Agent Based Information Aggregation Markets

Efthimios Bothos
National Technical University of Athens
Iroon Polytechniou 9 Zografou
Athens Greece 157 80
+30-210-772-3895
mpthim@mail.ntua.gr

Dimitris Apostolou
University of Piraeus
Karaoli & Dimitriou St. 80 Piraeus
Greece 185 34
+30-210-414-2314
dapost@unipi.gr

Gregoris Mentzas
National Technical University of Athens
Iroon Polytechniou 9 Zografou
Athens Greece 157 80
+30-210-772-3895
gmentzas@mail.ntua.gr

ABSTRACT
Information Aggregation Markets (IAMs) constitute a mechanism whose purpose is to collect and aggregate information. They are commonly known as 'prediction markets' because they are often used to predict future events. In order for IAMs to efficiently aggregate information, they should attract participants with diversity of opinion, independence of thought and decentralization of knowledge. When participants are typically well-informed, IAM prices will aggregate information into market prices. IAMs can be considered as a large-scale, open, distributed system used by many human participants whose information needs to be aggregated. In this paper we propose a different approach for information aggregation using IAMs. We use autonomous, interacting agents instead of human participants and we show that agent based IAMs can act as an information aggregation mechanism able to perform predictions without human involvement.

Categories and Subject Descriptors
I.2.11 [Computing Methodologies]: Artificial Intelligence-Intelligent Agents

General Terms
Performance, Design, Economics, Experimentation.

Keywords
Agent Cooperation, Collective intelligence, Collective decision making, Judgment aggregation and belief merging.

1. INTRODUCTION
The global build-up of the World Wide Web has made it possible that anyone with a computer and Internet access may explore, join, and contribute to any Web community at any time. This new freedom is often attributed to the "Web 2.0 era" of services and applications that let users easily share content, opinions and resources [1]. Various mechanisms have been proposed to aggregate and combine user generated content in such a way as to make it useful. The most prominent ones are collaborative filtering, collaborative tagging and folksonomies [2].

Cite as: Agent Based Information Aggregation Markets, Bothos E., Apostolou D. and Mentzas G., *Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, van der Hoek, Kaminka, Lespérance, Luck and Sen (eds.), May, 10–14, 2010, Toronto, Canada, pp. 449-454 .

Web 2.0 tools and methods allow users to express their opinion about different matters. This is typically done by using various rating and voting schemes or simply by using natural language. Different users may favor different methods while different Web 2.0 sites may employ different evaluation methods. But useful information can reside in e.g. both ratings and comments. It is apparent that a unifying mechanism for seamlessly aggregating user opinions in a single and actionable indictor is required.

A unifying mechanism for aggregating user opinions should incorporate at least two important characteristics. First it should not generate excessive overhead for users and second it must efficiently aggregate information without compromising the validity of the results over simplicity. In this paper we propose a new approach for aggregating the bits of information residing on the Internet, based on artificial agents and Information Aggregation Markets (IAMs). Agents act as proxies for humans, collect the available information and participate in markets where they trade based on the acquired information. We describe a novel system that implements our approach which we use to run a set of simulations to aggregate information relevant to the best movie Oscars competition. Our target is to identify the winner using publicly available information.

The rest of the paper is structured as follows. In section 2 we present the related work. Section 3 outlines our approach and analyzes the design options we followed when implementing our system. In section 4 we describe a set of simulations, we run and present the results. Some conclusive remarks and ideas for future work are given in the last section.

2. RELATED WORK
2.1 Collective intelligence using IAMs
IAMs are considered an example of collective intelligence because of their ability to aggregate the preferences of a potentially large cohort in the price of contracts representing different outcomes of future events. The contracts are a superset of the financial derivative called "future". Participants trade on futures, the final price of which depends on the outcome of a future event; upon market end, the price of these contracts incorporates the available information with respect to that event. IAMs are commonly known as 'prediction markets' because they are often used to predict future events.

