LNAI 4452

Maria Fasli
Onn Shehory (Eds.)

Agent-Mediated Electronic Commerce

Automated Negotiation and Strategy Design for Electronic Markets

AAMAS 2006 Workshop, TADA/AMEC 2006
Hakodate, Japan, May 2006
Selected and Revised Papers

Springer

Maria Fasli Onn Shehory (Eds.)

Agent-Mediated Electronic Commerce

Automated Negotiation and Strategy Design for Electronic Markets

AAMAS 2006 Workshop, TADA/AMEC 2006
Hakodate, Japan, May 9, 2006
Selected and Revised Papers

Springer

Series Editors

Jaime G. Carbonell, Carnegie Mellon University, Pittsburgh, PA, USA
Jörg Siekmann, University of Saarland, Saarbrücken, Germany

Volume Editors

Maria Fasli
University of Essex, Department of Computer Science
Wivenhoe Park, Colchester CO4 3SQ, UK
E-mail: mfasli@essex.ac.uk

Onn Shehory
IBM Haifa Research Lab, Tel Aviv Site
Haifa University Campus, Mount Carmel, Haifa 31905, Israel
E-mail: onn@il.ibm.com

Library of Congress Control Number: 2007926025

CR Subject Classification (1998): I.2.11, K.4.4, C.2, H.3.4-5, H.5.3, I.2, J.1

LNCS Sublibrary: SL 7 – Artificial Intelligence

ISSN 0302-9743
ISBN-10 3-540-72501-6 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-72501-5 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12063871 06/3180 5 4 3 2 1 0

Lecture Notes in Artificial Intelligence 4452

Edited by J. G. Carbonell and J. Siekmann

Subseries of Lecture Notes in Computer Science

Preface

The design and analysis of trading agents and electronic trading systems in which they are deployed involve finding solutions to a diverse set of problems, involving individual behaviors, interaction, and collective behavior in the context of trade. A wide variety of trading scenarios and systems, and agent approaches to these, have been studied in recent years. The present volume includes a number of papers that were presented as part of the Joint International Workshop on Trading Agent Design and Analysis and Agent-Mediated Electronic Commerce which was collocated with the Autonomous Agents and Multi-agent Systems (AAMAS) Conference in Hakodate, Japan, in May 2006.

The Joint TADA/AMEC Workshop brought together the two successful and well-established events of the Trading Agent Design and Analysis (TADA) and Agent-Mediated Electronic Commerce (AMEC) Workshops. The TADA series of workshops serves as a forum for presenting work on trading agent design and technologies, theoretical and empirical evaluation of strategies in complex trading scenarios as well as mechanism design. TADA also serves as the main forum for the Trading Agent Competition (TAC) research community. TAC is an annual tournament whose purpose is to stimulate research in trading agents and market mechanisms by providing a platform for agents competing in well-defined market scenarios (http://www.sics.se/tac). The AMEC series of workshops presents interdisciplinary research on both theoretical and practical issues of agent-mediated electronic commerce ranging from the design of electronic marketplaces and efficient protocols to behavioral aspects of agents operating in such environments. The merging of the two workshops was a unique opportunity for researchers working in agents and multi-agent systems, artificial intelligence, operational research, economics and game theory to explore issues pertinent to the development of agent-populated electronic markets. The collection of papers in this volume provides a glimpse into this wide field of research.

The papers presented at the workshop contribute to the theory and practice of agent-based electronic trade and commerce addressing both the agent level and the system level. The papers presented included work directly related to TAC, work related to generic markets and trading scenarios, theoretical and experimental studies, automated negotiation, market mechanism design as well as strategy design.

We hope that this collection of papers will be a useful resource for researchers, practitioners and students working in automated trading and electronic marketplaces.

March 2007

Maria Fasli
Onn Shehory

Organization

Workshop Chairs

Maria Fasli, University of Essex, UK (TADA Chair)
Onn Shehory, IBM - Haifa Research Lab, Israel (AMEC Chair)

