Developing Actionable Trading Strategies for Trading Agents

Longbing Cao, Chao Luo, Chengqi Zhang
Faculty of Information Technology
University of Technology, Sydney, Australia
{lbcao, chaoluo,chengqi}@it.uts.edu.au

Abstract

Trading agents are very useful for developing and back-testing quality trading strategies for actions taking in the real world. However, the existing trading agent research mainly focuses on simulation using artificial data and market models. As a result, the actionable capability of developed trading strategies is often limited. In this paper, we analyze such constraints on developing actionable trading strategies for trading agents. These points are deployed into developing a series of trading strategies for trading agents through optimizing, and enhancing actionable trading strategies. We demonstrate working case studies in large-scale of market data. These approaches and their performance are evaluated from both technical and business perspectives.

1. Introduction

Artificial financial market [1,6,16] provides an economic, convenient and effective electronic marketplace (also called e-market) for the development and back-testing of actionable strategies taking by trading agents without losing a cent. A typical simulation drive is the Trading Agent Competition [8,16], for instance, research work on auction-oriented protocol and strategy design [11], bidding strategy [12], design tradeoffs [9], and multi-attribute dynamic pricing [10] of trading agents. However, existing trading agent research presents a prevailing atmosphere of academia. This is embodied in aspects such as artificial data, abstract trading strategy and market mechanism design, and simple evaluation metrics. In addition, little research has been done on strategy optimization in continuous e-markets, while which consist of our daily financial life.

The above atmosphere has led to a big gap between research and business expectation. As a result, the developed techniques are not necessarily of business interest or cannot support business decision-making. In fact, the development of actionable strategies is a non-trivial task due to domain knowledge, constraints and expectation in the market [4]. Very few studies [18] on continuous e-markets have been conducted for actionable trading strategies in the above constrained practical scenarios [2,13,5]. Therefore, it is a very practical challenge and driving force to narrow down the gap towards workable trading strategies for action-taking to business advantage.

An *actionable* trading strategy can assist trading agents in determining *right actions at right time with right price and volume on right instruments to maximize the profit while minimize the risk.* The development of actionable strategies targets an appropriate combination or optimization of relevant attributes such as target market, timing, actions, pricing, sizing and traded objects based on proper business and technical measurement. The above combination and optimization in actionable trading strategy development should consider certain market microstructure and dynamics, domain knowledge and justification, as well as investors' aims and expectation. These form the constrained environment in developing actionable strategies for trading agents.

Following our previous work in developing actionable trading strategies [2,5,4,13,17] and an agent service-based trading support platform F-TRADE [19], this paper discusses lessons learnt in actionable trading strategy for trading agents in continuous markets based on our years of research and development. The main contributions consist of (1) discussing real-life constraints that need to be cared in trading agent research, (2) proposing an actionable trading strategy framework, and (3) investigating a series of approaches to actionable strategy discovery.

Following the above ideas, we study a few effective techniques for developing actionable trading strategies in continuous market. These include designing and discovering quality trading strategies, and enhancing the actionable performance of a trading strategy through analyzing its relationship with target stocks. All of these methods are simulated and back-tested in an agent service-based artificial financial market F-Trade with online connection to multiple market data. The experiments show that the introduced techniques have the potential for improving the actionability of trading strategies when they are deployed into the real market.

2. Constraints on actionable strategy development

Trading agents simulating the real-world business would be highly constraint-based. Participating agents should be capable of learning the domain-specific scenario to achieve their goals through accessing data while obeying the institutional rules. In this process, a trading agent is often constrained by technical, economic and social restrictions.

2.1. Domain constraint

The actionable capability of a trading agent following certain trading strategies is tightly coupled with the simulated market microstructure [14]. Organization factors of trading agent institution should consider the following fundamental aspects. We

0-7695-3027-3/07 $25.00 © 2007 IEEE
DOI 10.1109/IAT.2007.16

call this *domain constraint* $M = \{I, A, O, T, R, S, E\}$.