Interpreting prices to reveal information is not a new notion: Friedrich Hayek suggested in an essay published in 1948 that prices in naturally occurring, free markets make important contributions to information in economies [3]. This suggestion was theoretically and empirically grounded by Fama [4] when he established the efficient markets hypothesis that states an efficient market continuously reflects all available information about future events in the price of stocks. In this work, market equilibrium, i.e. the state in which all information is reflected in prices, is described using the theory of expected returns [5]. This means that the expected return conditional on the available information is fully utilized to determine the equilibrium prices as follows:

$$E(\tilde{p}_{j,t+1} \mid \Phi_t) = [1 + E(\tilde{r}_{j,t+1} \mid \Phi_t)]p_{jt} \qquad [1]$$

where Φt is a general symbol for the information available at time t, E is the expected value operator p , $\tilde{p}$ denote the price and the expected price respectively and denotes the expected percentage return over the period t and t+1.

Trading on futures was first introduced in 1973 by Richard Sandor [6]. Since then, futures prices have proven to be a valuable forecasting tool. The successful results of this application have impelled academics and practitioners to design and implement several successful trials of IAMs within companies, primarily for forecasting purposes. Examples range from identifying product launch dates to predicting future sales of products ([7], [8]).

In order for IAMs efficiently to aggregate information, they should attract a sound cohort. Surowiecki [9] has provided a qualitative analysis of participant characteristics necessary for the market to be trustworthy: diversity of opinion, independence of thought and decentralization of knowledge. Wolfers and Zitchevitz [10] established a theoretical model and provided an account of sufficient conditions under which IAM prices aggregate private information held amongst participants. They concluded that, when participants are typically well-informed, IAM prices will aggregate information into useful information. This is in accordance with the Condorcet Jury Theorem [11], which suggests that a group of individuals has a higher probability of making the correct decision as the size of the cohort increases, provided sufficient individuals have a better than even probability of making the correct decision, i.e. they are well informed.

2.2 Agent based trading

Artificial agents have been widely used in economics, mostly for simulations. The relevant field of applications is referred as agent-based computational economics (ACE). The advantages of using computational agents to abstract and model human behavior are many-fold. Agents do not require incentives and space thus they scale up fairly easy. To this direction various approaches have been followed by researchers for agents' design [12]. The use of genetic algorithms and genetic programming has shown positive results in experiments that try to mirror the laboratories with human subjects (see e.g. [13], [14]). Recent developments involve agents with varying cognitive ability and agents with personality [12]. Furthermore agent based simulation has been a popular technique in modeling and analyzing electricity markets in recent years [15]. Such applications try to address the distinct characteristics (such as physical transmission infrastructure, various types of intelligent market participants and their interactions, decision making and adaptation) of electricity markets.

Besides the use of agent based trading for simulations, autonomous artificial agents have been developed to participate in electronic market places. In such applications computer programs are faced with the challenge to effectively increase their revenue without human intervention [16]. Agents act as mediators or proxies for humans and their performance is measured by the profit or savings they materialize. Another application for agent based trading includes portfolio management artificial agents that help investors select an optimal portfolio that satisfies their objective, or, in other words, maximize the investment returns under given constraints [17]. Last but not least, other interdisciplinary approaches propose agent based trading for solving large scale problems such as road traffic management [18].

3. AGENT BASED INFORMATION AGGREGATION

3.1 Our approach

We consider a cloud of information sources scattered throughout the Internet, containing user opinions in various forms, e.g. comments, ratings etc. Our approach for information aggregation using IAMs uses autonomous, interacting agents instead of human participants. Each agent is comprised of two distinct functions. One that is responsible for opinion extraction and interpretation and another that performs trading decisions and engages in transactions.

The functions of our agents were inspired from the Belief-Desire-Intentions (BDI) paradigm. According to the BDI framework an agent is characterized by its beliefs, goals (desires), and intentions; it will intend to do what it believes and will achieve its goals given its beliefs about the world [19]. Our agents build a view (belief) of the world by extracting and interpreting opinions, have the goal (desire) of maximizing their financial resources in a virtual market through rational trading and engage (intend to do) in transactions based on the available information. This follows the processes of the real world IAMs where human traders receive information signals and based on their personal assessment of that information buy or sell contracts in order to maximize their portfolios. The transactions are executed in an IAM environment.