Program Committee

Chris Brooks, University of San Francisco, USA
Dave Cliff, University of Southampton/Syritta Algorithmics, UK
John Collins, University of Minnesota, USA
Marc Esteva, IIIA-CSIC, Spain
Boi Faltings, EPFL, Switzerland
Shaheen Fatima, Liverpool University, UK
Amy Greenwald, Brown University, USA
Minhua He, University of Southampton, UK
Sverker Janson, SICS, Sweden
Wolf Ketter, University of Minnesota, USA
Kate Larson, University of Waterloo, Canada
Kevin Leyton-Brown, University of British Columbia, Canada
Tracy Mullen, Penn State University, USA
Julian Padget, University of Bath, UK
David Pardoe, University of Texas at Austin, USA
Sun Park, Yonsei University, Korea
Simon Parsons, Brooklyn College, USA
Jeff Rosenschein, Hebrew University, Israel
Norman Sadeh, Carnegie Mellon University, USA
Alberto Sardinha, ScD - Computer Science - PUC-Rio, Brazil
Carles Sierra, IIIA, Spain
Peter Stone, University of Texas at Austin, USA
Ioannis Vetsikas, University of Southampton, UK
Michael Wellman, University of Michigan, USA
Steven Willmott, UPC, Spain
Dongmo Zhang, University of Western Sydney, Australia

Additional Reviewers

Viswanath Avasarala, Penn State University, USA
Peter Gradwell, University of Bath, UK
Terence Kelly, Hewlett-Packard Labs Palo Alto, USA
Christopher Kiekintveld, University of Michigan, USA
Xudong Luo, University of Southampton, UK
Casey Marks, Brown University, USA
Lars Rasmusson, SICS, Sweden

Lecture Notes in Artificial Intelligence (LNAI)

Vol. 4483: C. Baral, G. Brewka, J. Schlipf (Eds.), Logic Programming and Nonmonotonic Reasoning. IX, 327 pages. 2007.

Vol. 4481: J.T. Yao, P. Lingras, W.-Z. Wu, M. Szczuka, N.J. Cercone, D. Ślęzak (Eds.), Rough Sets and Knowledge Technology. XIV, 576 pages. 2007.

Vol. 4452: M. Fasli, O. Shehory (Eds.), Agent-Mediated Electronic Commerce. VIII, 249 pages. 2007.

Vol. 4438: L. Maicher, A. Sigel, L.M. Garshol (Eds.), Leveraging the Semantics of Topic Maps. X, 257 pages. 2007.

Vol. 4429: R. Lu, J.H. Siekmann, C. Ullrich (Eds.), Cognitive Systems. X, 161 pages. 2007.

Vol. 4426: Z.-H. Zhou, H. Li, Q. Yang (Eds.), Advances in Knowledge Discovery and Data Mining. XXV, 1161 pages. 2007.

Vol. 4411: R.H. Bordini, M. Dastani, J. Dix, A.E.F. Seghrouchni (Eds.), Programming Multi-Agent Systems. XIV, 249 pages. 2007.

Vol. 4410: A. Branco (Ed.), Anaphora: Analysis, Algorithms and Applications. X, 191 pages. 2007.

Vol. 4399: T. Kovacs, X. Llorà, K. Takadama, P.L. Lanzi, W. Stolzmann, S.W. Wilson (Eds.), Learning Classifier Systems. XII, 345 pages. 2007.

Vol. 4390: S.O. Kuznetsov, S. Schmidt (Eds.), Formal Concept Analysis. X, 329 pages. 2007.

Vol. 4389: D. Weyns, H.V.D. Parunak, F. Michel (Eds.), Environments for Multi-Agent Systems III. X, 273 pages. 2007.

Vol. 4384: T. Washio, K. Satoh, H. Takeda, A. Inokuchi (Eds.), New Frontiers in Artificial Intelligence. IX, 401 pages. 2007.

Vol. 4371: K. Inoue, K. Satoh, F. Toni (Eds.), Computational Logic in Multi-Agent Systems. X, 315 pages. 2007.

Vol. 4369: M. Umeda, A. Wolf, O. Bartenstein, U. Geske, D. Seipel, O. Takata (Eds.), Declarative Programming for Knowledge Management. X, 229 pages. 2006.

Vol. 4342: H. de Swart, E. Orłowska, G. Schmidt, M. Roubens (Eds.), Theory and Applications of Relational Structures as Knowledge Instruments II. X, 373 pages. 2006.

Vol. 4335: S.A. Brueckner, S. Hassas, M. Jelasity, D. Yamins (Eds.), Engineering Self-Organising Systems. XII, 212 pages. 2007.

Vol. 4334: B. Beckert, R. Hähnle, P.H. Schmitt (Eds.), Verification of Object-Oriented Software. XXIX, 658 pages. 2007.

Vol. 4333: U. Reimer, D. Karagiannis (Eds.), Practical Aspects of Knowledge Management. XII, 338 pages. 2006.

Vol. 4327: M. Baldoni, U. Endriss (Eds.), Declarative Agent Languages and Technologies IV. VIII, 257 pages. 2006.

