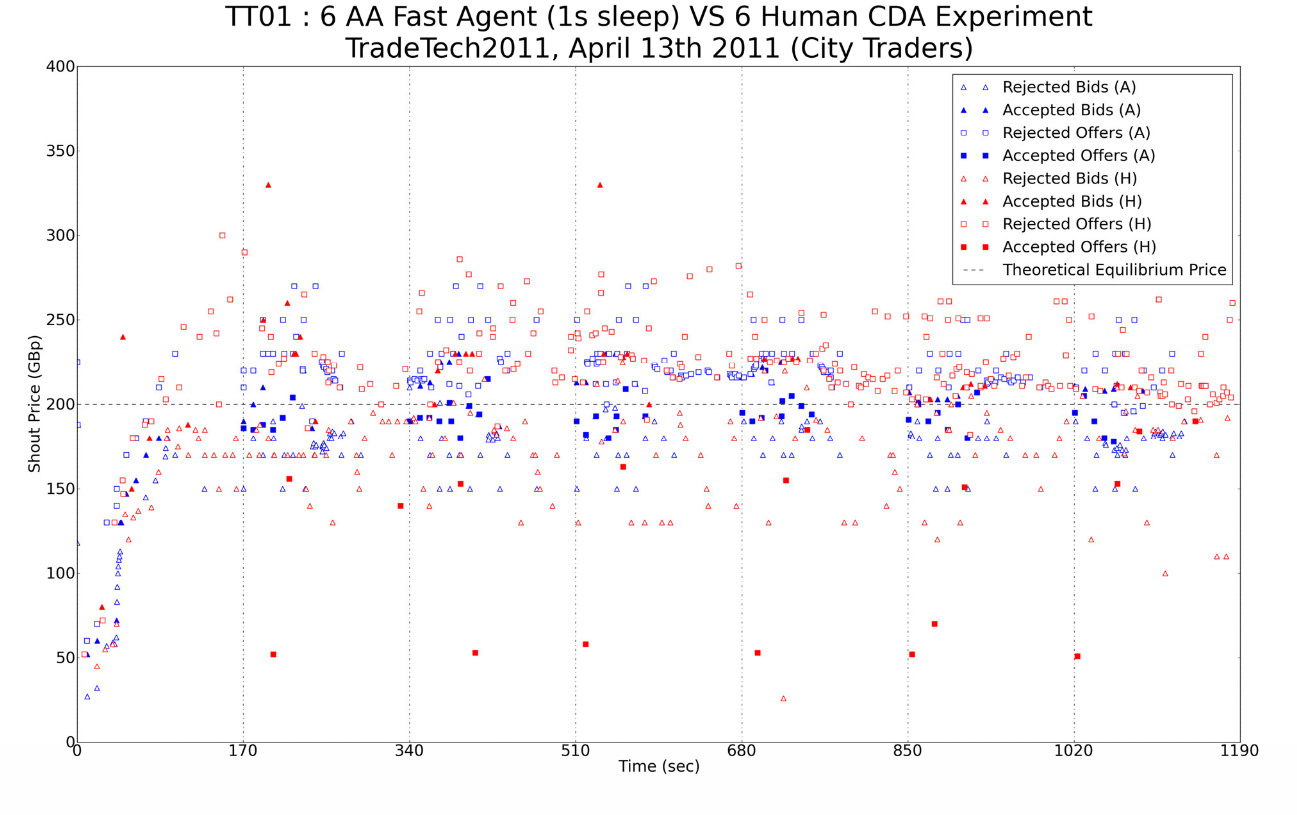


Figure 5.2: Series of quotes from participants in experiment TT1-AA-Fast, with human data red and agent data blue. See text for explanation and discussion.

It can be seen that there is regular market activity throughout the experiment from both humans and agents. Initially, the prices quoted are consistently below P_0, but by permit-cycle 2 they become roughly symmetric about P_0 and remain this way until market close, with the traders' transaction prices converging on the theoretical equilibrium. Interestingly, near the beginning of each cycle, there appears a filled red square at a price of approximately p=50. This represents a human seller having a quote accepted at a price well below the equilibrium P_0=200. However, this does not imply that the seller is executing at a foolishly low price. Rather, they are examples of aggressive *market orders*: bids (offers) placed on the exchange at a relatively high (low) price in order to execute at the current best sell (buy) price, thus effectively taking the market price. Since London Stock Exchange rules apply, matched orders execute at the price of the order that arrived first. Thus, a market ask with a very low limit price will often execute at a much higher value. A clearer pattern emerges when we consider the time-series of execution prices from the same experiment, shown in Figure 1.3. Here, we see that after an initial exploratory period, trades execute at a price relatively close to P_0.

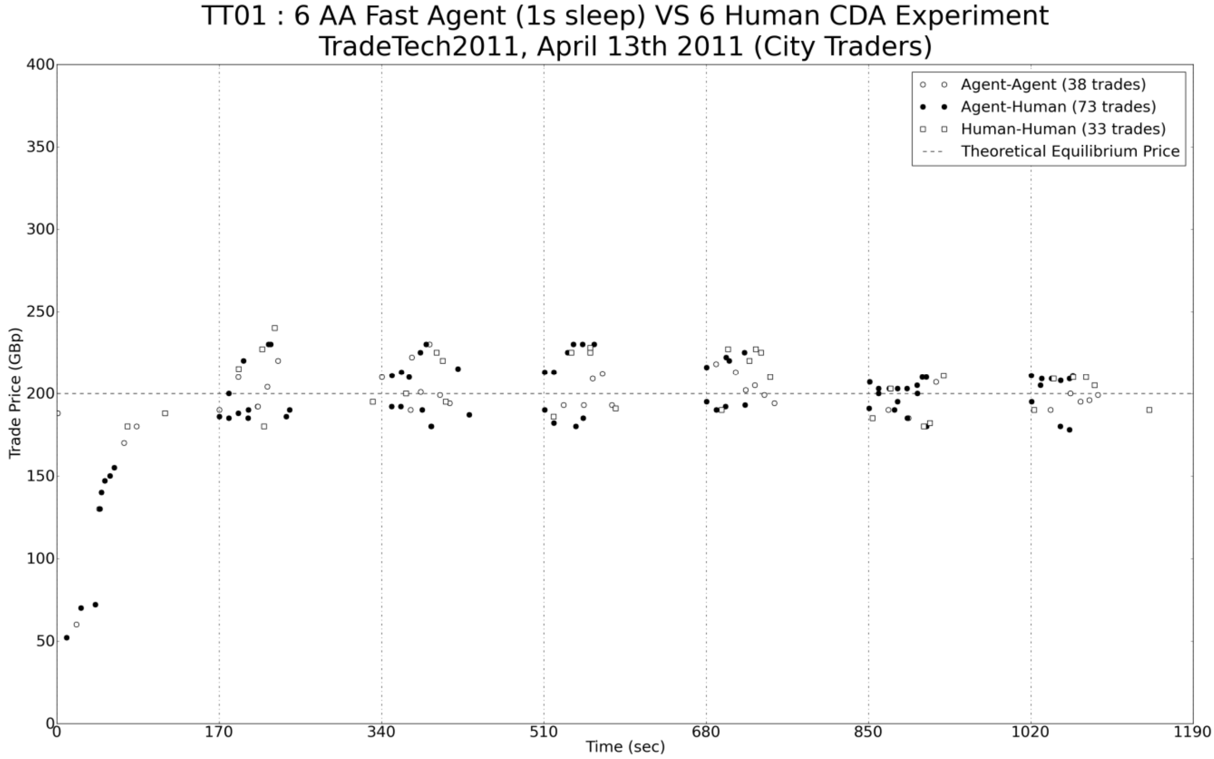


Figure 1.3: Series of trades from participants in experiment TT1-AA-Fast. Trades between humans are shown by clear squares; between agents by clear circles; and between a human and agent by filled square. Vertical dashed lines show the beginning of each permit cycle; the horizontal dashed line represents the theoretical market equilibrium price, *P_0*.

From Figure 1.3, it is clear that the majority of trading activity appears in clusters at the beginning of each permit cycle. This is an artefact of the permit schedule (Table 4). At the start of the permit cycle, the easiest assignments to trade are sent, i.e., buy (sell) orders with high (low) limit prices relative to P_0. Between 90-170s, traders receive assignments with buy (sell) limit prices below (above) P_0 (i.e., demand and supply to the right of equilibrium in Figure 5.1). Unable to form a natural match, these order assignments are difficult to trade and sit on the order book waiting to be filled. It is not until the start of a new cycle that these orders can be filled and a burst of activity ensues.

From Figure 1.3, it appears that trading activity in each permit cycle progressively converges toward P_0. Price convergence toward equilibrium is a theoretical property of *ideal markets* in economics and can be quantifiably measured using Smith's alpha (α) coefficient: calculated as the root mean square deviation of trade prices from the equilibrium price (i.e., the standard deviation of trade prices around P_0 rather than the mean). A low value of α is a desirable property, describing a stable market trading close to equilibrium. In Figure 5.4, mean α is plotted for each permit-replenishment cycle under each experimental condition.

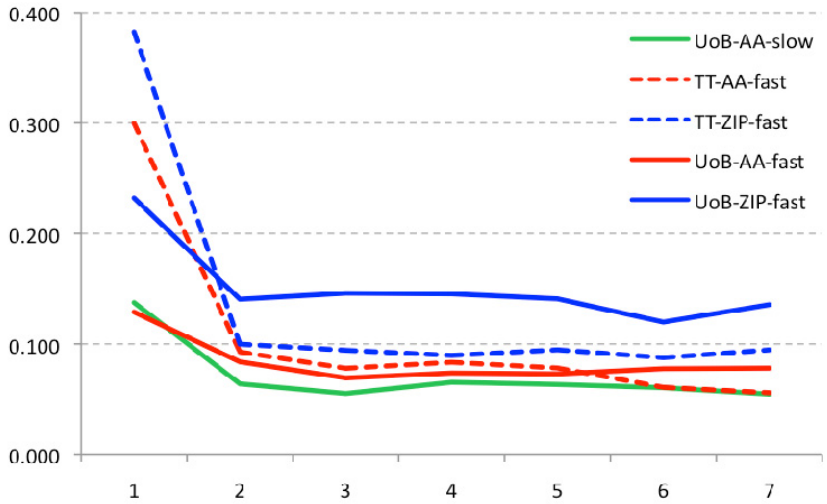


Figure 5.4: Mean value of Smith's alpha measure (vertical axis), i.e. the root mean square deviation of transaction prices from the underlying theoretical equilibrium price *P_0*; for each 170-second replenishment cycle, over a total of 7 cycles (horizontal axis), for each experiment condition. In all experiments, the market settles to a near steady-state condition within the first 170-second cycle. Data from TT experiments are shown as dashed lines; data from UoB experiments are shown as solid lines. Error bars omitted for clarity.

