






The Agents' Selection Methods for a Consensus-Based Investment Strategy in a Multi-agent Financial Decisions Support System

Marcin Hernes¹ , Adrianna Kozierekiewicz² ,
and Marcin Pietranik² 

¹ Wrocław University of Economics, ul. Komandorska 118/120,
53-345 Wrocław, Poland
marcin.hernes@ue.wroc.pl

² Faculty of Computer Science and Management,
Wrocław University of Science and Technology, Wybrzeże Wyspiańskiego 27,
50-370 Wrocław, Poland
{adrianna.kozierekiewicz, marcin.pietranik}@pwr.edu.pl

Abstract. Modern financial decision support systems are often based on a multi-agent approach to make advice for investors. However, having a large set of different decisions, collected from agents participating in the process, may entail problems related to data integration and its computational complexity. In this paper, we present some algorithms for selecting agents from a set of all available participants to be included in the eventual decision-making process. All algorithms have been experimentally verified using the a-Trader - a prototype of a multi-agent financial decision support systems on a Forex market.

Keywords: Consensus · Multi-agent system · Forex · Investment strategies · Agents' performance

1 Introduction

Contemporary financial decisions are often made in accordance with distributed information systems, including multi-agent systems. More importantly, these decisions must be taken in near real time, especially in case of High-Frequency Trading. The agents running in such a supporting system generate buy/sell decisions based on different methods and factors, such as technical, fundamental or behavioral analysis. It may highly increase their number within systems, frequently reaching hundreds of participating agents. Using their decisions, investment strategies are built. For example, in the Forex market, these strategies provide signals for open/close short/long positions.

Collecting decisions from distributed and independent agents entails two problems. The first concerns issues related to integrating partial outcomes into a unified, final decision that can be further utilized. This problem is widely known in the literature, and a plethora of different approaches, with their advantages and disadvantages, exist.

The second problem is directly related to the efficiency of both collecting participating agents' decisions and their further processing. There are many approaches to this issue, spanning from, architectural requirements, through parallelization of computations. The question appears: how much and which intelligent agents should participate in making the decision process to give the most profitable investment strategy?

In our earlier research, we showed that it is not required to have a large number of agents to assert a high quality of their decision integration. The still open problem is, the selection of agents to be included in the decision process.

In this paper, we want to investigate how reducing this number of participating agents, based on different selection methods, impacts the quality of strategies created by agents in a financial application (Fintech). Therefore, the main aim of this paper is to implement and evaluate different methods for agents' selection, to build a consensus-based strategy in a-Trader- a prototype of a multi-agent financial decision support systems on a Forex market.

The paper is structured as follows. In the next section, an overview of publications covering similar topics is given. Section 3 contains some basic information about the a-Trader system. The core of the article can be found in Sect. 4, where we present a set of algorithms for selecting agents in a multi-agent decision support system from a pool of available ones. The experimental verification of these algorithms is given in Sect. 5. Section 6 concludes that paper and sheds some light on our upcoming research plans.

2 Related Works

Incorporating agent-based solutions within contemporary financial applications is not an uncommon approach. In [8] authors give some characteristics and requirements, that such tools must fulfill. A starting point is a detailed description of the downfalls of a centralized approach. This included (among others) an enormous amount of knowledge that needed to be processed, dealing with bottlenecks and complex tasks related to data filtering. Distributing these tasks among a set of independent agents may be a good solution for given problems.

A practical example of a system fulfilling overviewed expectations is described in [9], where an application of a multi-agent based model in the stock exchange has been described. This system provides an environment capable of both performing efficient distributed computations and, by using a fuzzy expert system, intelligent decision making.

In [1] authors propose to use agents in a Forex market. The paper contains a description of an implementation of a set of independent currency trading agents, based on classification and regression models. Six trading agents were developed, each responsible for trading different currency pairs in the Forex market.

Authors of [4] describe the design of TACTic, which is a multi-agent based system built upon multi-behavioral techniques. It was developed to fulfill several goals: (i) provide bidding decisions in uncertain environments, (ii) to perform predictions

about the outcomes of different auctions, (iii) to modify agents' bidding strategies in response to trending market conditions. The architecture of the other client and server-side agents are described in [3]. Conducted experiments show the correctness of proposed ideas.