- the traded *instruments* I, such as stock or derivatives, I={*stock*, *option*, *feature*, …}
- the market *participants* A, namely agents in e-market, e.g. major types of trading agents include investors such as individuals, mutual funds, money managers, as well as financial traders like brokers and market makers, A={*investor*, *broker*, *maker*, …}
- the *order* forms O, e.g., limit order, market order, block order, O={*limit*, *market*, *quote*, *block*, *stop*}
- the trading *session*, whether includes call market session and continuous session, it is indicated by timeframe T
- the market *rules* R, e.g., restrictions on order execution
- trading *strategy* S executed by trading agents, S may take varying forms
- *execution* system E, e.g., order-driven or quote-driven

An e-market instantiates a certain combination m ($m \in M$) of the above organization factors, and forms a specific microstructure niche for trading agents. The specified market microstructure further determines the behavior of a trading agent. From the trading perspective, it directly affects the trading strategies that a trading agent can take.

2.2. Trading strategy design problem

The development of actionable trading strategies is a multi-attribute optimization problem in a constrained scenario, namely trading agents determine right action at right time with right price and volume size on right instruments to satisfy multi-attribute constraints. A trading strategy set Ω is defined as a tuple $\Omega = <T, B, P, V, I>$ where $T = \{t_1, t_2, \ldots, t_m\}$ is a set of all appropriate trading *time* triggering signals; B = {*buy*, *sell*, *hold*} is the set of all executable *behavior* (i.e., trading signals) of trading agents; $P = \{p_1, p_2, \ldots, p_m\}$ is the set of trading *price* matched with corresponding trading time; $V = \{v_1, v_2, \ldots, v_m\}$ is the set of trading *volume* size; and $I=\{i_1, i_2, . . . , i_m\}$ is a set of traded instruments. The optimized combinations of features T, B, V, P and I results in a set of actionable trading strategies Ω' ($\Omega' \subset \Omega$) that satisfy certain design and business objectives.

The set Ω' indicates all smart decision states of a trading agent a ($a \in A$, $A = \{a_1, a_2, \ldots, a_n\}$ is the participating trading agent set) in the market. The set Ω' is also highly affected by the real-world market microstructure, dynamics and constraints, as well as an investor's motivation and aims. All these aspects form a constraint set: $\sum = \{ \delta_i^k | c_i \in C, k \in N\}$ where δ_i^k stands for the k-th constraint state of a constraint type c_i; $C=\{M, D, Int, \ldots\}$ is a set covering all types of constraints in the market; N is natural integer set. Therefore, the trading strategy set Ω' is a conditional function of $\sum$, which is described as $\Omega' = \{(\omega, \delta) | \omega \in \Omega, \delta \in \{(\delta_i^k, a) | \delta_i^k \in \sum, a \in A\}\}$, where ω is an optimal strategy instance, δ is all constraint instances on the strategy taking by a trading agent a. To work out the actionable set Ω', its back-testing and simulation is better to be undertaken in real stock data, then deploy the findings to e-market for further study.

2.3. Measuring strategy actionability

Let $X = \{x_1, x_2, . . . , x_m\}$ be a set of items, DB be a database that consists of a set of transactions, x is an itemset in DB. Let S be an interesting strategy discovered in DB through a modeling method M. The following concepts are developed for the actionable strategy discovery.