Vol. 4314: C. Freksa, M. Kohlhase, K. Schill (Eds.), KI 2006: Advances in Artificial Intelligence. XII, 458 pages. 2007.

Vol. 4304: A. Sattar, B.-H. Kang (Eds.), AI 2006: Advances in Artificial Intelligence. XXVII, 1303 pages. 2006.

Vol. 4303: A. Hoffmann, B.-H. Kang, D. Richards, S. Tsumoto (Eds.), Advances in Knowledge Acquisition and Management. XI, 259 pages. 2006.

Vol. 4293: A. Gelbukh, C.A. Reyes-Garcia (Eds.), MICAI 2006: Advances in Artificial Intelligence. XXVIII, 1232 pages. 2006.

Vol. 4289: M. Ackermann, B. Berendt, M. Grobelnik, A. Hotho, D. Mladenič, G. Semeraro, M. Spiliopoulou, G. Stumme, V. Svátek, M. van Someren (Eds.), Semantics, Web and Mining. X, 197 pages. 2006.

Vol. 4285: Y. Matsumoto, R.W. Sproat, K.-F. Wong, M. Zhang (Eds.), Computer Processing of Oriental Languages. XVII, 544 pages. 2006.

Vol. 4274: Q. Huo, B. Ma, E.-S. Chng, H. Li (Eds.), Chinese Spoken Language Processing. XXIV, 805 pages. 2006.

Vol. 4265: L. Todorovski, N. Lavrač, K.P. Jantke (Eds.), Discovery Science. XIV, 384 pages. 2006.

Vol. 4264: J.L. Balcázar, P.M. Long, F. Stephan (Eds.), Algorithmic Learning Theory. XIII, 393 pages. 2006.

Vol. 4259: S. Greco, Y. Hata, S. Hirano, M. Inuiguchi, S. Miyamoto, H.S. Nguyen, R. Słowiński (Eds.), Rough Sets and Current Trends in Computing. XXII, 951 pages. 2006.

Vol. 4253: B. Gabrys, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part III. XXXII, 1301 pages. 2006.

Vol. 4252: B. Gabrys, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part II. XXXIII, 1335 pages. 2006.

Vol. 4251: B. Gabrys, R.J. Howlett, L.C. Jain (Eds.), Knowledge-Based Intelligent Information and Engineering Systems, Part I. LXVI, 1297 pages. 2006.

Vol. 4248: S. Staab, V. Svátek (Eds.), Managing Knowledge in a World of Networks. XIV, 400 pages. 2006.

Vol. 4246: M. Hermann, A. Voronkov (Eds.), Logic for Programming, Artificial Intelligence, and Reasoning. XIII, 588 pages. 2006.

Table of Contents

Evolutionary Optimization of ZIP60: A Controlled Explosion in Hyperspace

Dave Cliff

Foreign Exchange Complex Risk Group, Deutsche Bank
1 Great Winchester Street, London EC2N 2DB
dave.cliff@bcs.org

Abstract. The "ZIP" adaptive trading algorithm has been demonstrated to outperform human traders in experimental studies of continuous double auction (CDA) markets. The original ZIP algorithm requires the values of eight control parameters to be set correctly. A new extension of the ZIP algorithm, called ZIP60, requires the values of 60 parameters to be set correctly. ZIP60 is shown here to produce significantly better results than the original ZIP (called "ZIP8" hereafter), for negligable additional computational costs. A genetic algorithm (GA) is used to search the 60-dimensional ZIP60 parameter space, and it finds parameter vectors that yield ZIP60 traders with mean scores significantly better than those of ZIP8s. This paper shows that the optimizing evolutionary search works best when the GA itself controls the dimensionality of the search-space, so that the search commences in an 8-d space and thereafter the dimensionality of the search-space is gradually increased by the GA until it is exploring a 60-d space. Furthermore, the results from ZIP60 cast some doubt on prior ZIP8 results concerning the evolution of new 'hybrid' auction mechanisms that appeared to be better than the CDA.