An important issue related to all approaches is their efficiency. In modern applications (especially, while processing high traffic financial data) computational complexity of knowledge processing cannot be a bottleneck. Therefore, there are many attempts to overcome related problems. One of them is using a multi-level approach to partial results integration [7, 8], which has been proved useful in terms of quality of obtained outcomes and its time-efficiency. More issues related to this problem can also be found in [2].

3 A-Trader System

The a-Trader system has been broadly described in [1]. Due to the page limit, in this paper, we provide only general information about this system. The a-Trader consists of about 1500 agents divided into the following groups: Notification Agent, Market Communication Agents, Basic Agents, Intelligent Agents, and Supervisor Agents. The strategies are built based on agents generating buy or sell a financial instrument. Thus, in our system a single agent's decision is defined as follows [6]:

Definition 1. *Decision D about finite set of financial instruments $E = \{e_1, e_2, \dots, e_N\}$ is defined as a set:*

$$D = EW^+, EW^\pm, EW^-, Z, SP, DT \quad (1)$$

where: $EW^+ = \{\langle e_o, pe_o \rangle, \langle e_q, pe_q \rangle, \dots, \langle e_p, pe_p \rangle\}$. A couple $\langle e_x, pe_x \rangle$, where: $e_x \in E$ and $pe_x \in [0, 1]$ denote a financial instrument and this instrument's participation in the set EW^+ , $x \in \{o, q, \dots, p\}$. The financial instrument $e_x \in EW^+$ is denoted by e_x^+ . $EW^\pm = \{\langle e_r, pe_r \rangle, \langle e_s, pe_s \rangle, \dots, \langle e_t, pe_t \rangle\}$. A couple $\langle e_x, pe_x \rangle$, where: $e_x \in E$ and $pe_x \in [0, 1]$ and denote a financial instrument and this instrument's participation in EW^\pm the set, $x \in \{r, s, \dots, t\}$. The financial instrument $e_x \in EW^\pm$ will be denoted by e_x^\pm . $EW^- = \{\langle e_u, pe_u \rangle, \langle e_v, pe_v \rangle, \dots, \langle e_w, pe_w \rangle\}$. The couple $\langle e_x, pe_x \rangle$, where: $e_x \in E$ and $pe_x \in [0, 1]$ and, denote a financial instrument and this instrument's participation in a set EW^- , $x \in \{u, v, \dots, w\}$. The financial instrument $e_x \in EW^-$ will be denoted by e_x^- . $Z \in [0, 1]$ - predicted rate of return, $SP \in [0, 1]$ - degree of certainty of rate Z , DT - date of a decision.

A-Trader system is a multi-agent system which gives some suggestion about buying or selling some financial instruments. Different agents in the system determine decisions based on their knowledge. The set of decisions provided by each Base and Intelligent Agents are understood as the knowledge profile. The Supervisor Agent provide the final advice to the trader. In our previous work [5] we have proposed an

algorithm for generating a final decision of a multi-agent system. For this purpose, a Consensus Theory has been applied.

4 Approaches to Agents' Selection

The effectiveness of the A-Trader system has been proved in our previous works [5, 6]. It has been pointed out that the strategies implemented in Intelligent Agents have the biggest influence on the profitability of the final decision. Thus, to compare the agents' performance, the following evaluation function was used [5]:

$$y = (a_1x_1 + a_2x_2 + a_3x_3 + a_4x_4 + a_5x_5 + a_6x_6 + a_7x_7 + a_8x_8 + a_9x_9) \tag{2}$$

where: x_1 - rate of return, x_2 - gross profit, x_3 - 1-gross loss, x_4 - the number of profitable transactions, x_5 - the number of profitable transactions in a row, x_6 - 1-the number of unprofitable transactions in a row, x_7 - Sharpe ratio, x_8 - 1-the average coefficient of volatility, x_9 - the average rate of return per transaction, counted as the quotient of the rate of return and the number of transactions.