DEFINITION 1. Technical Interestingness *Tint*– The technical interestingness measure *tech_int*() of a trading pattern is highly dependent on certain technical measure of interest specified for a data mining method. It could be a set of criteria. For instance, the following logic formula indicates that an associated strategy S is technically interesting if it satisfies *min_support* and *min_confidence*.

$$\forall x \in X, \exists S: x.min\_support(S) \wedge x.min\_confidence(S) \rightarrow x.tech\_int(S)$$

DEFINITION 2. Business Interestingness *Bint*– The business interestingness measure *biz_int*() of a strategy is determined by some domain-oriented social and/or economic criteria. Similar to technical interestingness, business interestingness is also represented by a collection of criteria. For instance, if the *profit* and *roi* (*return on investment*) of a stock price predictor S are satisfied, then S is interesting to trading.

$$\forall x \in X, \exists S: x.profit(S) \wedge x.roi(S) \rightarrow x.biz\_int(S)$$

DEFINITION 3. Actionability of a strategy – Given a strategy S, its actionable capability *act*() is described as to what degree it can satisfy both the technical and the business interestingness. If both technical and business interestingness or a hybrid interestingness measure integrating both aspects are satisfied, it is called an *actionable* strategy. It is not only interesting to data modelers, but generally interesting to traders.

$$\forall x \in X, \exists S: x.tech\_int(S) \wedge x.biz\_int(S) \rightarrow x.act(S)$$

Actionable strategies can be created through rule reduction, model refinement or parameter tuning by optimizing technically interesting trading patterns. They may also be directly discovered from data set with sufficient consideration of business constraints. The next section discusses discovering actionable trading strategies from a great number of generic rules.

3. Developing actionable trading strategies

Following the ideas introduced in Section 3, this section illustrates some of approaches for developing actionable trading strategies for trading agents. They consist of optimizing trading strategies, enhancing trading strategies, and discovering trading strategies. We demonstrate their promising business performance in real tick-by-tick data.

3.1. Optimizing trading strategies

In trading agent design, there are huge quantities of variations and modifications of a generic trading strategy by parameterization. For instance, MA(2, 50, 0.01) and MA(10, 50, 0.01) refer to two different strategies. However, it is not clear to a trader which specific rule is more actionable for his or her particular investment situation. In this case, trading strategy optimization may generate an optimal trading rule from the generic rule set.

Optimizing trading strategies is to find trading strategies with better target performance. This can be through developing varying optimization methods. Genetic Algorithm (GA) is a valid optimization technique, which can be used for searching combinations of trading strategy parameters satisfying user-specified performance [13]. However, a simple use of GA may not necessarily lead to trading strategies of business interest. To this end, domain knowledge must be considered in fitness function design, search space and speed design, etc. The fitness function we used for strategy optimization is Sharpe Ratio (*SR*).

$$SR = (R_p - R_f) / \sigma_p ,$$

Where R_p is the expected portfolio return, R_f is the risk free rate, and σ_p is the portfolio standard deviation. When *SR* is higher, it indicates higher return but lower risk.

Figure 1 illustrates some results of GA-based trading strategy optimization. The trading strategy is Filter Rule Base, namely FR(δ). It actually indicates a generic class of correlated trading strategies, by which you go long on the day that the price rises by δ% and hold until the price falls δ%, at which time you close out and go short, where $\delta \in [0,1]$ the percentage price movement of highest high and lowest low.

TRADING STRAGE 1: A generic strategy FR(δ)
At time point *t*, get *high*(*t*) and *low*(*t*)
  IF *price*(*t-1*) > *high*(*t-1*)
    high(*t*)= *price*(*t-1*)
  ELSE
    high(*t*)= *high*(*t-1*)
  IF *price*(*t-1*) < *low*(*t-1*)
    low(*t*)= *price*(*t-1*)
  ELSE
    low(*t*)= *low*(*t-1*)
Generate trading signals
  IF *price*(*t*) < *high*(*t*)*(1- δ)
    Generate *SELL signal*
  IF *price*(*t*) > *low*(*t*)*(1+ δ)
    Generate *BUY signal*

In this rule, there is only one parameter δ, which can be used for optimization because δ is hard to be managed well in real-life market. Figure 1 shows the optimization results of the stock Australian Commonwealth Bank (CBA) in Australian Stock Exchange (ASX) in 2003~2004. It shows that from 14 July 2003, the cumulative payoff with δ = 0.04 always beats other δs.