1 Introduction

The Zero-Intelligence Plus (ZIP) adaptive automated trading algorithm [6] has been demonstrated to outperform human traders in experimental studies of continuous double auction (CDA) markets populated by mixtures of human and "robot" traders [15]. To successfully populate a market with ZIP traders, the values of eight real-valued control parameters need to be set correctly. While these eight values can of course be set manually, previous papers have demonstrated that this 8-d parameter-value vector can be automatically optimized using a simple genetic algorithm (GA) search to tailor ZIP traders to particular markets, thereby producing results superior to those from ZIP traders with manually-set parameter values [7, 8]. Furthermore, a simple extension of the GA-ZIP approach (i.e., adding a single additional real-valued numeric parameter, its value set by the GA) allows for automated market-mechanism design, and has been demonstrated as a possible way of automatically discovering novel "hybrid" forms of auction mechanism that appear to be more efficient than the CDA [10, 11, 12]. This paper introduces a more sophisticated version of the ZIP algorithm, which is shown to produce significantly better results. The extended variant is known as "ZIP60", because it requires 60 real-valued control parameters to be set

M. Fasli and O. Shehory (Eds.): TADA/AMEC 2006, LNAI 4452, pp. 1–16, 2007.

correctly, and thus the original algorithm is now re-named as "ZIP8". Manually identifying the correct values for 60 control parameters could be a very laborious task, but it is demonstrated here that an appropriate automatic search or optimization process (such as a GA) can reliably discover good sets of values for the parameters, so long as some care is exercised in controlling a gradual expansion of the dimensionality of the search-space. The GA operating in the 60-dimensional parameter space is shown to produce markets populated by ZIP60 traders with mean scores significantly better than those of ZIP8s. Moreover, the ZIP60 results presented in this paper, while better than ZIP8, show a markedly reduced incidence of cases where the GA also discovers novel hybrid auction mechanisms within which the ZIP traders perform significantly better than when they interact within the fixed CDA mechanism. A plausible conclusion drawn from this is that it indicates that the earlier ZIP8 results (where apparent "improvements" on the CDA were common) were actually consequences of the relative lack of sophistication in the ZIP8 algorithm, rather than consequences of previously-undiscovered weaknesses in the CDA mechanism that the ZIP8 traders were operating within.

In the interests of scientific openness and ease of replicability, the C source-code that was used to generate the ZIP60 results in this paper has been published in a technical report freely available on the web [13].

This paper reports on an ongoing line of research, and there are several open avenues of research that could be pursued to extend or further explore the ideas presented here. In particular, it is important to note that the results in this paper are certainly not intended as an absolute and conclusive demonstration that ZIP60 is superior to all other CDA bidding algorithms, or that the solutions discovered by the GA are optimal in the sense of the GA routinely discovering Nash equilibria in the experimental markets that ZIP60 is studied within here. This paper studies the equilibrating performance of markets that are homogeneously populated with one type of trader-agent, in the style of frequently-cited prior work such as that by Gode & Sunder [20], Cliff [6, 9, 12], Preist & van Tol [30], and Gjerstad & Dickhaut [19]; rather than studying strategic interactions within markets heterogeneously populated by two or more different types of trading algorithms or market mechanisms, such as is exemplified by [38, 39, & 29]. Although the original paper [6] that introduced the ZIP8 algorithm also studied ZIP8's performance *only* in homogeneously populated markets, nevertheless ZIP8 was subsequently used as a benchmark trading algorithm in numerous studies of strategic interactions between heterogeneous mixes of trading algorithms, performed by several independent groups of researchers. The number of such papers in which ZIP8 (or close derivatives of ZIP8) have been used is fairly large, and the list includes: [15, 38, 39, 23, 29, 40, & 1]. Thus, given that so much prior work exploring strategic interactions and heterogeneous populations has been based on ZIP8, it seems reasonable at least to presume that researchers with an interest in studying heterogeneous marketplaces might find ZIP60 a useful new benchmark, even though this current paper reports only on ZIP60 in homogeneous settings. While the study of ZIP60's strategic interactions with other CDA bidding algorithms is certainly an important topic of further research, it is beyond the scope of this current paper.

Furthermore, it is worth noting that in pretty much all of the above-cited papers studying strategic interactions between heterogeneous mixtures of bidding algorithms, the results come from experiments in which the nature of the market supply and demand curves are essentially fixed for the duration of each experiment. That is, studies exploring the effects that significant changes to the supply or demand (or both) curves can have on the trading-agent market's internal dynamics seem pretty rare. Most often, the supply and demand curves in any one trader-agent experiment remain largely the same for the entire duration of that experiment. This seems very curious, given that one commonly-claimed motivation for studying market systems is that mechanisms such as the CDA are interesting because of their ability to quickly and robustly adapt to dynamic and unexpected changes in supply and/or demand; that studies of shock-changes in *human* CDA markets date back as far as Vernon Smith's seminal 1962 paper [36]; and that such changes are known to occur in real-world markets.[1] If CDA markets are interesting because they exhibit attractive adaptation to dynamic changes in supply and demand, why is there this affection in the trading-agent literature for studying CDA systems where such changes are largely absent? In contrast, the results reported in this paper all come from experiments in which the marketplaces periodically undergo sudden "shock" changes to the supply and/or demand curves, and where the ZIP60 traders are optimized on the basis of their ability to rapidly and stably adapt to the new market conditions prevailing after each shock-change.