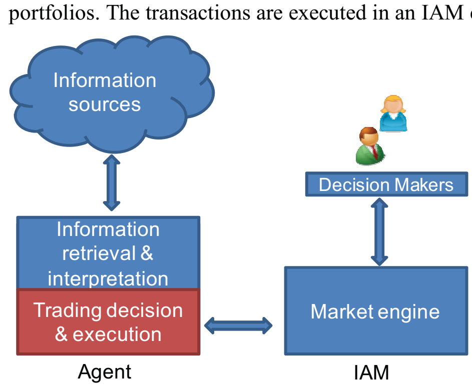

Figure 1. The conceptual architecture of our approach.

A market mechanism is responsible for matching buyers and sellers. Finally the output of the trading process is available to decision makers in the form of a simple indicator, the market prices. The conceptual architecture of our approach is summarized in Figure 1.

From a technical perspective opinion extraction is based on software that parses digital content while opinion interpretation varies from calculation of ratings to the extraction of sentiments from text-based comments. The trading mechanism is selected among the prominent ones currently used in IAM implementations. The design of the artificial agents and the trading mechanism which is the core of the IAM are described in the following sections.

3.2 Agents design

3.2.1 Opinion extraction and interpretation

Although various opinion sources could be employed, in this paper we describe the use of Web users generated comments and ratings. Typically, a parser is required in order to retrieve these comments and ratings. Although many websites offer feeds in the form of Really Simple Syndication (RSS), custom made parsers have to be created for others that do not provide such functionalities. Considering that the proper parser is in place, opinions need to be interpreted and transformed in a way that trading decisions can be made by our agents. Incoming opinions can be labeled as being Positive, Negative or Neutral. For ratings we use the following function to interpret information:

$$f(r) = \begin{cases} Positive, if\ r > 0{,}5 \\ Neutral, if\ r = 0{,}5 \\ Negative, if\ r < 0{,}5 \end{cases} \qquad [2]$$

where r denotes the provided normalized rating $\in [0,1]$.

For user generated comments we applied what is commonly known as sentiment analysis techniques [20]. In general, textual information can be mainly categorized into facts and opinions. Facts are objective expressions about entities, events and their properties. Opinions are usually subjective expressions that describe sentiments and feelings about an event or an entity and their properties. A lot of academic research has been focused on factual information mining and retrieval, like information retrieval, Web search, text clustering and classification etc. Recently, a growing interest in mining people's opinions has been observed. Opinions are important not only for individuals, but also for companies and organizations that are interested in analyzing consumer sentiment in order to understand how their products and services are perceived.

Sentiment analysis is treated as a text-classification problem. We make use of the extensively studied technique of classifying an opinionated document as expressing a positive, negative or a neutral sentiment, commonly known as sentiment classification or document-level sentiment classification. For more information please refer to [20]. After evaluating the available systems, we concluded to Jane16[1], an open-source component that performs sentiment analysis. Jane16 provides a pure statistical engine and no lexical analysis is incorporated in the system. It is based on a database that has been built by automatically "reading" a vast amount of online review sites (news, articles etc.) that have been rated either as positive (5/5 stars) or negative (1/5 stars). The phrases, taken from these reviews, were then counted and a number indicating their absolute frequency in positive and respectively in negative reviews was assigned as weight in every sentence.

In order to check the effectiveness of the system, we used the publically available labeled data from Pang and Lee, used in [21]. This dataset contains 1000 positive samples and 1000 negative samples taken from Rotten Tomatoes[2] pages. They were automatically labeled as positive if they were marked as "fresh" and negative if marked with "rotten". This component was incorporated into our artificial agents' opinion extraction engine.

3.2.2 Trading & belief update

Agents' uncertain beliefs are modeled with a beta distribution function which can be used to derive probability distributions based on observations of past events. The mathematical analysis leading to this result can be found in probability theory, e.g. Casella & Berger 1990 [22], and we will only present the results here.