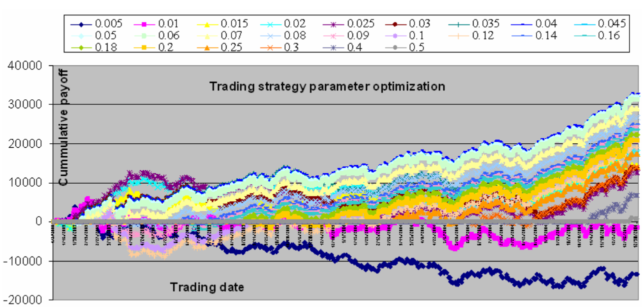


Figure 1. Some results of GA-based trading strategy optimization

3.2. Enhancing trading strategies

In many real-life cases, a given trading strategy may not work well due to missing considerations of some organizational factors and constraints. To this end, we need to enhance a trading strategy by involving real-life constraints and factors. For instance, the above rule FR(δ) does not consider the noise impact of false trading signals and dynamic difference between high and low sides. These aspects can be reflected into the rule by introducing new parameters.

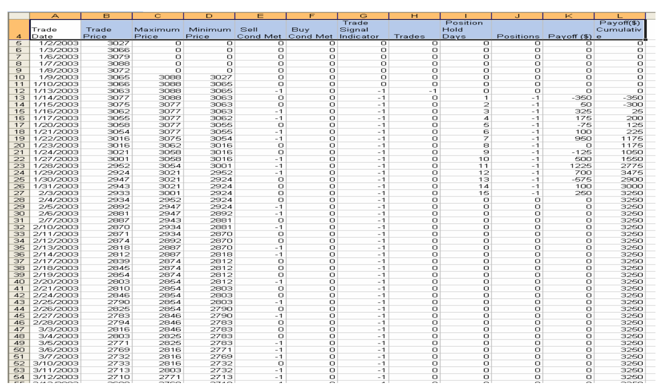
Enhancing trading strategies is not a trivial task. It needs to consider domain knowledge and expert advice, massive back-testing in historical data, and mining hidden trading patterns in market data. Otherwise a developed strategy likely does not make sense to business. For instance, we create an Enhanced Filter Rule FR(t, δ_H, δ_L, h, d) as follows.

TRADING STRATEGY 2: An enhanced FR(*t*, δ_H, δ_L, *h*, *d*)
At time point *t*, get *high*(*t*) and *low*(*t*)
  IF *price*(*t-1*) > *high*(*t-1*)
    high(*t*)= *price*(*t-1*)
  ELSE
    high(*t*)= *high*(*t-1*)
  IF *price*(*t-1*) < *low*(*t-1*)
    low(*t*)= *price*(*t-1*)
  ELSE
    low(*t*)= *low*(*t-1*)
Generate trading signals
  IF *price*(*t*) < *high*(*t*)*(1- δ_H)
    Generate *SELL signal*
    IF *position*(*t-1*) <> 0 & *hold*(*t-1*) = *h*
      position(*t*) = 1
  IF *price*(*t*) > *low*(*t*)*(1+ δ_L)
    Generate *BUY signal*
    IF *position*(*t-1*) <> 0 & *hold*(*t-1*) = *h*
      position(*t*) = -1

This enhanced version considers the following domain-specific aspects, which make it more adaptive to the real market dynamics compared with the generic rule MA(*sr*, *lr*, δ).

- More filters are imposed on the generic FR to filter out false trading signals which would result in losses, say fixed percentage band filter δ_H and δ_L for high and low price movement respectively, and time hold filter *h*;
- The fixed band filter δ_H (or δ_L) requires the buy or sell signal to exceed *high* or *low* by a fixed multiplicative band δ_H (or δ_L);
- The time hold filter *h* requires the buy or sell signal to hold the long or short position for a pre-specified number of transactions or time *h* to effectively ignore all other signals generated during that time;

Figure 2 shows the trading results of a trading agent taking the above strategy in ASX data 2003~2004.