Under all conditions, α is highest during the initial cycle. Intuitively, this is due to the volatile activity of the market shortly after opening as traders probe demand and supply at varying price levels. By the second cycle, α drops significantly, indicating that the market is trading much closer to equilibrium. Under most conditions, the market then remains relatively stable, with little variation, suggesting that the cyclical nature of the experimental setup counteracts further convergence once the initial learning period is over. Only the combination of experienced financial professionals and the AA algorithm (TT-AA-fast) produces a progressive pattern of convergence, with α steadily reducing throughout. Overall, ZIP agents (blue) produce higher α than AA (red and green lines), suggesting AA encourages more stable market convergence. Similarly, finance professionals (dashed lines) appear to encourage lower α than inexperienced students (solid lines). Finally, reducing the reaction times of agents (UoB-AA-slow, solid green line) produces lower α than faster agents under the same conditions (UoB-AA-fast, solid red line); although by cycles 6 and 7 the α scores for professionals (TT-AA-fast) are indistinguishable from those for UoB-AA-slow. This interesting result suggests that when agents are able to react quickly to market data there is an increase in price volatility and that, when the activity of robot traders is reined in, even inexperienced human traders can produce relatively high market stability (the lowest α under all conditions). To explore this possibility, we will now take a more detailed look at the behaviour of the traders in these experiments.

Table 6 displays the proportion of trades executed by each of the four counterparty pairings: agent<human, human<agent, human<human, and agent<agent; where X<Y denotes: "Y hits (executes against) an offer previously posted on the order book by X".

	A<H	H<A	H<H	A<A	*<Human	*<Agent
TT-AA-Fast	0.10	0.41	0.23	0.27	0.33	0.67
TT-ZIP-Fast	0.05	0.51	0.21	0.23	0.26	0.74
UoB-AA-Fast	0.16	0.28	0.25	0.31	0.41	0.59
UoB-ZIP-Fast	0.14	0.32	0.24	0.30	0.38	0.62
UoB-AA-Slow	0.13	0.48	0.17	0.22	0.30	0.70

Table 6: Execution counterparties as a proportion of all trades. X<Y denotes: "Y hits an offer previously posted on the order book by X"

Under all conditions, agents perform more execution hits than humans (final two columns). This dynamic can be more clearly seen by plotting the relative frequency of executions by counterparty. Figure 5.5 (UoB-AA-Fast) and Figure 5.6 (UoB-AA-Slow) show plots of these distributions against time since last order assignment, effectively displaying the response profiles of traders to new assignments (sent every 10s).

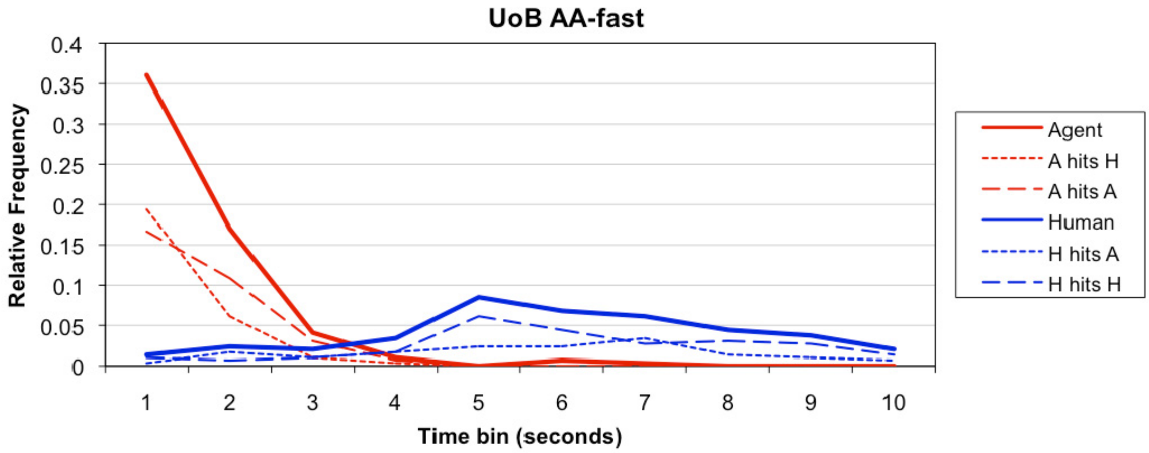


Figure 5.5: Relative frequency distribution of trade executions for UoB-AA-Fast.

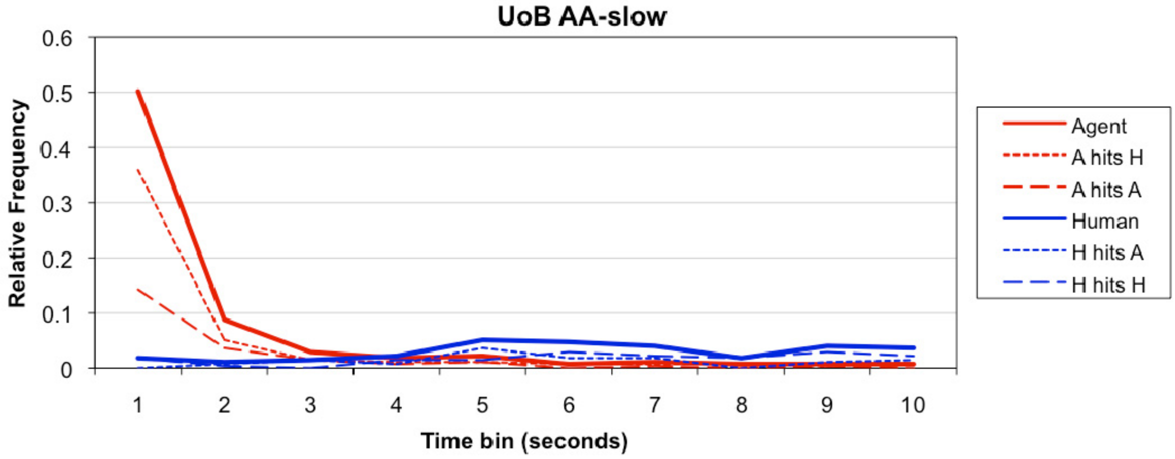
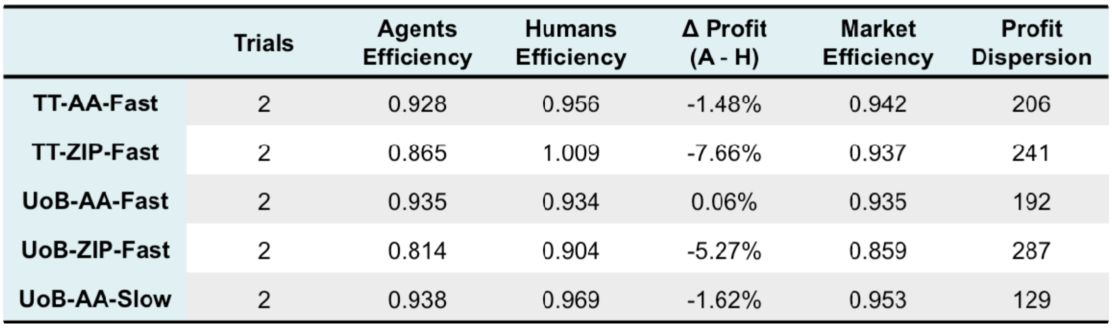


Figure 5.6: Relative frequency distribution of trade executions for UoB-AA-Slow.

We see that agents react more quickly than humans to new assignments, with a significant majority of agents' executions occurring within the first 2s. In contrast, humans have a more even distribution, with a modal maximum at approximately 5s. For all fast-agent conditions, distributions are qualitatively similar to Figure 5.5. However, when agents are slowed (Figure 5.6), we see that the majority of executions are agents hitting humans in the first 2s after a new assignment enters the market, with humans trading more passively throughout. When agents are fast (Fig.5.5) they are able to trade between themselves before humans can react to new market orders: A<A executions in the first 2s account for more than one quarter of all executions (red dash). However, this proportion of trades is reduced when agents are slowed (Fig 5.6). The combination of human passivity and reduced A<A executions when agents are slowed thus results in less market volatility and lower α.

However, Smith's α is only one market metric. We are also interested in the efficiency of the market (profit extracted by the traders, as a proportion of maximum theoretical profit) and the relative efficiency of different trading algorithms (see Table 7).



	Trials	Agents Efficiency	Humans Efficiency	Δ Profit (A - H)	Market Efficiency	Profit Dispersion
TT-AA-Fast	2	0.928	0.956	-1.48%	0.942	206
TT-ZIP-Fast	2	0.865	1.009	-7.66%	0.937	241
UoB-AA-Fast	2	0.935	0.934	0.06%	0.935	192
UoB-ZIP-Fast	2	0.814	0.904	-5.27%	0.859	287
UoB-AA-Slow	2	0.938	0.969	-1.62%	0.953	129

Table 7: Summary of results. For each experiment the table displays: the number of experimental repetitions; the average efficiency of agents and humans; the percentage difference between agents' surplus and humans' surplus; the market efficiency and the profit dispersion. Lower profit dispersion and higher mean efficiency values are better.

For all conditions, the relative ranking of agents and humans within each experiment were mixed. However, performance differences become clearer when comparing the mean scores for humans and agents. UoB-AA-Fast was the only condition under which agents generated greater profit than humans, although the difference was marginal. In terms of profit, both AA-Fast and ZIP-Fast performed better against inexperienced UoB humans than against experienced TT humans, suggesting that professional traders were tougher competitors than postgraduate students; also, against both students and professionals, AA-Fast performed better than ZIP-Fast. Profit dispersion, shown in the final column (lower values are better) suggests that AA-Fast consistently produces lower profit dispersion than ZIP-Fast. Overall, Table 7 suggests that AA is a more efficient trading algorithm than ZIP.

However, the data in Table 6 shows that, across all conditions, in our experiments the humans are more efficient than the agents: the mean efficiency for humans over the ten experiments is 0.95, while the mean efficiency for robots is 0.90. We applied a Robust Rank-Order (RRO) test (Feltovich, 2003) to the raw efficiency data from the ten experiments. The RRO is a nonparametric test that is better-suited to these experiments than the more commonly-used Wilcoxon-Mann-Whitney 'U' test. The outcome of our RRO test indicates that the difference between the humans' efficiency and that of the agents is significant at the 2.5% level (that is, $0.010<p<0.025$), and that hence the humans are more efficient in our experiments than the agents are. This is a surprising result, and is in stark contrast with the results from Das *et al.* (2001). Further experimentation and analysis is required to determine the precise reason for this. As it stands, because the primary difference between our experiments and those of Das *et al.* (2001) is the abandonment of the regular periodic replenishments, it seems plausible that the machine superiority in the CDA claimed by IBM may have been an artefact of their experiment design.

We also used the RRO to test for significance of difference between the human traders' efficiency scores in the TT and UoB experiments. That is, we wanted to see if the markets populated by finance professionals (TT) were more efficient than the naïve subjects (UoB). With the data that our experiments have generated thus far, a visual inspection indicates that it might be plausible that the market-efficiency of the TT markets are better than that of the UoB-market data, but the RRO does not find a significant result at 9% or less (i.e., p>0.090). It is possible that with a larger sample (i.e., more experiments), or with analysis of our existing results using a more sophisticated comparator metric than aggregate market efficiency, a significant difference in trading behaviour might be detected.