All values of x_i (where $i \in (1, 2, \dots, 9)$) are normalized. It was adopted in the test that coefficients a_i to $a_9 = 1/9$. The final investment strategy is established based on partial decisions of Intelligent Agents. However, the question which appears: how much and which Intelligent Agents should participate in making the decision process to give the most profitable investment strategy? In this paper, we propose some heuristic algorithms for choosing agents, which can assert the best final decision. The result of the proposed algorithms is a generated set of chosen agents BA , based on input data: the performance of M agents: $y^{(1)}, y^{(2)}, \dots, y^{(M)}$, M is the number of agents, $k \leq M$ - the number of classes, $x_1^{(1)}, \dots, x_9^{(1)}, x_1^{(2)}, \dots, x_9^{(2)}, \dots, x_1^{(M)}, \dots, x_9^{(M)}$.

Algorithm 1. Simple agents' choosing	Algorithm 2. The ranking based agents' choosing
<p>Begin</p> <p>1: Order of the performance values $y^{(1)}, y^{(2)}, \dots, y^{(M)}$ in descending order.</p> <p>2: Put into the set BA them agents with the highest value of performance.</p> <p>End</p> <p>Return BA</p>	<p>Begin</p> <p>1: Choose the biggest value $y^{(max)}$ from $y^{(1)}, y^{(2)}, \dots, y^{(M)}$.</p> <p>2: Calculate the mean value of the performance: $\mu = \frac{\sum_{i=1}^M y^{(i)}}{M}$</p> <p>3: Calculate the standard deviation of the performance: $\sigma = \sqrt{\frac{\sum_{i=1}^M (y^{(i)} - \mu)^2}{M}}$</p> <p>4: For each agent $i \in \{1, \dots, M\}$ if $y^{(i)} \geq y^{(max)} - \sigma$ put i-th agent into set BA.</p> <p>End Return BA</p>

Algorithm 3. The k-means based agents' choosing	Algorithm 4. Modification of k-means based agents' choosing
<p>Begin</p> <p>1: $S^{(1)} = \emptyset, S^{(2)} = \emptyset, \dots, S^{(k)} = \emptyset$</p> <p>2: Randomly choose k initial centroid: $c^{(1)}, c^{(2)}, \dots, c^{(k)}$ of class $S^{(1)}, S^{(2)}, \dots, S^{(k)}$</p> <p>3: Assign each agent x to the nearest cluster $S^{(i)}$ based on the following assumptions: $\arg \min_{c^{(i)}} \sum_{j=1}^9 c_j^{(i)} - x_j$</p> <p>4: Recompute the centroids by taking the mean of all agents assigned to that centroid cluster: $c_j^{(i)} = \frac{1}{ S^{(i)} } \sum_{x \in S^{(i)}} x_j$</p> <p>5: Repeat steps 3 and 4 until the centroids no longer move.</p> <p>6: Calculate the mean performance for each class: $y_S = \frac{1}{ S^{(i)} } \sum_{x \in S^{(i)}} \sum_{j=1}^9 \alpha_j * x_j$</p> <p>7: Put into the set BA the agents from class S with the highest value of the performance y_S</p> <p>End Return BA</p>	<p>Begin</p> <p>1: Run the algorithm 4.</p> <p>2: Calculate $y_{max} = \frac{1}{ BA } \sum_{x \in BA} \sum_{j=1}^9 \alpha_j * x_j$</p> <p>3: For each agent z from the set BA if $\frac{1}{ BA \setminus \{z\} } \sum_{x \in BA \setminus \{z\}} \sum_{j=1}^9 \alpha_j * x_j \geq y_{max}$ then $BA = BA \setminus \{z\}$</p> <p>End</p> <p>Return BA</p>

The simplest is Algorithm 1, which serves as the reference algorithm to compare with others. Algorithm 2 depends on choosing agents with the best performance value. However, in comparison with Algorithm 1, this algorithm does not need to get the number of agents in an arbitrarily way. Algorithm 3 depends on a well-known from the literature *k-means* algorithm. This algorithm is very simple; however, it requires to get the number of classes. Algorithm 4 is a simple modification of Algorithm 3. It depends on removing agents that spoil the performance of the final decision.