The beta-family of distributions is a continuous family of distribution functions indexed by the two parameters α and β. The beta distribution can be expressed using the gamma function Γ as:

$$f(p \,\|\, a,b) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1} \qquad [3]$$

Where $0 \le p \le 1$, $\alpha > 0$, $\beta > 0$

The probability expectation value and the variance of the beta distribution are given by:

$$E(p) = \frac{a}{a+\beta} \quad [4],\quad Var(p) = \frac{\alpha\beta}{((\alpha+\beta)^2(\alpha+\beta+1))} \quad [5]$$

We use the Bayes theorem to find the posterior probability of a future event being realized conditional on the received information signal, or more formally:

$$p(\theta \mid y) = \frac{p(y \mid \theta)p(\theta)}{p(y)} \qquad [6]$$

Where θ is the future event of interest which can have two possible values in our case (either 0 if it is not realized or 1 if it is realized), y denotes the observed data, $p(y|\theta)$ the probability of data y for a value of θ (likelihood), $p(\theta)$ the prior, initial distribution for θ, $p(\theta|y)$ the posterior distribution for θ, given the data and $p(y)$ the total probability for the given data y.

If we consider a future event with two possible outcomes $\{x, \bar{x}\}$ like in our case and r is the observed number of outcome x and s the observed number of outcome $\bar{x}$, using beta distribution and Bayes theorem the probability distribution of observing outcome x in the future can be expressed as a function of past observations by setting:

$\alpha = r+1$, $\beta = s+1$ where $r, s \ge 0$.

The agents are initialized with $\alpha = \beta = 1$, values that give the beta distribution the form of a uniform distribution. If a positive information signal is received, we consider a success and we update the distribution by increasing the parameter α with a value of one. Instead if a negative signal is received, we update the distribution and we increase the parameter β. Each agent's beta

[1] http://sourceforge.net/projects/jane16api/

[2] http://www.rottentomatoes.com/

distribution will converge to a probability based on the information the agent receives. So agents build a stronger belief as their beta distribution converges to a specific probability value.

The aggregation of the agents' distributions is performed through trading in the market. We consider that agents have infinite resources (money and number of future contracts). Before trading each agent i generates his fair price (valuation) by selecting a random number V_i from his beta distribution and submits orders for 1 contract according to that fair price (this means that agents try to move prices closer to their valuation).

Orders are divided into two sub-classes. An offer to sell is called an ask, and an offer to buy is called a bid. If A is the set of all agents, an order can be represented as a tuple of the form:

$$\rho = (\rho_c \in \{\text{bid, ask}, \emptyset\}, \rho_a = 1, \rho_p \in R, \rho_q \in N, \rho_t \in N) \qquad [7]$$

where ρ_c is the sub-class of the offer, ρ_a is the agent placing the offer, ρ_p is the price the agent is willing to buy or sell at, ρ_q is the quantity of contracts to be sold or bought (always 1), and ρ_t is the time at which the order is being submitted.

Before an agent generates an order, he inspects the current market price MP_i. The following rules determine whether an ask or a bid will be placed:

- If $V_i > MP_i$ the agent bids at the price of V_i
- If $V_i < MP_i$ the agent asks at the price of V_i
- If $V_i = MP_i$ the agent does not engage in a transaction

3.3 Information Aggregation Market design

3.3.1 Prominent IAM mechanisms

In IAMs, a market mechanism is employed to allow trading on the virtual contracts. The main purpose of such a mechanism is to serve traders' buy or sell orders. Although several mechanisms can be used such as call auctions or pari-mutuel mechanisms (e.g. the dynamic pari-mutuel [23]) the majority of IAMs make use of the following three [24]: 1) Continuous Double Auction (CDA), 2) Market Maker with Market Scoring Rules (MM) and 3) a hybrid of the two above mentioned often referred to as Continuous Double Auction with Market Maker (CDAwMM).

The Continuous Double Auction mechanism is a process in which market participants freely enter limit orders (bid and asks) and accept already submitted orders by other participants. The clearing operation is performed continuously as new orders arrive. In a CDA buyers enter bids by stating their identity, unit price and quantity. The same buyer can subsequently remove her/his order or modify it by raising or lowering the price and/or the quantity. Similar rules apply for sellers. They can enter an ask order by stating quantity and price which they can modify according to their valuation and beliefs in future time. A transaction occurs when bids and asks match or cross. If the bid or ask price of a submitted order doesn't match the ask (or bid) price of an existing order, it will be added to a queue often referred as the '*book of orders*'. The result of a transaction is an exchange of assets (future contracts in the case of IAMs) and money between the two trading parties. The major advantage of double auctions compared to other market institutions is that order placements occur simultaneously and in a decentralized manner.