To explore that prospect, we analyzed the scores for Smith's α (i.e., root mean square deviation of transaction prices from the underlying theoretical equilibrium price) for all the AA-fast and ZIP-fast data summarized in Figure 5.4. We were interested in the steady-state behavior of the markets, in replenishment cycles 2-7, rather than the initial transient behavior that all our experiments show in cycle 1 as they settle to equilibrium conditions. We again used the RRO test, and our four key findings for cycles 2-7 are:

1. Over all "fast" experiments, the α scores of ZIP were worse than those of AA (p<0.0005).
2. Over all "fast" experiments, the α scores of UoB were worse than those of TT (0.05<p<0.1).
3. There was no significant difference between α scores of TT-AA-fast and UoB-AA-fast.
4. The α scores of UoB-ZIP-fast were worse than those of TT-ZIP-fast (p<0.0005).

Finding [1] reinforces the conclusion in (De Luca & Cliff 2011b): human-agent markets populated by AA traders perform better than those populated by ZIP traders. Finding [2] indicates that there is indeed a detectable difference between the trading behavior of finance professionals (TT data) and that of naïve subjects (UoB data). Findings [3] and [4] indicate that the difference identified in Finding [2] is due to the relative sophistication of AA over ZIP: in human-vs-ZIP markets, professional traders score better on α than naïve traders do, but in human-vs.-AA markets, that difference is no longer detectable. Precisely why this is so remains a topic for further research, although it is perhaps a moot point: our results indicate that AA clearly dominates ZIP, and so questions about the performance of ZIP are now largely of historical interest.

The final question we should address is the issue concerning the extent to which the market's scores for Smith's α metric change when the robot traders are slowed down in their response times. Currently we have data from two UoB-AA-Slow experiments. These results can reasonably be compared to the two sets of results we have from UoB-AA-Fast, or arguably (considering Finding 3, above) from the four sets of results we have from the combined UoB-AA-Fast and TT-AA-Fast experiments. Either way, it is safe to say that we are very firmly in the realms of small-sample comparative statistics: the choice here is between running (n=2, m=2) vs. (n=2, m=4) tests; the RRO test is only defined for (m>2, n>2). In Figure 5.7 we illustrate the mean and standard deviations of the α scores for UoB-AA-Fast and UoB-AA-Slow, in an expanded "close-up" form (this is the same data that was shown in Figure 5.4). As can be seen, the evidence we have thus far suggests that when the agents are slowed, the market's performance as measured by the α metric improves, but further experiments would be required to generate sufficient data to firmly resolve this issue.

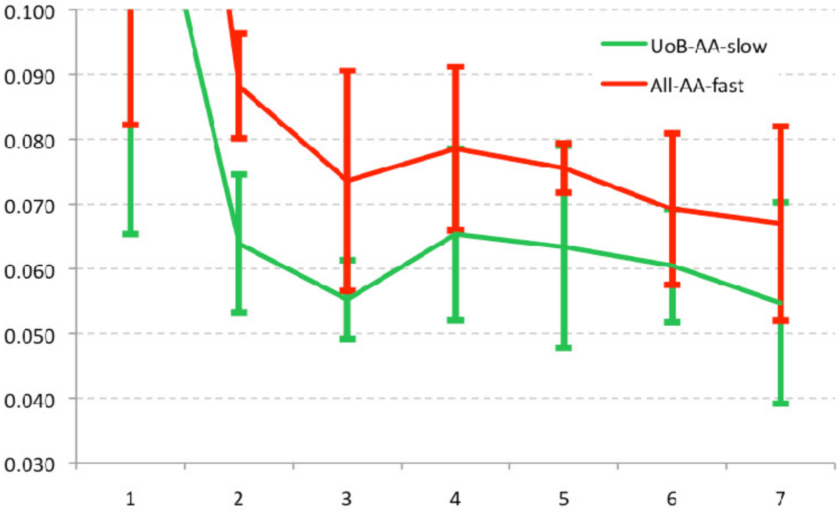


Figure 5.7: "Close-up" comparison of Smith's α metric for all our AA-Fast data (UoB and TT: *n*=4) and UoB-AA-Slow experiments (*n*=2). As in Figure 5.4, vertical axis is α; horizontal axis is replenishment cycle number over which α-value is calculated. Lower values of α are better; data plotted is mean α over *n* experiments; error bars indicate the standard deviation about the mean.

6. Discussion & conclusions

In this review document we have given, in Sections 2 and 3, a brief overview of the background literature relevant to the scientific study of interactions between human traders and "robot" trading algorithms (autonomous software agents), and summarized the surprisingly small number of papers that have addressed the core issue of interactions between humans and robots in auction markets. The only peer-reviewed papers that we have found are those published by the IBM researchers, by Grossklags & Schmidt, and by De Luca & Cliff, all of which rely very heavily on the experiment design introduced by Smith in his 1962 paper.

In an attempt to boost replication and further research in this area, in Section 4 we presented details of OpEx, a platform intended for release as a free, open-source resource that can become a common base for the research community to use and to extend over time.

Then in Section 5 we presented results from a new series of trading experiments between humans and agents, using the OpEx framework, where we started to explore the performance of well-known robot trading algorithms in experimental settings that are closer to the reality of the global financial markets (i.e., we moved away from the artificial constraint of regular simultaneous replenishments of currency and stock, toward a continuous drip-feed) and in settings where the effect of the robots' greater speed than human traders could be explored.

While it is difficult to draw cast-iron conclusions from the small set of new results presented here, our results do provide a teasing insight into the effects that agent strategy, agent speed, human experience and experiment-design have on the dynamics of heterogeneous human-agent markets. In general, agents perform less well (are less efficient) than humans across the majority of conditions in our new experiments. This is a surprising result that appears to be in conflict with previous OpEx experiments and the literature in general, where agents are generally found to outperform humans. However, unlike previous experiments, the set-up used here has been designed to more accurately reflect real financial markets, with a continuously open exchange, asynchronous order assignments distributed at a slow pace (one per trader every 30s), and the ability for traders to work multiple assignments in any sequence and at any time. This additional “realism” appears to favour humans, affording them more time to formulate a coherent strategy that can be used for the duration of the experiment. Humans are generally inclined to post orders on the book and wait for a match, whereas agents are more likely to send an aggressive market order soon after receiving an assignment. While this strategy favours agents in faster markets, it falters under the experimental conditions chosen here. We flag this as a warning to carefully frame results in the context of the experimental conditions in which they are generated.

Following this advice, we note that the success of humans observed here might not be a direct result of “realism”, but rather an artefact of the permit schedule or some other experimental condition. Thus, it is *not* our conclusion from the results gathered here that humans will outperform these agents in real financial markets. However, we *can* safely conclude that timing is an important factor in the CDA-market performance of agents relative to humans, with reduced reaction speed significantly hindering the performance of agents, and slower markets favouring humans. This raises the interesting possibility that the strong performance of agents versus humans in previous experiments is purely down to speed, rather than the algorithmic intelligence of the agents. Overall, across all metrics measured, the performance of AA is better than ZIP, resulting in higher efficiency, greater convergence to the theoretical market equilibrium (lower α) and lower profit dispersion. This is perhaps unsurprising, given that AA is essentially an improved version of the original ZIP algorithm. Our results also suggest that, when trading against ZIP, experienced finance professionals can perform better than inexperienced students, gaining a greater proportion of surplus profit and lowering α; but this performance advantage is not nearly so clear when the professionals trade against AA. Also, the evidence that we have available thus far indicates that slowing agents down improves the market dynamics in our experiments, but we have not yet generated enough experiment data to firmly resolve this issue. Finally, our results offer a hint that market volatility increases with the proportion of agent-agent executions. This interesting conjecture encourages us to draw the tentative comparison with high-frequency trading algorithms in real financial markets and the impact they may have on volatility, a subject with clear industrial relevance given current speculation surrounding the causes of the May 6th 2010 “Flash-Crash” (see e.g. Cliff & Northrop, 2011).

Perhaps though, the bigger message of this review is methodological: very significant amounts of research effort have been expended in experimental and behavioural economics over the years, and similar amounts in agent-based AI. The present-day financial markets are the scene of interactions between human traders and algorithmic trading systems in which billions of dollars change hands every day, and yet there is a staggeringly small number of academic peer-reviewed papers that explore interactions between human and robot traders in electronic markets. Moreover, for an entire decade there was been no peer-reviewed published attempt at replicating IBM’s results presented by Das *et al.* (2001), a paper that is widely and rightly seen as one of the key results in the literature. We hope that this Foresight driver review document, and the coming open-source release of OpEx, provoke greater research activity, with rapid replication of key results becoming the routine norm, and with moves towards widespread use of experiment designs that more accurately reflect current real-world scenarios, rather than the one design that a brilliant young economist happened to choose for his first CDA experiments, over 50 years ago.

Acknowledgements

We are grateful to the two anonymous peer-reviewers appointed by Foresight whose comments on an earlier version of this paper were very useful. Equipment costs for constructing De Luca's *Open Exchange* (OpEx) were met by a research grant from the UK Engineering and Physical Sciences Research Council (EPSRC) made to Dave Cliff, number EP/I001603/1. Primary financial support for Dave Cliff's research comes from EPSRC grant number EP/F001096/1; John Cartlidge is supported by EPSRC grant number EP/H042644/1; Charlotte Szostek's PhD research is funded by EPSRC via the Bristol Centre for Complexity Sciences. Marco De Luca's PhD research is partially supported by funds from the University of Bristol Merchant Venturers' School of Engineering. Financial assistance in the funding of the prizes we used as rewards and incentives for our experiments came from Syritta Algorithmics Ltd.. We are extremely grateful to Katia Wyatt and her colleagues at WBR for their help and assistance in allowing us to run experiments at TradeTech 2011, and of course to all the participants of all our experiments both at TradeTech and at the University of Bristol.

References

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

R. Axelrod (1984). *The Evolution of Cooperation.* Basic Books.

R. Axelrod (1997). *The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration*. Princeton University Press.

N. Bardsley, R. Cubitt, G. Loomes & P. Moffatt (2009). *Experimental Economics: Rethinking the Rules.* Princeton University Press.

S. Clearwater, editor, (1996). *Market-Based Control: A Paradigm for Distributed Resource Allocation.* World Scientific Press.

S. Clearwater, R. Costanza, M. Dixon, & B. Schroeder (1996). Saving energy using market-based control. In S. Clearwater, editor, *Market-Based Control: A Paradigm for Distributed Resource Allocation.* World Scientific Press, pp.253-273.

D. Cliff (1997). *Minimal-Intelligence Agents for Bargaining Behaviors in Market-Based Environments.* Technical Report HPL-97-91, Hewlett-Packard Laboratories. http://www.hpl.hp.com/techreports/97/HPL-97-91.pdf.

D. Cliff & J. Bruten (1998a). More than Zero Intelligence Needed for Continuous Double-Auction Markets. In *Proceedings of the 1998 IFAC/SCE/IFIP/IFORS Society for Computational Economics*

Symposium on Computation in Economics, Finance & Engineering: Economic Systems (CEFES'98). http://www.hpl.hp.com/techreports/97/HPL-97-157.pdf.