The rest of this paper is structured as follows. Section 2 gives an overview of ZIP traders and of the experimental methods used, including a description of the continuously variable space of auction types. This description is largely identical to the account given in previous papers (e.g., [10, 12]), albeit extended to describe how the new experiments whose results are presented here differ from the previous work. The new ZIP60 results are then presented, analyzed, and discussed in Section 3.

2 Methods

2.1 The Original Eight-Parameter ZIP

The original eight-parameter ZIP trading algorithm was first described fully in a lengthy report [6], which included source-code (in ANSI C) of an example implementation. For the purposes of this paper, a high-level description of the algorithm and its eight key parameters is sufficient. Illustrative C source-code for ZIP60 has been published in [13]. As will be seen in Section 3, there are in fact a family of ZIP algorithms between ZIP8 and ZIP60, and so hereinafter the acronym "ZIP" with no numeric suffix is intended to mean "all ZIPn for $8 \leq n \leq 60$ and beyond".

ZIP traders deal in arbitrary abstract commodities. Each ZIP trader i is given a private (i.e., secret) limit-price, λ_i, which for a seller is the price below which it must not sell and for a buyer is the price above which it must not buy. If a ZIP trader completes a transaction at its λ_i price then it generates zero utility ("profit" for the sellers or "saving" for the buyers). For this reason, each ZIP trader i maintains a time-varying utility margin $\mu_i(t)$ and generates quote-prices $p_i(t)$ at time t using $p_i(t)=\lambda_i(1+\mu_i(t))$ for sellers and $p_i(t)=\lambda_i(1-\mu_i(t))$ for buyers. The "aim" of traders is to

[1] E.g., in high-frequency foreign-exchange price time series, "gap" step-changes in price are not unusual.

maximize their utility over all trades, where utility is the difference between the accepted quote-price and the trader's λ_i value. Trader i is given an initial value $\mu_i(0)$ (i.e., $\mu_i(t)$ for $t=0$) which is subsequently adapted over time using a simple machine learning technique known as *the Widrow-Hoff rule* which is also used in back-propagation neural networks and in learning classifier systems. This rule has a "learning rate" parameter β_i that governs the speed of convergence between trader i's quoted price $p_i(t)$ and the trader's idealized "target" price $\tau_i(t)$. When calculating $\tau_i(t)$, ZIP traders introduce a small random *absolute* perturbation generated from[2] $\boldsymbol{U}[0,c_a]$ (this perturbation is positive when increasing $\tau_i(t)$, negative when decreasing) and also a small random *relative* perturbation generated from $\boldsymbol{U}[1\text{-}c_r,1]$ when decreasing $\tau_i(t)$, or from $\boldsymbol{U}[1,1+c_r]$ when increasing $\tau_i(t)$, where c_a and c_r are global system constants. To smooth over noise in the learning system, there is an additional "momentum" parameter γ_i for each trader (such momentum terms are also common in back-propagation neural networks).

So, adaptation in each ZIP trader i has the following parameters: initial margin $\mu_i(0)$; learning rate β_i; and momentum term γ_i. In an entire market populated by ZIP traders, values for these three parameters are randomly assigned to each trader via $\mu_i(0)=f_a(\mu_{min}, \mu_\Delta)$, $\beta_i=f_a(\beta_{min}, \beta_\Delta)$, and $\gamma_i=f_a(\gamma_{min}, \gamma_\Delta)$; for $f_a(\alpha, \kappa)=\boldsymbol{U}[\alpha, \alpha+\kappa]$. Hence, to initialize an entire ZIP-trader market, it is necessary to specify values for the six market-initialization parameters μ_{min}, μ_Δ, β_{min}, β_Δ, γ_{min}, and γ_Δ; and for the two system constants c_a and c_r. Thus any set of initialization parameters for a ZIP-trader market exists within an eight-dimensional real space – hence "ZIP8".

Vectors in this 8-space can be considered as "genotypes" in a genetic algorithm (GA), and from an initial population of randomly generated genotypes it is possible to allow a GA to find new genotype vectors that best satisfy an appropriate evaluation function. This is exactly the process that was first introduced in [7, 8]. For the purposes of this paper, we will consider the GA optimizer as a "black box" and leave it largely un-discussed: full details accompany the source-code in [13].