The alternative to maintaining an order book is to have an automated market maker. Market makers in stock markets ensure that the market is liquid by always having a published price at which they are willing to buy or sell. They make profit by maintaining a spread between the prices at which they buy and sell; the quoted price is the price at which you can buy from them. They set the price according to their beliefs about the current demand for the asset, and change the price as their opinion (or their exposure) changes.

Automated market makers set the price according to a rule that tells them how much to raise (lower) the price when a buy (sell) order is processed. A simple rule would be to buy or sell a single contract at the current price, and change the price by a constant amount. If the market maker bases its prices on a rule that produces consistent prices whether buying or selling, it can sell an unlimited number of shares while limiting its losses.

In [25] a logarithmic market scoring rule is introduced where a market is organized with claims of the form "Pays $ 1 if the state is i". The cost and pricing functions for this logarithmic market scoring rule mechanism are computed in [26]. These cost and pricing functions allow the mechanism to be implemented as an automated market-maker where orders will be accepted or rejected based on these functions. Let's assume that there are S states over which contingent claims are traded. If the vector q ∈ RS represents the number of orders on each state that have already been accepted by the market organizer, the total cost of all the orders already accepted is calculated via the cost function C(q). A trader submits an order characterized by the vector r ∈ RS where ri reflects the number of claims over state i that the trader desires. The market organizer will charge the trader C(q + ri) − C(q) for his order. The pricing function is simply the derivative of the cost function with respect to one of the states. It represents the instantaneous price for an order over one state. The cost and pricing functions for the logarithmic market scoring rule are computed as follows:

$$C = b * Ln\left(\sum_j e^{q_i/b}\right) \qquad [8], \qquad p_i(q) = \frac{e^{q_i/b}}{\sum_j e^{q_j/b}} \qquad [9]$$

where b is a parameter that must be chosen by the market organizer. It represents the risk that the organizer is willing to accept. The greater the value of b, the more orders the organizer is likely to accept and more liquidity or depth is added to the market, meaning that traders can buy more shares at or near the current price without causing massive price swings [26].

A weakness of algorithmic market makers is that they don't adjust well as the volume of trading changes. As mentioned above, the algorithm is parameterized by a constant that controls how quickly prices move in response to trading. When the market has many participants, prices should move more slowly in response to trading than when there are only a few traders. If the wrong constant is chosen, then it will be too hard to move the price in a thin market, or in a thick market it might move back and forth too often. On the other hand order books work well in thick markets (in which there are plenty of offers for people to trade with), but less well with fewer traders. Combining the two should produce markets that work well in both cases.

The idea of using a CDA with market maker has been proposed in [25]. In order to run a market with both a market maker and book orders, it has to be ensured that the market maker gets priority and that the book orders are satisfied in the correct order as the market maker's price changes. Buy and sell orders are tracked in queues, each sorted by the offered price. The market maker can freely trade and adjust its price as long as the price remains between the highest offer to buy and the lowest offer to sell. If the market receives a new offer, the market maker trades first, until it reaches the price of the best offer, then the book orders are used. If the new offer hasn't reached its limit, the process iterates. Using a market maker increases the number of trades that are possible, especially when the markets are thin or many traders are reluctant to enter book orders. As a result, adding book orders makes it easier for the market to adjust to increasing volume.

In order to select a proper market mechanism we analyzed their behavior on a set of measures, namely the forecasting error, price stability and percentage of accepted transactions using two different types of agents. For a thorough analysis please refer to [27].

Given the facts that 1) the agents in our approach do not have constant beliefs 2) we require a high percentage of accepted orders as our artificial agents should always be given the opportunity to modify market prices, we selected the CDAwMM mechanism in our simulations.