	A	B	C	D	E	F	G	H	I	J	K	L
4	Trade Date	Trade Price	Maximum Price	Minimum Price	Sell Cond Met	Buy Cond Met	Trade Signal Indicator	Trades	Position Hold Days	Positions	Payoff ($)	Payoff($) Cumulative
5	1/2/2003	3027	0	0	0	0	0	0	0	0	0	0
6	1/3/2003	3066	0	0	0	0	0	0	0	0	0	0
7	1/6/2003	3079	0	0	0	0	0	0	0	0	0	0
8	1/7/2003	3088	0	0	0	0	0	0	0	0	0	0
9	1/8/2003	3072	0	0	0	0	0	0	0	0	0	0
10	1/9/2003	3065	3088	3027	0	0	0	0	0	0	0	0
11	1/10/2003	3066	3088	3065	0	0	0	0	0	0	0	0
12	1/13/2003	3063	3088	3065	-1	0	-1	-1	0	0	0	0
13	1/14/2003	3077	3088	3063	0	0	-1	0	1	-1	-350	-350
14	1/15/2003	3075	3077	3063	0	0	-1	0	2	-1	50	-300
15	1/16/2003	3062	3077	3063	-1	0	-1	0	3	-1	325	25
16	1/17/2003	3055	3077	3062	-1	0	-1	0	4	-1	175	200
17	1/20/2003	3058	3077	3055	0	0	-1	0	5	-1	-75	125
18	1/21/2003	3054	3077	3055	-1	0	-1	0	6	-1	100	225
19	1/22/2003	3016	3075	3054	-1	0	-1	0	7	-1	950	1175
20	1/23/2003	3016	3062	3016	0	0	-1	0	8	-1	0	1175
21	1/24/2003	3021	3058	3016	0	0	-1	0	9	-1	-125	1050
22	1/27/2003	3001	3058	3016	-1	0	-1	0	10	-1	500	1550
23	1/28/2003	2952	3054	3001	-1	0	-1	0	11	-1	1225	2775
24	1/29/2003	2924	3021	2952	-1	0	-1	0	12	-1	700	3475
25	1/30/2003	2947	3021	2924	0	0	-1	0	13	-1	-575	2900
26	1/31/2003	2943	3021	2924	0	0	-1	0	14	-1	100	3000
27	2/3/2003	2933	3001	2924	0	0	-1	0	15	-1	250	3250
28	2/4/2003	2934	2952	2924	0	0	-1	0	0	0	0	3250
29	2/5/2003	2892	2947	2924	-1	0	-1	0	0	0	0	3250
30	2/6/2003	2881	2947	2892	-1	0	-1	0	0	0	0	3250
31	2/7/2003	2887	2943	2881	0	0	-1	0	0	0	0	3250
32	2/10/2003	2870	2934	2881	-1	0	-1	0	0	0	0	3250
33	2/11/2003	2871	2934	2870	0	0	-1	0	0	0	0	3250
34	2/12/2003	2874	2892	2870	0	0	-1	0	0	0	0	3250
35	2/13/2003	2818	2887	2870	-1	0	-1	0	0	0	0	3250
36	2/14/2003	2812	2887	2818	-1	0	-1	0	0	0	0	3250
37	2/17/2003	2839	2874	2812	0	0	-1	0	0	0	0	3250
38	2/18/2003	2845	2874	2812	0	0	-1	0	0	0	0	3250
39	2/19/2003	2854	2874	2812	0	0	-1	0	0	0	0	3250
40	2/20/2003	2803	2854	2812	-1	0	-1	0	0	0	0	3250
41	2/21/2003	2810	2854	2803	0	0	-1	0	0	0	0	3250
42	2/24/2003	2846	2854	2803	0	0	-1	0	0	0	0	3250
43	2/25/2003	2790	2854	2803	-1	0	-1	0	0	0	0	3250
44	2/26/2003	2825	2854	2790	0	0	-1	0	0	0	0	3250
45	2/27/2003	2783	2846	2790	-1	0	-1	0	0	0	0	3250
46	2/28/2003	2794	2846	2783	0	0	-1	0	0	0	0	3250
47	3/3/2003	2816	2846	2783	0	0	-1	0	0	0	0	3250
48	3/4/2003	2803	2825	2783	0	0	-1	0	0	0	0	3250
49	3/5/2003	2771	2825	2783	-1	0	-1	0	0	0	0	3250
50	3/6/2003	2769	2816	2771	-1	0	-1	0	0	0	0	3250
51	3/7/2003	2732	2816	2769	-1	0	-1	0	0	0	0	3250
52	3/10/2003	2733	2816	2732	0	0	-1	0	0	0	0	3250
53	3/11/2003	2713	2803	2732	-1	0	-1	0	0	0	0	3250
54	3/12/2003	2710	2771	2713	-1	0	-1	0	0	0	0	3250