In addition to using the GA to optimize the control parameters for the trader-agents, one more real-valued numeric parameter was introduced in [10–12] to give the GA automated control over the auction mechanism. This market-mechanism parameter is called Q_s and it governs the *exogenously imposed* probability that the next quote in the marketplace will be taken from a seller, so Q_s=0.0 is a pure one-sided auction where only buyers can quote (and hence is similar to an English auction); Q_s=1.0 is pure one-sided with only sellers quoting (as in a Dutch Flower auction); and Q_s=0.5 makes quotes from buyers or sellers equi-probable (as in a CDA). The surprising result reported in [10–12] is that "hybrid" auction mechanisms (such as Q_s=0.25) were found by the GA to give the best evaluation scores when the value of Q_s was evolved alongside the values of the eight ZIP control parameters. Experiments where the value of Q_s was under control of the GA are referred to here as "EM" (for "evolving mechanism") experiments, and experiments where the value of

[2] Here $v=\boldsymbol{U}[x,y]$ denotes a random real value v generated from a uniform distribution over the range $[x,y]$.

Q_s was fixed, typically at the CDA value of 0.5, are referred to as "FM" experiments (for "fixed mechanism").

The fitness of genotypes was evaluated here using the methods described previously [7, 8, 10–12]: one *trial* of a particular genome was performed by initializing a ZIP-trader market from the genome, and then allowing the ZIP traders to operate within the market for a fixed number of trading periods (often colloquially referred to as "days"), with allocations of stock and currency being replenished between each trading period. During each trading period, Smith's [36] α measure (root mean square deviation of transaction prices from the market's theoretical competitive equilibrium price) was monitored, and a weighted average of α was calculated across the days in the trial, using a method described in more detail in the next section. As the outcome of any one such trial is influenced by stochasticity in the system, the final evaluation score for an individual was calculated as the arithmetic mean of 100 such trials. Note that as *minimal* deviation of transaction prices from the theoretical equilibrium price is desirable, lower scores are better: we aim here to *minimize* the evaluation scores. That is, individuals with lower scores have greater reproductive fitness.

2.2 Previous ZIP8 Results

In [12], results from 32 sets of experiments were published, where each experiment involved sequences built from one or more of four specific market supply and demand schedules. These four schedules are referred to as markets M1, M2, M3, and M4, and are illustrated in [12, 13]. In all four schedules there are 11 buyers and 11 sellers, each empowered to buy/sell one unit of commodity. Market M1 is taken from Smith's seminal 1962 paper [36] on his early experimental economics work, and the remaining three markets are variations on M1. In M2 the slope of the demand curve has been greatly reduced while the slope of the supply curve has been increased only slightly; and in M4 the slope of the supply curve has been greatly reduced while the slope of the demand curve has been increased only slightly. In M3 the slopes of both the supply and demand curves are only slightly steeper than the slopes in M1, yet these minor differences between the supply and demand curves in M1 and M3 can still lead to significant differences in the final best evolved solutions.

The experiments reported in this paper use a method first explored in ZIP8 experiments, involving "shock changes" being inflicted on the market by swapping from one schedule to another partway through the evaluation process. Here, two shocks occurred during each evaluation process (i.e., switching between three schedules). For instance, in one experiment referred to here as M121, the evaluation involved six trading periods ("days") with supply and demand determined by M1, then a sudden change to M2, then six periods/days later a reversion to M1 for a final six periods. The other sets of experiments are similarly named M212, M123, M321, and so on. Each of the three market schedules was used for six "days", so the two-shock trials last for 18 days. As in the previous GA-ZIP work, the evaluation function was a weighted average of Smith's [36] "α" measure of root mean square deviation of transaction prices from the underlying theoretical equilibrium price at the start of the experiment, measured across the six periods for each schedule used: in each trading period p the value α_p was calculated, and the evaluation score was computed as

$(1/\Sigma w_p).\Sigma(\alpha_p.w_p)$ for $p=1...18$ with weights $w_1=1.75$, $w_2=1.5$, $w_3=1.25$, $w_{3<p<7}=1.0$, $w_{p>6}=w_{p-6}$, and $w_{p>12}=w_{p-12}$.