4. SIMULATION

4.1 Purpose and Description

The purpose of our simulation was to test our approach by aggregating information relevant to events that have already occurred thus we know the outcome and we can validate the performance of our approach. We chose the event of Oscar winners with data drawn from the well-known movies database IMDB (www.imdb.com). We run three simulations for the best picture award of the years 2006, 2007, 2008 and 2009. For that purpose we created a parser that is able to retrieve user generated comments and ratings placed on the IMBD web-site. All the comments and ratings placed up to the Oscar awards date were retrieved since users could also provide new information even after that date. Two agents were created per movie nominee, one that traded based on the information retrieved for ratings and another based on the information retrieved for comments. Agents processed the daily information and placed their orders as explained in section 2.2.

We executed our simulation with two assumptions in mind. First, we consider that the best movie gains wide acceptance from the public and second the public opinion is expressed through the comments and ratings. Given these assumptions we consider that the aggregated information should point out the Oscar awards winner.

In order to evaluate our approach, we compared our results first with those of Hollywood Stock Exchange (HSX), an IAM used for events relevant to entertainment, including Oscar awards with over 1.4 million registered users[3] and second with the polls offered by IMDB where users vote for the winning movie. Furthermore we compare all approaches with the final decision of the Oscars' Academy which consists the reference point. We have to note that HSX provides only the closing prices of the contracts that have expired. We noticed that, except for the winning contract, all others had a very low price which can be the effect of last minute trading. Without the full historical data from HSX available we decided to avoid more rigorous comparisons thus we compare on the basis of predicting the correct outcome of the Oscars' contest.

[3] http://www.hsx.com/about/bloomberg_20060410.pdf

4.2 Results

Tables 1 and 2 summarize the results of our simulation and subsequent comparisons. HSX appears to hold the greatest prediction accuracy as shown in Table 1. With respect to our approach, we can observe that when we don't use sentiments the results match those of IMDB polls. An inconsistency emerges when we add information relevant to the sentiments of the user generated comments. This can be attributed to noise created from the relevant dataset or the sentiment analysis tool we selected.

In Table 2 we performed comparisons between the ranking from IMDB polls and the one generated by our approach. A positive correlation was found for all years.

5. CONCLUSIONS AND FURTHER WORK

In this paper we proposed and presented a new approach for aggregating dispersed bits of information created by users in the Web 2.0 internet. We run an inaugural simulation in order to test the validity of our approach. Our purpose was to aggregate user generated movie ratings and comments with a target to predict Oscar awards winners. The results are encouraging regarding the validity of our approach.

Table 1. Predictive accuracy of HSX, IMDB polls and our approach

Year	HSX to predict the winning movie	IMDB polls to predict the winning movie	Our approach to predict the winning movie when ratings are considered	Our approach to predict the winning movie when ratings and sentiments are considered
2006	Yes	No	No	No
2007	Yes	Yes	Yes	No
2008	Yes	Yes	Yes	Yes
2009	Yes	Yes	Yes	Yes

Table 2. Spearman's rank correlation coefficient for IMDB polls and our approach

Year	Spearman's rank correlation coefficient for IMDB polls and our approach when ratings are considered	Spearman's rank correlation coefficient for IMDB polls and our approach when ratings and sentiment are considered
2006	1	0.77
2007	0.57	0.17
2008	0.9	0.87
2009	0.7	0.85

Nonetheless our research is not without limitations. First, we observed deterioration of the predictive accuracy when we infused sentiments from user comments. Further simulations and the use of alternative sentiment analysis approaches may lead to better information extraction.

Another important issue is that our artificial agents followed a fairly simple strategy for assessing information and for trading. Their level of intelligence can be improved using more sophisticated techniques. For example approaches where agents observe the market prices and learn from their fluctuations can be employed. Moreover experimentation with a larger set of information sources is required. In our simulation we used information from IMDB only, which expresses the views of the users that participate in that website. Similar information exists in other places as well and should be provided as input to our approach. Information can take several representations out of which we selected ratings and sentiments extracted from user comments. Other forms of information that can be aggregated include quotes from IAMs such as HSX, rankings provided by experts, and even human traders who can interact with our artificial agents in a unified system.

6. REFERENCES

[1] Lin, K.-J. 2007. "Building Web 2.0". IEEE Computer. 40, 101-102.