5 Research Experiment

For the purpose of performing the research experiment, the following assumptions were made:

1. GBP/PLN M1 quotations were randomly selected for four periods: 16-04-2018, 00:00 to 17-04-2018, 23:59, 19-04-2018, 00:00 to 20-04-2018, 23:59, 07-05-2018, 00:00 to 08-05-2018, 23:59, 21-05-2018, 00:00 to 22-05-2018, 23:59.
2. Simple look-back window is used for testing algorithms, i.e. the first period is used for selection agents, which determine decisions in the second period.
3. Signals for open long/close short position equals 1, close long/open short position equals -1) were generated by the Supervisor agents, which implements certain algorithms. The Buy and Hold (B&H) strategy were used as a benchmark (trader buys on the beginning of a given period and sell at the end of this period).
4. The unit of the agents' performance analysis was pips.
5. The transaction costs were directly proportional to the number of transactions.
6. The investor engages 100% of the capital held in each transaction.
7. The measures (ratios) and the evaluation function presented in Sect. 4 were used in performance analysis, which can be found in Table 1.

Taking into consideration the values of the evaluation function, in the second period, Algorithm 1 was the best one. In the third period, Algorithm 4 was the best. In the fourth period the Algorithm 2 was the best. The Buy and Hold benchmark were ranked lowest in all periods. The values of obtained ratios are characterized by a very large distribution. For example, the number of transactions in the period 1 in case Standard Algorithm of consensus determining (without a selection of agents) equals 65 and in case Algorithm 3 it equals 103. Therefore, a transactions' cost in case of using Algorithm 3 is twice as high as the transactions' cost in case of using the Standard Algorithm.

The very important ratio is the number of unprofitable consecutive transactions. If it is higher than 2 than most often the strategy must perform a market exit. Algorithms characterized by the highest rate of the return not always were evaluated as the best, because also other ratios, such as risk ratios or an average rate of the return per transaction, were taken into consideration.

Reducing the risk level is a very important task of investment strategies. All Algorithms from 1 to 4 are characterized by a higher number of transactions, in comparison with the standard consensus determination algorithm. It may be a result of a fact, that the convergence of agents' decisions is not taken into a consideration in proposed algorithms. After the selection, the set of agents may consist of a large number of agents characterized by a very high level of decision's convergence. Agents with the highest level of this convergence could be removed during the selection process. Therefore, the evaluation function value and ratio values cannot be the only selection criterion. However, for most of the investors, the plausible method of evaluating a stock exchange strategy is a final rate of return. Utilizing any of the proposed algorithms (1–4) gives better results than a strategy without agents' selection. A total rate of return is the highest for Algorithm 4 (and equal 6212) for periods 2–4. The second contestant (with a slightly lower total rate of return equal to 6149) is Algorithm 1. The result obtained by the Buy and Hold (B&H) strategy is only 5892.

Table 1. The trading performance

Ratio	Standard Algorithm				Algorithm 1				Algorithm 2				Algorithm 3				Algorithm 4				B & H			
	Period 1	Period 2	Period 3	Period 4	Period 2	Period 3	Period 4	Period 1	Period 2	Period 3	Period 4	Period 2	Period 3	Period 4	Period 1	Period 2	Period 3	Period 4	Period 1	Period 2	Period 3	Period 4		
Rate of return [Pips]	153	2020	3563	-136	2576	3676	-103	2283	3620	-98	2638	3489	-315	2736	3592	-116	3454	3454	-1336	2697	3454	-259		
Number of transactions	56	65	105	44	83	126	56	78	123	47	103	138	67	98	112	48	1	1	1	1	1	1		
Gross profit [Pips]	78	158	163	144	186	174	168	164	189	148	179	185	192	193	239	151	0	2697	3454	0	0	0		
Gross loss [Pips]	-93	-110	-154	-236	-129	-103	-207	-132	-147	-117	-127	-154	-147	-147	-175	-138	0	0	-1336	0	0	-259		
Number of profitable transactions	38	47	28	23	59	64	39	52	68	28	68	75	37	69	74	23	0	1	0	1	1	0		
Number of profitable consecutive transactions	11	6	5	5	9	13	8	7	10	6	14	13	4	16	17	6	0	1	0	1	1	0		
Number of unprofitable consecutive transactions	2	2	1	4	2	3	3	2	3	2	3	4	3	2	3	4	1	0	1	0	0	1		
Sharpe ratio	0.68	0.52	0.46	0.39	0.36	0.32	0.34	0.34	0.29	0.37	0.33	0.21	0.34	0.48	0.40	0.27	0	0	0	0	0	0		
Average coefficient of volatility	0.27	0.34	0.27	0.15	0.42	0.35	0.27	0.38	0.42	0.29	0.29	0.48	0.42	0.25	0.28	0.41	0	0	0	0	0	0		
Average rate of return per transaction	2.73	31.07	33.93	-3.09	31.04	29.17	-1.84	29.27	29.43	-2.09	25.61	25.28	-4.70	27.91	32.07	-2.41	-1336	2697	3454	2697	3454	-259		
Value of evaluation function (y)	0.62	0.24	0.41	0.29	0.41	0.37	0.41	0.32	0.39	0.43	0.38	0.36	0.30	0.46	0.42	0.39	0.12	0.19	0.21	0.19	0.21	0.04		