The process used to compare the EM and FM cases is as follows. In any one experiment, here involving a population of 30 genotypes over 500 generations, in each generation the elite (best-scoring) individual is of most interest, and so the time-series of the elite fitness score for the population is monitored across the 500 generations. These results are non-deterministic: different runs of the GA (with different seed values for its random number generator) will yield different elite trajectories. Examining the results from 50 repetitions of an experiment (varying only the random seed between repetitions) often gives multimodal results, and in all experiments we are interested only in the best elite mode (i.e. the mode with lowest scores), which can be summarized by the mean and standard deviation (s.d.) of the scores within that mode at each generation: these two values will be referred to here as the *best elite-mode* fitness mean and s.d.. For comparison purposes, in the ZIP8 work reported in [12], similar trajectories of best elite-mode fitness values were recorded from 50 repetitions of the each experiment in fixed-mechanism (FM) conditions, where the value of Q_s was *not* evolved but instead was fixed at the CDA value of Q_s=0.5.

The results from 18 dual-shock (triple-schedule) experiments were presented in four separate data-tables in [12], grouped by the nature of the shocks (i.e., the "treatment regime"). Table 3 showed results from experiments where only the demand curve undergoes a major change on each shock (i.e.: M121, M212, M232, M323, M123, and M321). Table 4 showed results from experiments where only the supply curve undergoes a major change on each shock (i.e.: M141, M414, M434, M343, M143, and M341). In Table 5, one of the two shocks involves a major change only to the demand curve while the other shock involves a major change only to the supply curve (i.e.: M432, M234, M412, and M214); and in Table 6 each shock involved a major change to both the supply curve and the demand curve (i.e.: M242 and M424). In this paper, all 18 dual-shock results are shown together in a single graph, but the results appear in table order, as was just listed.

Analysis of the ZIP8 results showed that the GA never failed to discover EM genotypes that were at least as good (i.e. had elite evaluation scores at least as low) as the corresponding FM genotypes, and in several cases the EM result was significantly better (lower) than the FM result, at the 1% confidence level, using appropriate non-parametric significance tests such as the Wilcoxon-Mann-Whitney (see, e.g., [35]), or latterly the Robust Rank Order test [16].

The histogram in Figure 1 shows the results for GA-optimized ZIP8 in FM and EM conditions. Fig.1 also shows the results from various styles of ZIP60 EM experiments, discussed further in Section 3 of this paper. The ZIP8 statistics in Fig.1 are the results of conducting a more rigorous and careful analysis (discussed in [13]) of the data than was originally summarized and tabulated in [12]. The final evaluation score recorded as the outcome of any one experiment is now taken as an average of the final few elite scores (over generations 490 to 500) to smooth over noise in the evaluation process; and the summary statistics for each type of experiment are here always calculated from the top 10% (i.e., the upper decile) of the 50 repetitions of each type of experiment, regardless of how many repetitions converged on solutions with final elite scores in the best elite mode. So, the data in Fig.1 show the mean and s.d. of the final outcome elite scores from the best (lowest-scoring) five experiments in each study.

2.3 Related Work

These previous GA-ZIP results have subsequently been replicated, adapted, and extended in a number of independent studies. Robinson [32] explored the use of evolved market-mechanisms in the context of market-based control (e.g. [4]) of scarce resources in utility-scale corporate data centers. Walia [41] explored the use of the same evolving-mechanism techniques but with markets populated by Gode & Sunder's [20] ZI trader-agents rather than ZIP traders, again finding evidence that non-standard hybrid mechanisms were discovered as good/best solutions by the GA; and Byde [2] demonstrated that the same techniques could lead to the evolution of hybrid sealed-bid auction mechanisms, regardless of the type of trader operating in the market. Shipp [34] investigated how the nature of the evolved solutions changed as the number of "market shocks" used in the evaluation process increased; and Wichett [43] explored a system in which multiple reproductively separate "gene-pools" of ZIP traders competed, co-adapted, and co-evolved along with the market mechanism. Other recent uses of ZIP include modifying it for bargaining in sealed-bid auctions [1]; using ZIP traders to study speculative trading in business-to-business exchanges [25]; and using ZIP traders to explore issues of reputation and information quality in a variety of market configurations [24].

The results in [10] were the first demonstration that radically new market mechanisms for artificial traders may be designed by automatic means. But, at much the same time as they were being generated, Steve Phelps and his colleagues were independently working on a conceptually very similar (but algorithmically rather different) theme of using artificial evolution to develop and study new auction-market mechanisms [29]. In addition to the contemporaneous work of Phelps *et al.*, a number of other authors have more recently reported on the results of using artificial evolution and other forms of automated search, learning, or optimization for exploring spaces of possible trader-agent strategies, and possible new auction mechanisms, generally with positive results [39, 18, 26, 28, 21, 27, 31, & 42]. Of course, the paper introducing ZIP [6] was not the first-ever study of artificial trading agents in double-auction markets; notable prior work includes [44], [17], and [33]. Also, [19] was developed independently at much the same time. For additional discussion of earlier work, see [6].