D. Cliff & J. Bruten (1998b). Less than Human: Simple Adaptive Trading Agents for CDA Markets.

In *Proceedings of the 1998 IFAC/SCE/IFIP/IFORS Society for Computational Economics Symposium on Computation in Economics, Finance & Engineering: Economic Systems (CEFES'98).* http://www.hpl.hp.com/techreports/97/HPL-97-155.pdf.

D. Cliff & C. Preist (2001). *Days Without End: On the Stability of Experimental Single-Period Continuous Double Auction Markets.* Technical Report HPL-2001-325, Hewlett-Packard Laboratories. http://www.hpl.hp.com/techreports/2001/HPL-2001-325.pdf

D. Cliff, D. Brown, & P. Treleaven (2011). *Technology Trends in the Financial Markets: A 2020 Vision.* Foresight Driver Review DR3.

D. Cliff & L. Northrop (2011). *The Global Financial Markets: an Ultra-Large-Scale Systems Perspective.* Foresight Driver Review DR4.

R Das, J. Hanson, J. Kephart, & G. Tesauro (2001). Agent-human interactions in the continuous double auction.*Proceedings IJCAI-01*, http://www.research.ibm.com/infoecon/paps/AgentHuman.pdf

D. Davis & C. Holt (1992). *Experimental Economics.* Princeton University Press.

M. De Luca & D. Cliff (2011a). Agent-Human Interactions in the Continuous Double Auction, Redux: Using the OpEx Lab-in-a-Box to explore ZIP and GDX. In *Proceedings of the Third International Conference on Agents and Artificial Intelligence (ICAART-2011).*Rome, January 2011. http://lscits.cs.bris.ac.uk/docs/DeLuca_Cliff_ICAART_CAMERA_READY.pdf.

M. De Luca & D. Cliff (2011b). Human-Agent Auction Interactions: Adaptive-Aggressive Agents Dominate. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI-2011).* Forthcoming. http://www.cs.bris.ac.uk/Publications/attachment-delivery.jsp?id=51.

J. Duffy (2006). Agent-Based Models and Human Subject Experiments. In L. Tesfatsion and K. Judd (editors) *The Handbook of Computational Economics, Volume 2.* Elsevier. Pp 949-1011.

S. Durlauf & L. Blume (2009). *Behavioural and Experimental Economics.* Palgrave MacMillan.
J. Epstein & R. Axtell (1996). *Growing Artificial Societies: Social Science from the Bottom Up.* MIT Press.

J. Epstein (2007). *Generative Social Science: Studies in Agent-Based Computational Modeling.* Princeton University Press.

N. Ehrentreich (2007). *Agent-Based Modeling: The Santa Fe Institute Artificial Stock Market Model Revisited.* Lecture Notes in Economics and Mathematical Systems. Springer.

N. Feltovich (2003). Nonparametric Tests of Differences in Medians: Comparison of the Wilcoxon-Mann-Whitney and Robust Rank-Order Tests. *Experimental Economics,* **6**: 273-297. http://www.uh.edu/~nfelt/papers/robust.pdf.

S. Gjerstad & J. Dickhaut (1998). Price Formation in Double Auctions. *Games and Economic Behavior*, **22**(1):1–29. http://www.research.ibm.com/iac/papers/pfdalnai.pdf.

D. Gode & S. Sunder (1993). Allocative Efficiency of Markets with Zero-Intelligence Traders: Markets as a Partial Substitute for Individual Rationality. *Journal of Political Economy*. **101**(1):119-137.

D. Gode & S. Sunder (1994). Human and artificially intelligent traders in computer double auctions. In K. Carley & M. Prietula (editors) *Computational Organization Theory*, Lawrence Erlbaum Associates.

J. Grossklags (2007). Experimental Economics and Experimental Computer Science: A Survey. In *Proceedings of the 2007 Workshop on Experimental Computer Science, ACM Federated Computer Research Conference*, pp.202–212. http://people.ischool.berkeley.edu/~jensg/research/expcs.html.

J. Grossklags & C. Schmidt (2003). Artificial software agents on thin double auction markets: a human trader experiment. In: *Proceedings of the IEEE/WIC International Conference on Intelligent Agent Technology (IAT 2003)*. Pp.400-407. http://www.sims.berkeley.edu/~jensg/research/agents_iat.html.

J. Grossklags & C. Schmidt (2006). Software agents and market (in)efficiency: a human trader experiment. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews),* **36**(1):56-67. http://www.sims.berkeley.edu/~jensg/research/agents_SMC.html.

F. Guala (2005). *The Methodology of Experimental Economics.* Cambridge University Press.

M. He, H. Leung, & N. Jennings (2003). A Fuzzy Logic Based Bidding Strategy for Autonomous Agents in Continuous Double Auctions. *IEEE Transactions on Knowledge and Data Engineering*, **15**(6): 1345-1363.

IBM (2006). *The Trader Is Dead: Long Live The Trader!* Report by IBM and The Economist Intelligence Unit, from: http://www-935.ibm.com/services/us/index.wss/ibvstudy/imc/a1024121?cntxt=a1005266.

N. Jennings, K. Sycara, & M. Wooldridge (1998) A Roadmap of Agent Research and Development. *Autonomous Agents and Multi-Agent Systems*, **1:**275–306.

N. Jennings, P. Faratin, A. Lomuscio, S. Parsons, C. Sierra, & M. Wooldridge (2001). Automated Negotiation: Prospects, Methods and Challenges. *International Journal of Group Decision and Negotiation*. **10**(2):199-215.

J. Kagel & A. Roth (1995). *The Handbook of Experimental Economics.* Princeton University Press.

M. Kearns & L. Ortiz (2003). The Penn-Lehman Automated Trading Project. *IEEE Intelligent Systems*, Nov/Dec 2003, pp.22-31. http://www.cis.upenn.edu/~mkearns/papers/plat-ieee.pdf.

B. LeBaron, W. Arthur, & R. Palmer (1999). Time-series properties of an artificial stock market. *Journal of Economic Dynamics & Control,* **23**(9-10):1487-1516.

B. LeBaron (2000). Agent-based computational finance: Suggested readings and early research. *Journal of Economic Dynamics and Control,* **24**:679-702. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.142.5342&rep=rep1&type=pdf.

B. LeBaron (2006). Agent-based computational finance. In K. Judd & L. Tesfatsion, editors, *Handbook of Computational Economics, Vol.2, Agent-Based Computational Economics*, North Holland. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.90.7228&rep=rep1&type=pdf.

P. Maes, editor (1990). Designing Autonomous Agents: Theory and practice from biology to engineering and back. MIT Press.

P. Maes (1994). Modeling Adaptive Autonomous Agents. *Artificial Life.* **1**: 135-162.

M. Miller & K. E. Drexler (1988a). Comparative ecology: A computational perspective. In B. Huberman (editor), *The Ecology of Computation.* Elsevier/North-Holland, pp.51-76.

M. Miller & K. E. Drexler (1988b). Markets and Computation: Agoric open systems. In B. Huberman (editor), *The Ecology of Computation.* Elsevier/North-Holland, pp.133-176.

R. Palmer, W. Arthur, J. Holland, B. LeBaron, & P. Tayler (1994). Artificial economic life: a simple model of a stock market. *Physica D* **75**:264-274.

C. Plott & V. Smith (2008). *The Handbook of Experimental Economics Results.* North Holland.

J. Rust, J. H. Miller, & R. Palmer (1993). Behavior of Trading Automata in a Computerized Double Auction Market. In D. Friedman & J. Rust (eds) *The Double Auction Market:*

*Institutions, Theories, and Evidence.*pp.155—198. Addison Wesley.

V. Smith (1962). An experimental study of comparative market behavior. *The Journal of Political Economy,* **70:**111–137.

V. Smith & A. Williams (1983). An experimental comparison of alternative rules for competitive market exchange. In M. Engelbrecht-Wiggans *et al.* (editors) *Auctions, Bidding, and Contracting: Uses and Theory. New* York: New York University Press, pp.307–334.

V. Smith (2005). *Bargaining and Market Behavior: Essays in Experimental Economics.* Cambridge University Press.

V. Smith (2006). *Papers in Experimental Economics.* Revised Edition. Cambridge University Press.

L. Sterling & K. Taveter (2009). *The Art of Agent-Oriented Modeling: Intelligent Robotics & Autonomous Agents.* MIT Press.

G. Tesauro & R. Das (2001). High-Performance Bidding Agents for the Continuous Double Auction. In *Proceedings of IJCAI 2001 Economic Agents Models and Mechanisms Workshop* and/or *Proceedings of the Third ACM Conference on Electronic Commerce,* pp.206-209. http://www.research.ibm.com/infoecon/paps/ec01_dblauc.ps.

G. Tesauro & J. Bredin (2002). Strategic sequential bidding in auctions using dynamic programming. *Proc. AAMAS-02*, pp. 591–598. http://www.research.ibm.com/infoecon/paps/gdx_aamas02.pdf.

L. Tesfatsion & K. Judd, editors (2006). *The Handbook of Computational Economics: Agent-Based Computational Economics.* North Holland.

I. Toft (2007). *Adaptive Bidding Agents for Single Seller and Continuous Double Auctions.* PhD Thesis, Department of Computer Science, University of East Anglia, UK.

P. Vytelingum (2006). *The Structure and Behaviour of the Continuous Double Auction.* PhD thesis, University of Southampton School of Electronics and Computer Science, Southampton, UK.

P. Vytelingum, D. Cliff, & N. Jennings (2008). Strategic bidding in continuous double auctions. *Artificial Intelligence,* **172:**1700–1729. http://eprints.ecs.soton.ac.uk/15895/2/crc.pdf.

W. Walsh, R. Das, G. Tesauro, & J. Kephart (2002). Analyzing Complex Strategic Interactions in Multi-Agent Systems” In *Proc. Game Theoretic and Decision Theoretic Agents Workshop(AAAI-02)*, Edmonton, Canada.http://www.research.ibm.com/infoecon/paps/MultiAgentGame.pdf

M. Wellman*,* A.Greenwald, P. Stone, & P. Wurman, (2001). The 2001 Trading Agent Competition, *Electronic Markets*, **13**(1):4–12. http://www.cs.utexas.edu/~pstone/Papers/bib2html-links/EM02.pdf.

M. Wooldridge & N. Jennings (1995). Intelligent Agents: Theory & Practice. *The Knowledge Engineering Review,* **10**:115-152.

Appendix A

In this appendix we show the full data from all the experiments described in Section 5.