6 Conclusions

We presented four different heuristic algorithms for the problem, built on top of a variety of well-known tools. Some results of the experimental comparison were presented. It should be noted, that in the case of High-Frequency Trading the computational complexity of agents' selection algorithms should as low as possible. The algorithms considered in this paper allow for achieving better performance than the consensus determined without agents' selection. The total rate of return gathered by some of our algorithms is higher than the benchmark strategy. However, they generate a larger number of transactions, which may entail a higher transaction cost. Therefore, the agents' evaluation based only on historical data is not sufficient.

The problem of the evaluation function definition is still open. The linear function is not the best solution from a scientific point of view, but it is easy to use from the investors' point of view. We want to address this issue in our upcoming research.

References

1. Barbosa, R.P., Belo, O.: Multi-agent forex trading system. In: Håkansson, A., Hartung, R., Nguyen, N.T. (eds.) *Agent and Multi-agent Technology for Internet and Enterprise Systems*, pp 91–118. Studies in Computational Intelligence, vol. 289. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13526-2_5
2. Bohm, V., Wenzelburger, J.: On the performance of efficient portfolios. *J. Econ. Dyn. Control* **29**(4), 721–740 (2005)
3. Ivanović, M., Vidaković, M., Budimac, Z., Mitrović, D.: A scalable distributed architecture for client and server-side software agents. *Vietnam J. Comput. Sci.* **4**(2), 127–137 (2017)
4. Khosravi, H., Shiri, M.E., Khosravi, H., Iranmanesh, E., Davoodi, A.: TACTic-a multi behavioral agent for trading agent competition. In: Sarbazi-Azad, H., Parhami, B., Miremadi, S.-G., Hessabi, S. (eds.) *CSICC 2008. CCIS*, vol. 6, pp. 811–815. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-89985-3_109
5. Korczak, J., Hernes, M., Bac, M.: Risk avoiding strategy in multi-agent trading system. In: *Proceedings of Federated Conference Computer Science and Information Systems (FedCSIS)*, Kraków, pp. 1131–1138 (2013)
6. Korczak, J., Hernes, M., Bac, M.: Collective intelligence supporting trading decisions on FOREX market. In: Nguyen, N.T., Papadopoulos, G.A., Jędrzejowicz, P., Trawiński, B., Vossen, G. (eds.) *ICCCI 2017. LNCS (LNAI)*, vol. 10448, pp. 113–122. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67074-4_12
7. Kozierekiewicz-Hetmańska, A., Pietranik, M.: The knowledge increase estimation framework for ontology integration on the concept level. *J. Intell. Fuzzy Syst.* **32**, 1–12 (2016)
8. Sycara, K.P., Decker, K., Zeng, D.: Intelligent agents in portfolio management. In: Jennings, N., Wooldridge, M. (eds.) *Agent Technology*, pp. 267–282. Springer, Heidelberg (2002). https://doi.org/10.1007/978-3-662-03678-5_14
9. Tatikunta, R., Rahimi, S., Shrestha, P., Bjursel, J.: TrAgent: a multi-agent system for stock exchange. In: *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IATW 2006)*, pp. 505–509. IEEE Computer Society, Washington, DC (2006)