3 ZIP60

3.1 From 8 to 60 in Five Paragraphs

The results from using a GA to fine-tune the ZIP8 trader were sufficiently encouraging that they provoke the question of whether new variants of ZIP can be developed to take advantage of the fact that we can now (generally, at least) rely on automated optimizers like the GA to set appropriate values for the numeric parameters affecting the traders. If we commit to using an optimizer to set the parameter values, we don't need to keep the number of parameters small enough for them all to be manageable or comprehensible by humans. That's the rationale for ZIP60.

To this end, observe that in ZIP8 the genome specifies the same vector of eight real values $\{\mu_{min}, \mu_{\Delta}, \beta_{min}, \beta_{\Delta}, \gamma_{min}, \gamma_{\Delta}, c_a, c_r\}$ whether the trader is a buyer or a seller. But in some situations it's plausible that the market dynamics might be better if the parameter-values used by the buyers were different to those used by the sellers, so we could in principle have a GA-ZIP system dealing with these two cases (i.e. where *Case 1* is that the trader is a buyer; *Case 2* is that the trader is a seller) and hence optimizing sixteen real parameters (i.e., "ZIP16"), with the first vector of eight values being used to initialize the buyers and the second being used to initialize the sellers.

Next, note that in some situations a ZIP trader (whether it is a buyer or a seller) has to increase its margin, and in others it has to decrease its margin, and that it may be useful to have different parameter-values depending on which of the four cases we are in, i.e. whether the trader is a buyer raising its margin, a buyer lowering its margin, a seller raising, or a seller lowering. That's 4 cases, each with 8 values, and so "ZIP32". But we can then additionally note that, in the original specification of the ZIP algorithm, both for buyers and for sellers, there are actually *three* different cases or circumstances in which the trader alters its margin (see [6] pp.42-43 for the details of and rationale for this design). For example, a seller's margin is *raised* if *one* condition holds true (i.e., if the last quote was accepted and the seller's current price is less than the price of the current quote); but a seller's margin is *lowered* if either of *two* other possible conditions are true (i.e.: if the last quote was an accepted bid and the seller is active and the seller's price is greater than the price of the last quote; *or* if the last quote was an offer that was accepted and the seller is active and its price is greater than the price of the last quote). So we could have the genome specify *three* corresponding parameter-value vectors for the buyers and also *three* such vectors for the sellers, i.e. a total of six different vectors for six different cases, which at eight values per vector gives us "ZIP48".

And in a final flourish of parameter-count inflation, let's abandon the use of a mere pair of system-wide global constants c_a and c_r and in place initialize each trader i with its own corresponding "personal" values $c_{a,i}$ and $c_{r,i,}$ generated at initialization from the uniform distributions $\boldsymbol{U}[c_{a:min}, c_{a:min}+c_{a:\Delta}]$ & $\boldsymbol{U}[c_{r:min}, c_{r:min}+c_{r:\Delta}]$. This addition of extra parameters still allows solutions involving the old system-wide constant c_a and c_r values to be "discovered" by the GA – that will happen if better evaluation scores are associated with (near-)zero values of $c_{a:\Delta}$ and $c_{r:\Delta}$. So, the parameter-value vectors for each case needs now to specify not only the six previous system parameters (μ_{min}, μ_{Δ}, β_{min}, β_{Δ}, γ_{min}, and γ_{Δ}) but also the values for the four newly-introduced system parameters $c_{a:min}$, $c_{a:\Delta}$, $c_{r:min}$, and $c_{r:\Delta}$ – i.e., ten values per vector. For six cases, each with ten values per vector, we get to sixty values: "ZIP60".

It is worth noting that this final increase from eight to ten parameter-values per case could also be applied to any of the other ZIP*n* versions mentioned in the preceding paragraphs. That is, by the expansion of the specification of c_a and c_r, ZIP8 becomes ZIP10; ZIP16 becomes ZIP20; and ZIP32 becomes ZIP40.

We need also to introduce some terminology that will ease the analysis and discussion that come later. While a ZIP8 trader has one genetically-specified value for each parameter (so, for example, it has only one β_{min} value), a ZIP60 genome specifies six related parameter values – one for each case – which we will refer to by adding