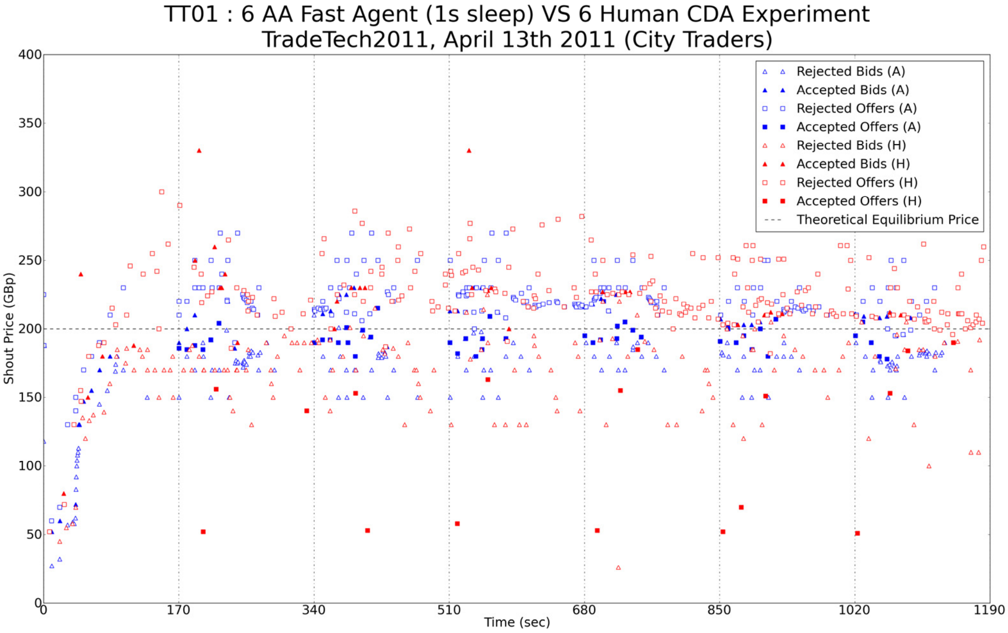


Figure A.1a: Series of quotes from participants in experiment TT1-AA-Fast.

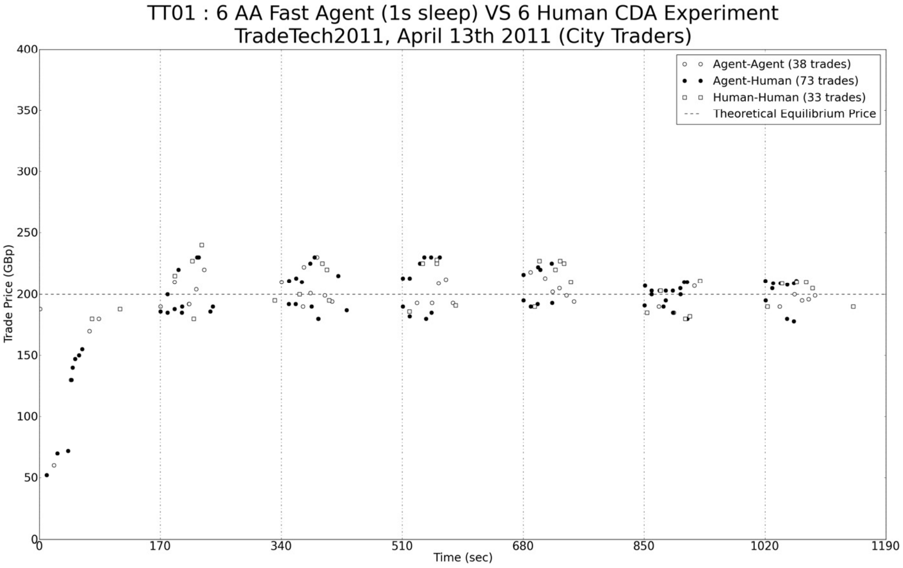


Figure A.1b: Series of transaction prices from participants in experiment TT1-AA-Fast.

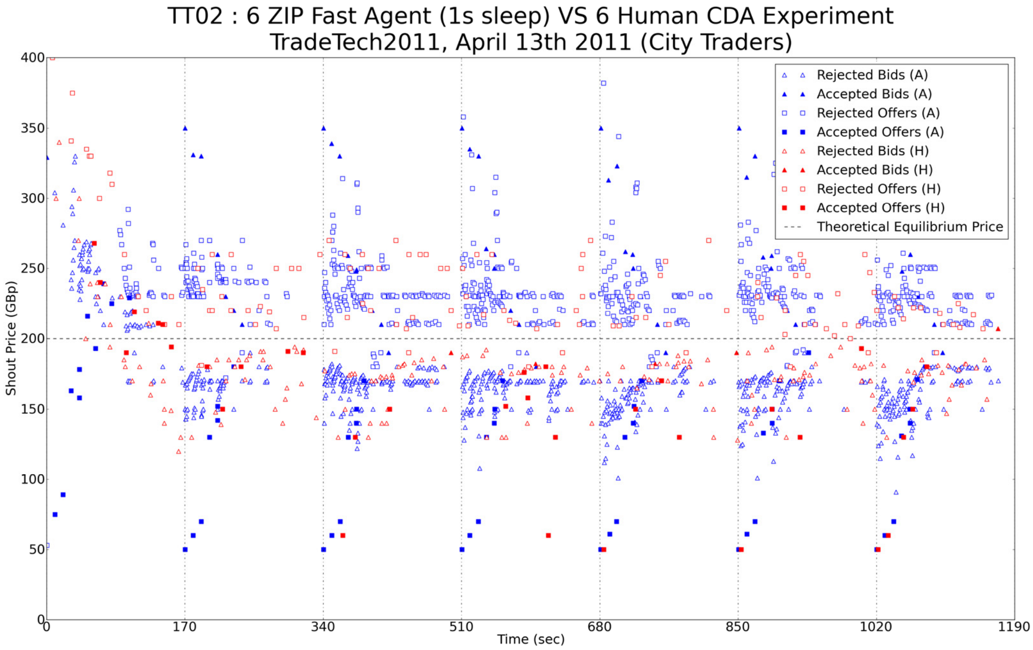


Figure A.2a: Series of quotes from participants in experiment TT2-ZIP-Fast.

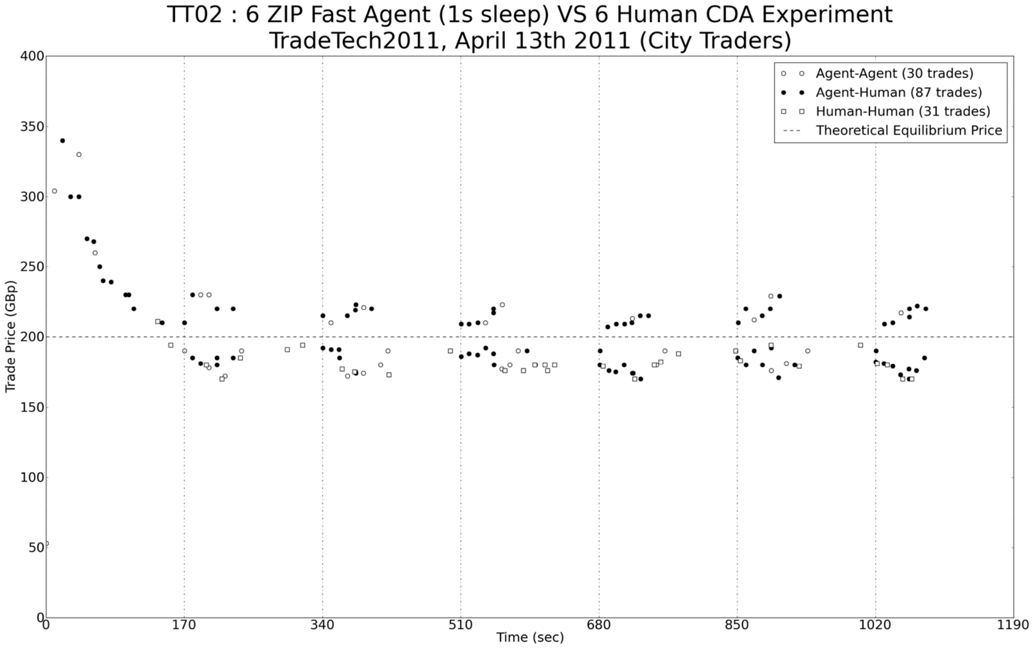


Figure A.2b: Series of transaction prices from participants in experiment TT2-ZIP-Fast.

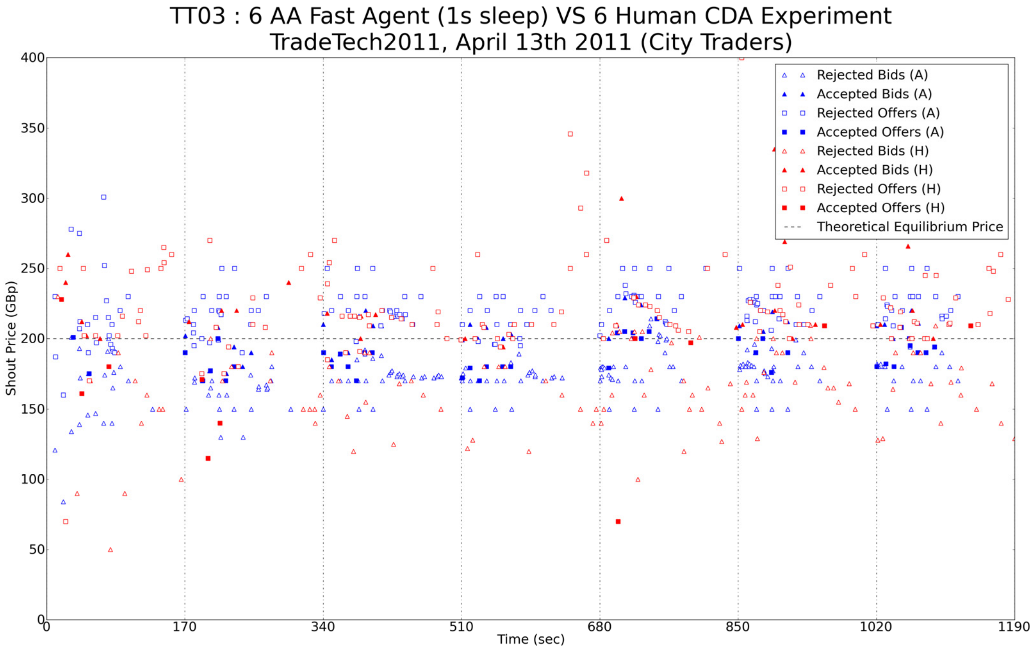


Figure A.3a: Series of quotes from participants in experiment TT3-AA-Fast.