[2] Gartner, 2006. "Web 2.0 Basics: Principles, Practices and Platforms for the Enterprise". Symposium ITExpo.

[3] Hayek, F. A. 1948. Individualism and Economic Order. University of Chicago Press.

[4] Fama E. 1965. "Behavior of stock market prices". Journal of Business, 38, 34–105.

[5] Sharpe W. F. 1964. "Capital Asset Prices: A Theory of Market Equilibrium under Conditions of Risk". The Journal of Finance, 19, 3, 425-442.

[6] Sandor R. L. 1973. "Innovation by an Exchange: A Case Study of the Development of the Plywood Futures Contract". Journal of Law and Economics, 16, 1, 119-136.

[7] Skiera B. and Spann M. 2004. "Opportunities of Virtual Stock Markets to Support New Product Development", Verlag Gabler Wiesbaden, 227-242.

[8] Chen K. Y. and Plott C. 2002. "Information Aggregation Mechanisms: Concept, Design and Implementation for a Sales Forecasting Problem". California In-stitute of Technology Social Science Working Paper 1131.

[9] Surowiecki J. 2004. The Wisdom of Crowds. Doubleday.

[10] Wolfers J. and Zitzewitz E. 2004. "Prediction markets". J. of Economic Perspective, 18, 107-126.

[11] Berend D. and Paroush J. 1998. "When is Condorcet's Jury Theorem valid?" Social Choice and Welfare, 15, 4, 481-488.

[12] Chen S.-H. 2008. Software-agent designs in economics: An interdisciplinary framework. IEEE Computational Intelligence Magazine 3(4), 18-22.

[13] Arifovic J. 1994. "Genetic algorithm learning and the cobweb model". Journal of Economic Dynamics and Control, 18, 1, 3–28.

[14] Chen S.-H. and Tai C.-C. 2003. "Trading restrictions, price dynamics, and allocative efficiency in double auction markets: Analysis based on agent-based modeling and simulations," Advances in Complex Systems, 6, 3, 283–302.

[15] Zhou, Z., Chan, W., & Chow, J. 2007. Agent-based simulation of electricity markets: a survey of tools. Artificial Intelligence Review , 28 (4), 305-342.

[16] Panos Toulis , Dionisis Kehagias , Pericles A. Mitkas, Mertacor: a successful autonomous trading agent, 2006. AAMAS, May 08-12, Hakodate, Japan.

[17] Chiu-Che Tseng, 2003. "Comparing Artificial Intelligence Systems for Stock Portfolio Selection," Computing in Economics and Finance 2003 236, Society for Computational Economics.

[18] Vasirani, M. and Ossowski, S. 2009. A market-inspired approach to reservation-based urban road traffic management. AAMAS - Volume 1 (Budapest, Hungary, May 10 - 15, 2009), 617-624.

[19] Norling, E., Sonenberg, L.: Creating Interactive Characters with BDI Agents. In: Proceedings of the Australian Workshop on Interactive Entertainment IE(2004).

[20] Pang, B. and Lee, L. 2008. "Opinion Mining and Sentiment Analysis", Foundations and Trends in Information Retrieval 2(1-2), 1-135.

[21] Pang, B. and Lee, L. 2005. "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales", Proceedings of the ACL.

[22] George Casella and Roger L. Berger 1990. Statistical Inference. Duxbury Press.

[23] Pennock, D. 2004. A Dynamic pari-mutuel market for hedging, wagering, and information aggregation. In: Proceedings of Electronic Commerce.

[24] Wolfers, J. and Zitzewitz E. 2004. "Prediction Markets", Journal of Economic Perspectives, 18, 2, 107-126.

[25] Hanson, R. 2002. Logarithmic Market Scoring Rules for Modular Combinatorial Information Aggregation, available at: http://hanson.gmu.edu/mktscore.pdf.

[26] Pennock D.. 2006. Implementing Hanson's Market Maker , http://blog.oddhead.com/2006/10/30/implementing-hansons-market-maker/.

[27] Bothos E., Apostolou D., Mentzas G., "A Comparative Study of Market Mechanisms for Information Aggregation with Agent Based Simulation", 2009, online, available at: http://www.imu.iccs.gr/