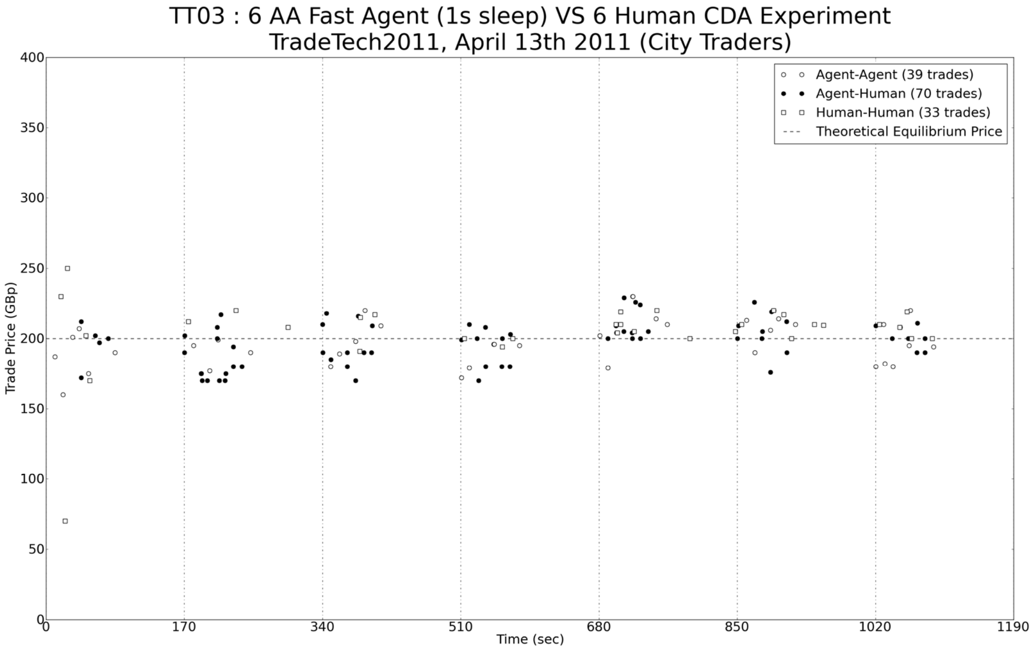


Figure A.3b: Series of transaction prices from participants in experiment TT3-AA-Fast.

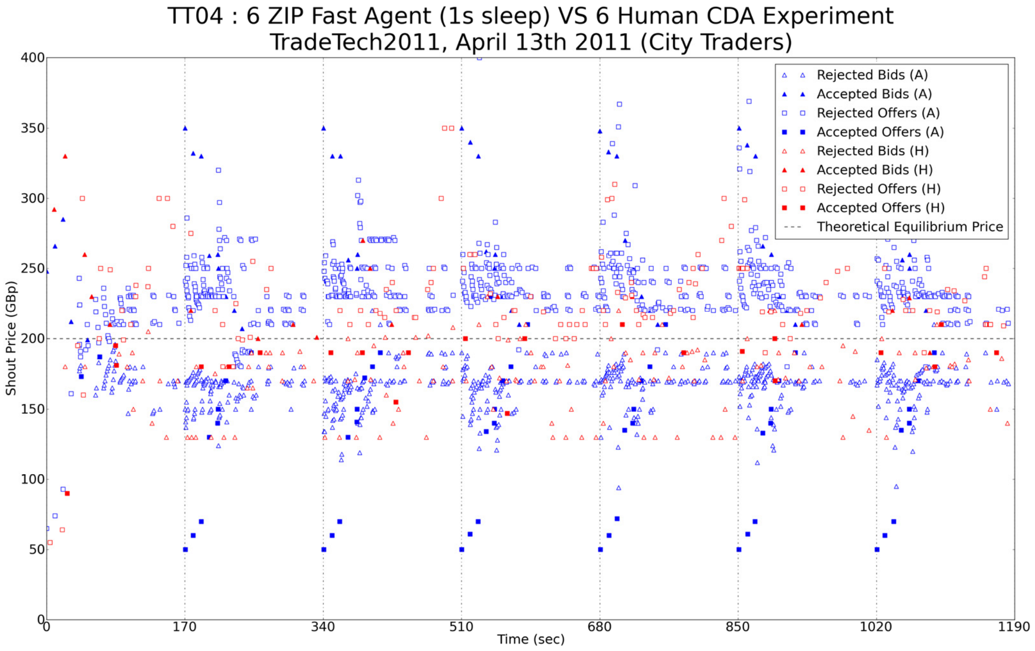


Figure A.4a: Series of quotes from participants in experiment TT4-ZIP-Fast.

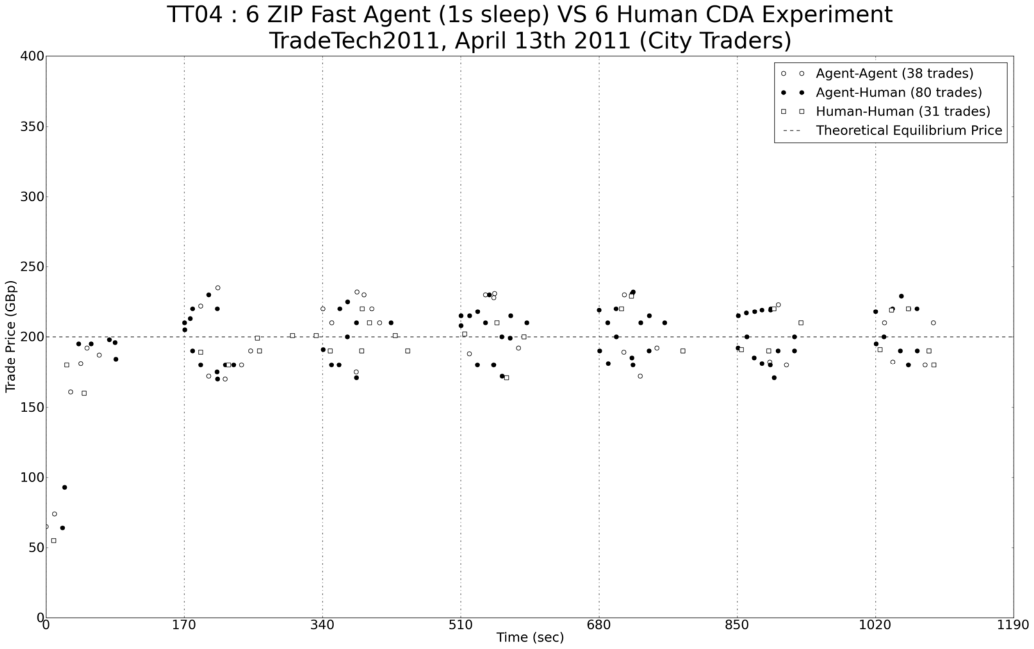


Figure A.4b: Series of transaction prices from participants in experiment TT4-ZIP-Fast.

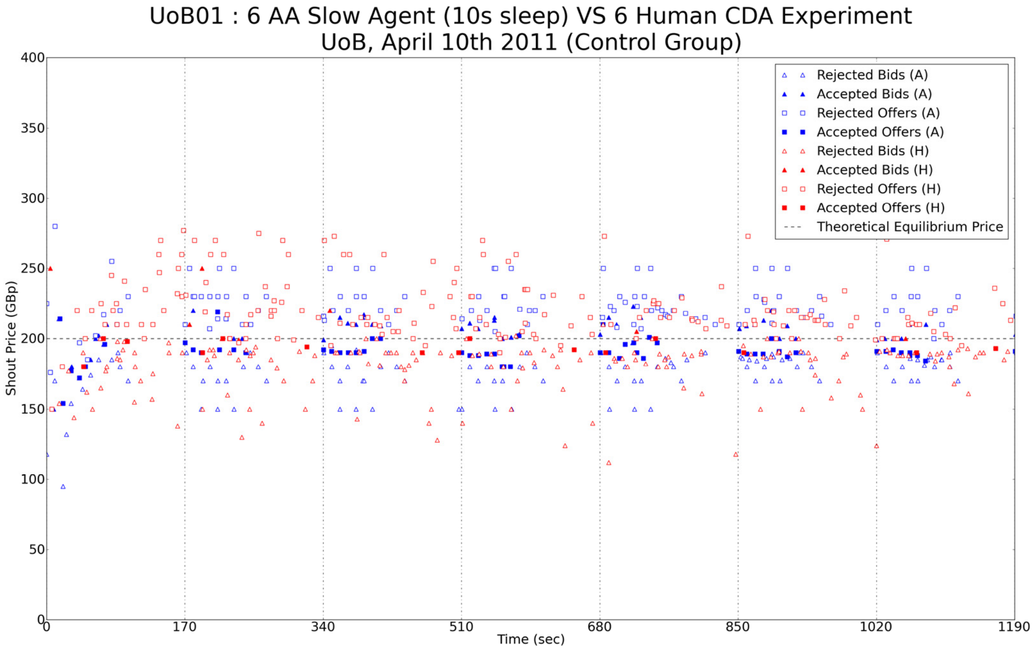


Figure A.5a: Series of quotes from participants in experiment UoB1-AA-Slow.

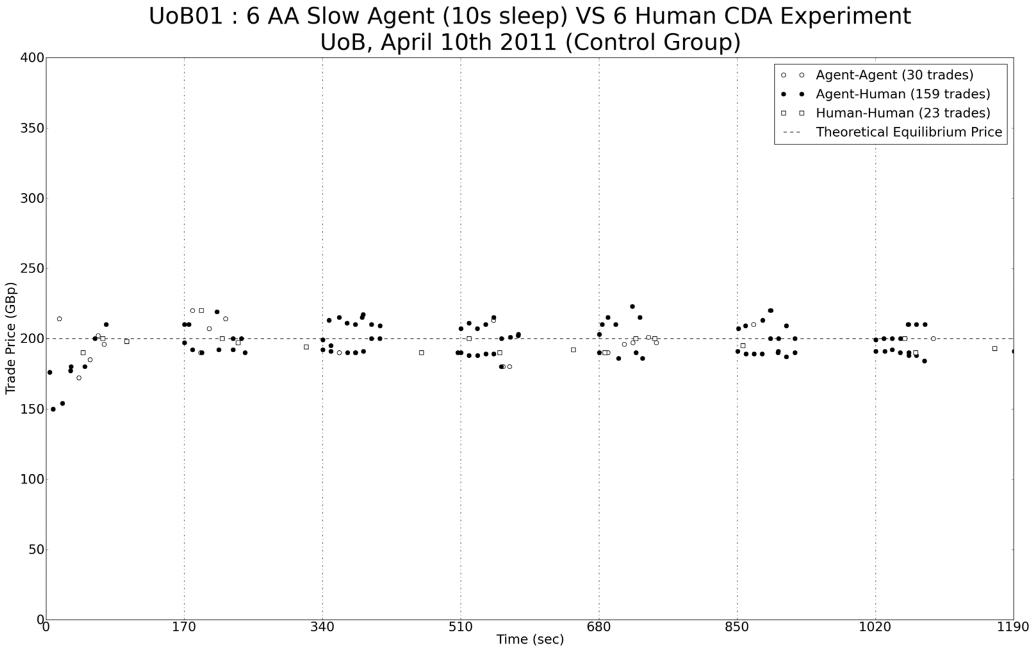


Figure A.5b: Series of transaction prices from participants in experiment UoB1-AA-Slow.

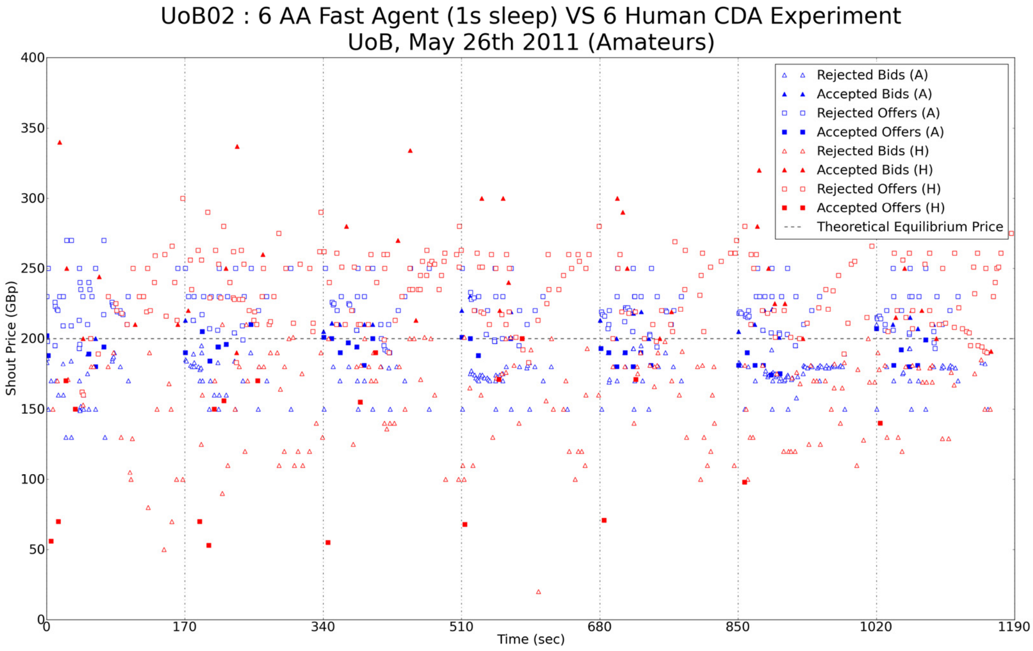


Figure A.6a: Series of quotes from participants in experiment U0B2-AA-Fast.

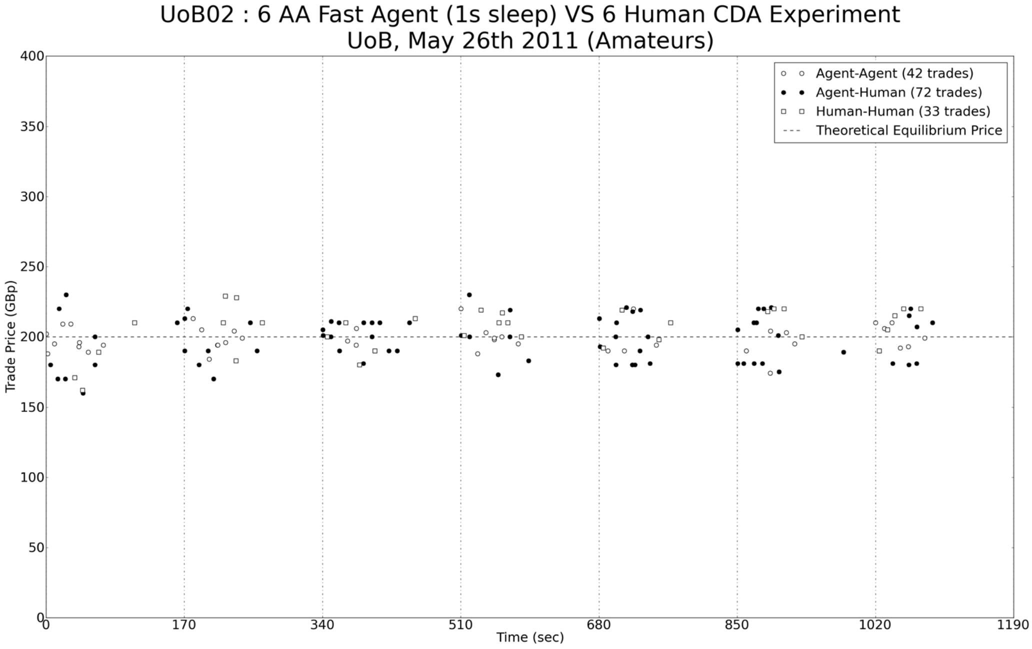


Figure A.6b: Series of transaction prices from participants in experiment U0B2-AA-Fast.

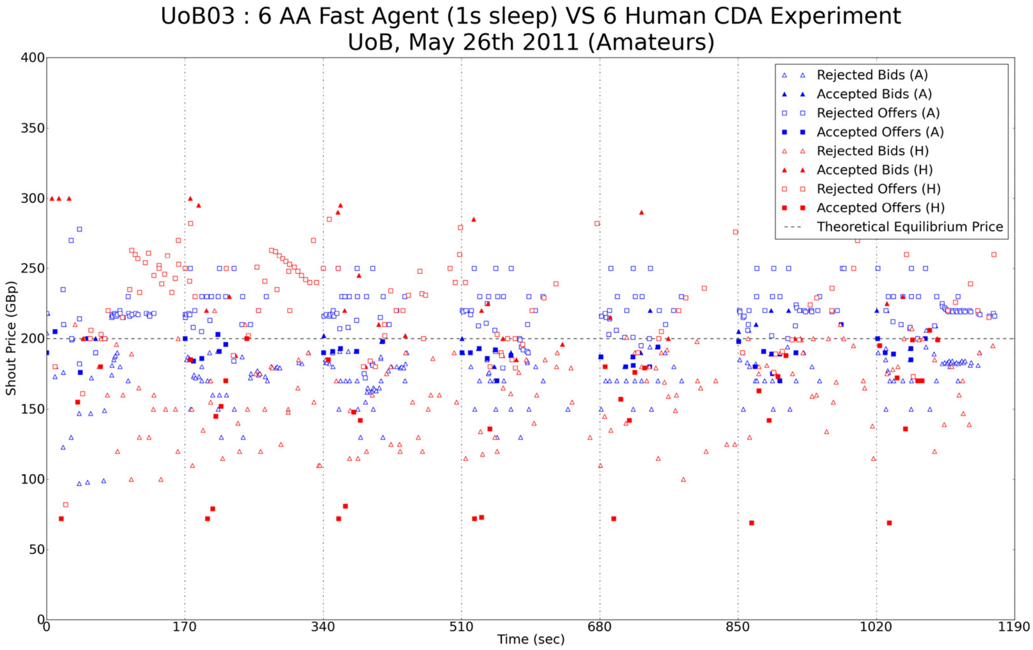


Figure A.7a: Series of quotes from participants in experiment U0B3-AA-Fast.

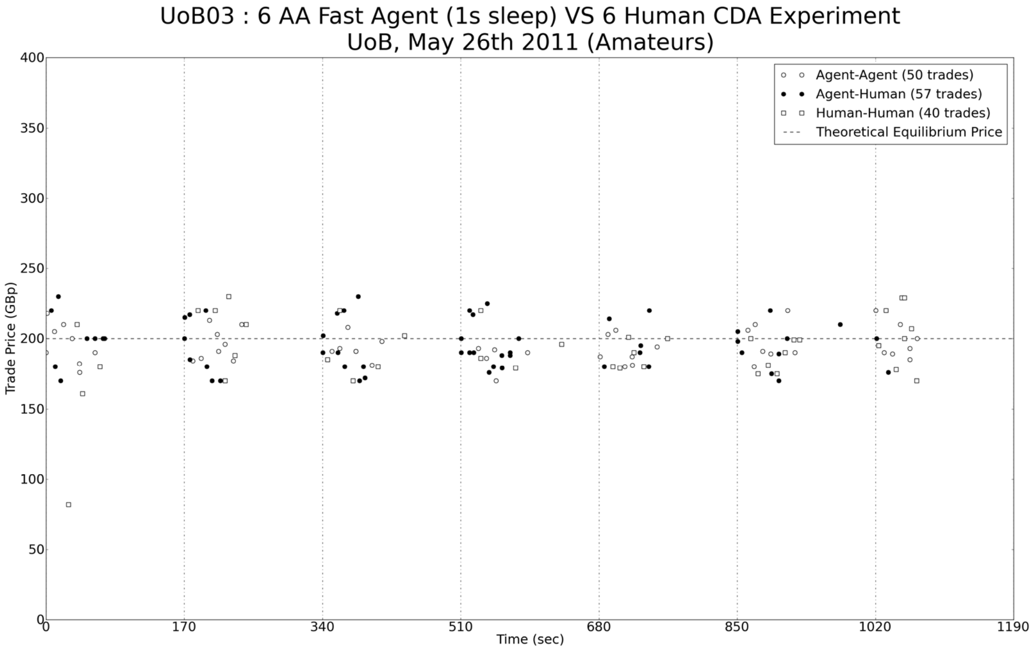


Figure A.7b: Series of transaction prices from participants in experiment U0B3-AA-Fast.

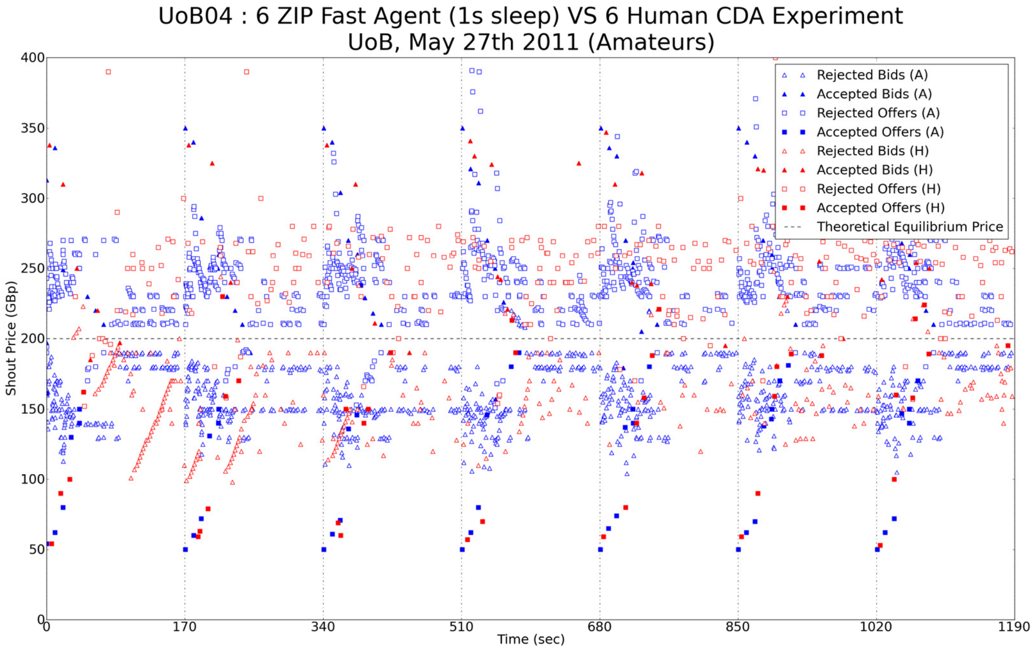


Figure A.8a: Series of quotes from participants in experiment U0B4-ZIP-Fast.

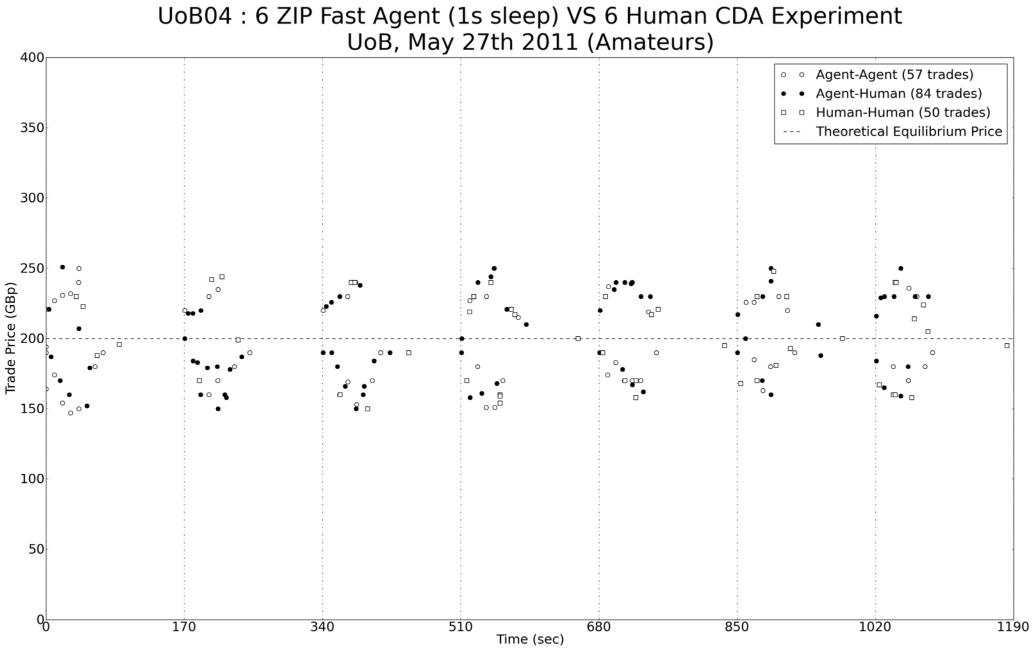


Figure A.8b: Series of transaction prices from participants in experiment U0B4-ZIP-Fast.

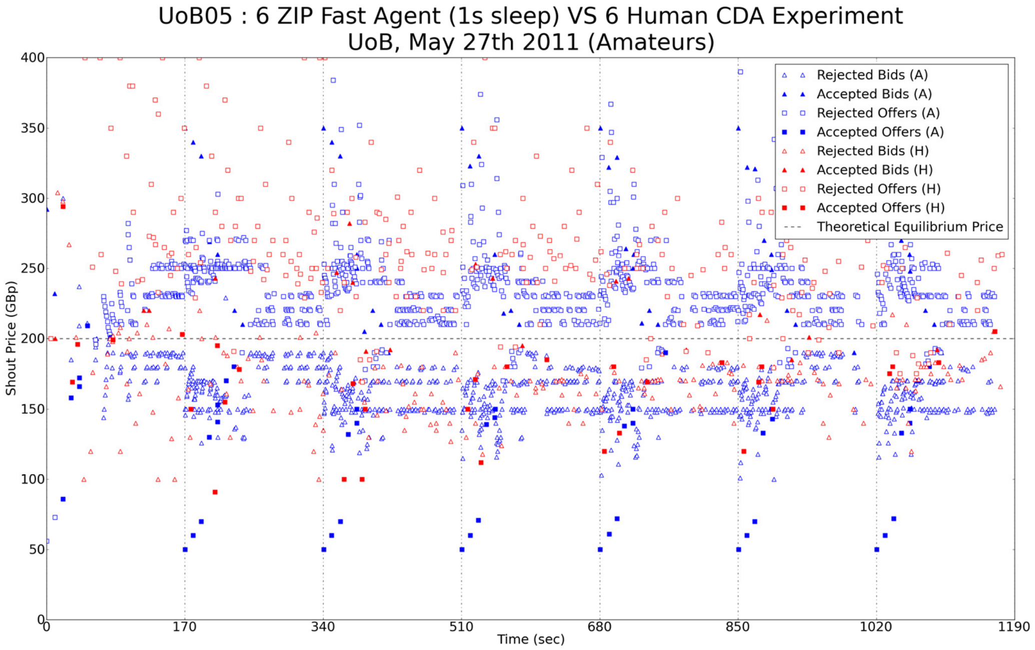


Figure A.9a: Series of quotes from participants in experiment U0B5-ZIP-Fast.

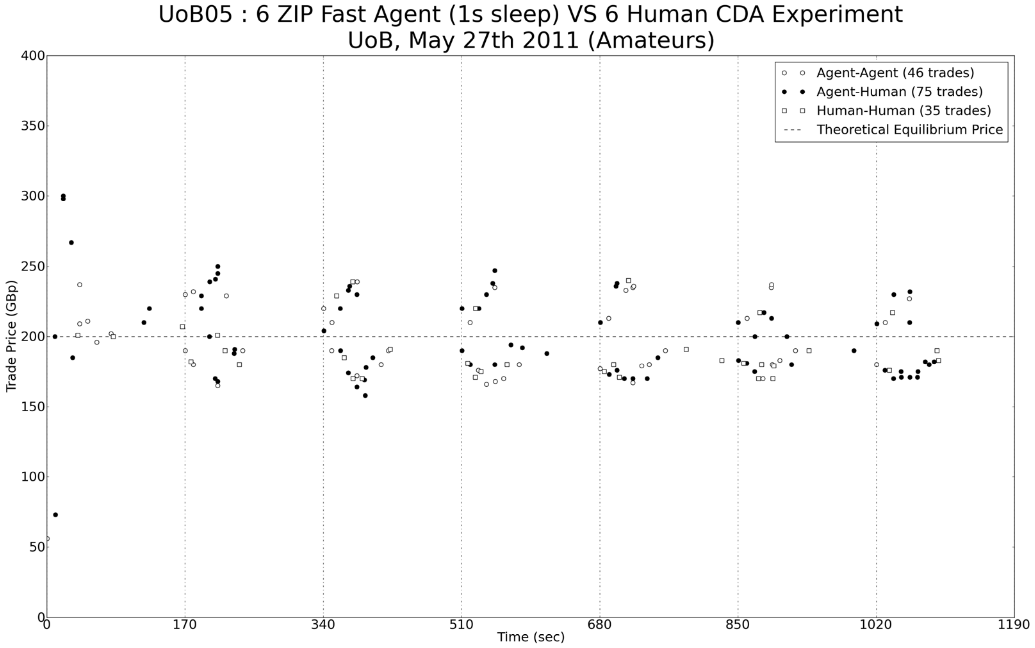


Figure A.9b: Series of transaction prices from participants in experiment U0B5-ZIP-Fast.

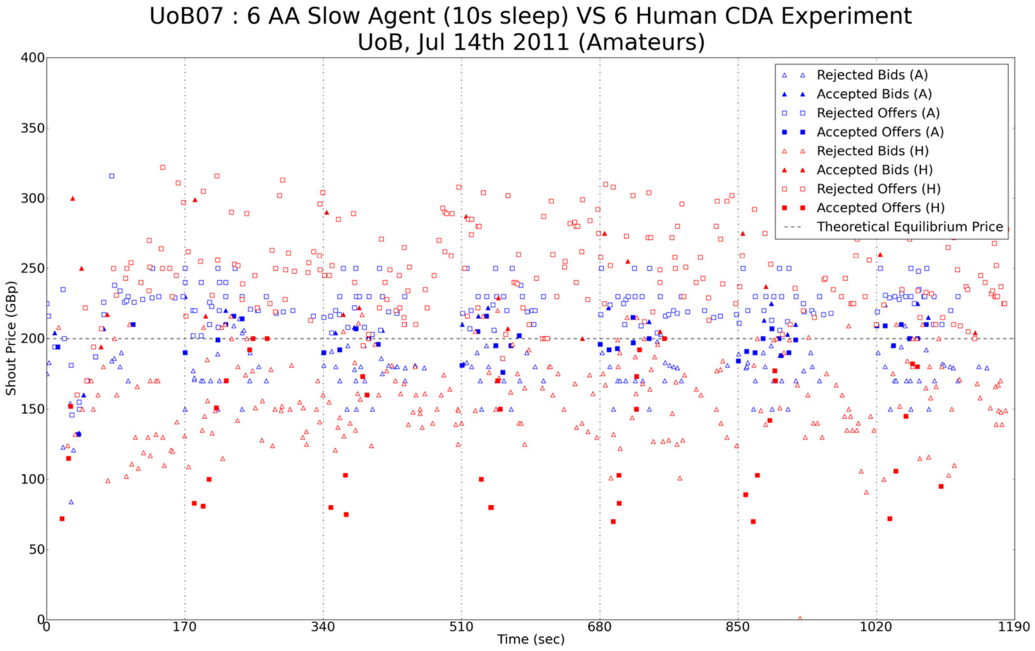


Figure A.10a: Series of quotes from participants in experiment U0B7-AA-Slow.

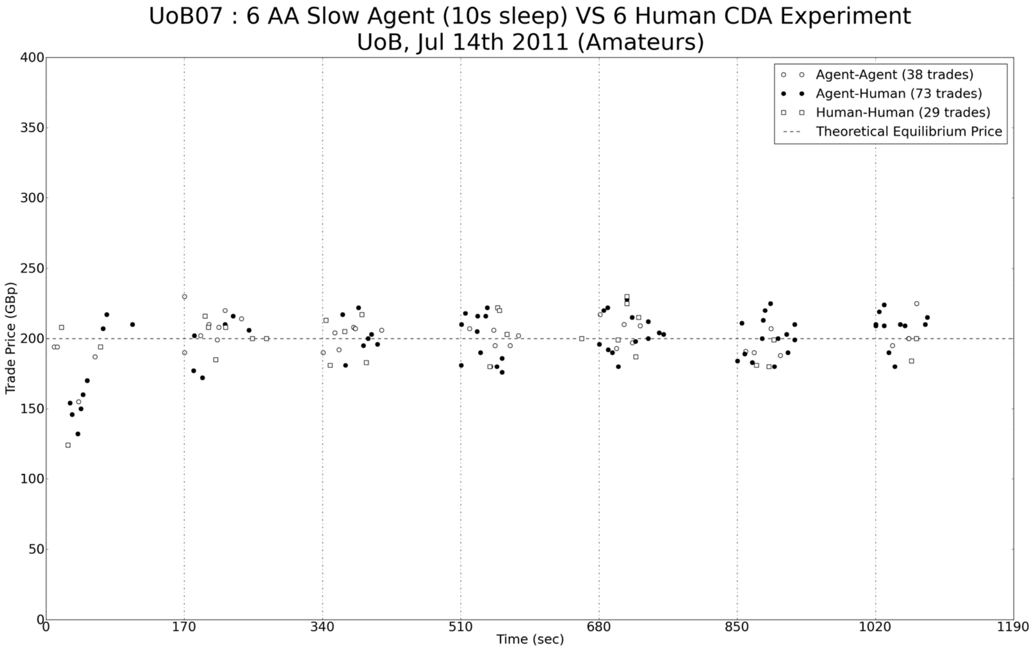


Figure A.10b: Series of transaction prices from participants in experiment U0B7-AA-Slow.

Foresight
1 Victoria Street
London SW1H 0ET
www.foresight.gov.uk
URN 11/1232