Figure 2. Some results of enhanced trading strategy FR

Figure 3 further shows the performance difference between a base rule and its enhanced version. It indicates that more involvement of domain knowledge and organizational constraints can to most extent enhance the business performance (cumulative payoff in our case) of trading agents. Most importantly, the results generated by trading agents make more sense to traders.

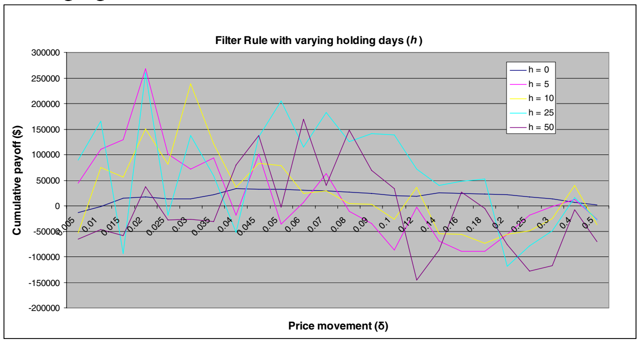


Figure 3. Performance comparison between base and enhanced trading strategies

4. Experiments

36 trading strategies have been developed for the study. They include classes of FR, MA, OBV, CB and SR. We did the experiments in 10 years (1997-2006) of stock interday data from five markets including HK, ASX, LSX, NYSE and SXE. The data was split into training and test sets based on a sliding window strategy. For instance, data in 1997-1998 is used for training, and 1998-1999 for testing in the first round. Then 1998-1999 is used for training and 1999-2000 for testing. The best strategy identified in training set is deployed to test set to see the performance. A payoff threshold is set for judging whether a strategy is positive or not. If the payoff of a strategy is bigger than the threshold, it is assumed *positive*, otherwise *negative*. We then define the following metric *lift* to measure the performance of a strategy in all data sets. The lift values are compared with that of randomly chosen. We choose 100 parameter combinations randomly by the computer and then calculate their payoffs following the same methods as used for the optimal one.

DEFINITION 7. Lift *Lift measures how much good a trading strategy is in all split data sets.*

$$Lift = \frac{\sum dataset\ strategy\ payoff\ above\ the\ threshold}{\sum split\ dataset}$$

Table 1. Lift values

	MA-CMN	FR-XY	OBV-B	CB-NXC	SR-NC
Lift (*Random*)	10%	0%	20%	10%	10%
Lift (*Optimal*)	70%	80%	80%	90%	100%

5. Conclusion

Trading agents have potential in providing traders with trading strategies that can support action-taken in the market given real-life constraints and organizational factors considered. However, existing trading agent research is mainly based on artificial data and artificial game mechanism design. However, the studied results may not necessarily be of business interest.

In this paper, we have developed actionable trading strategies for trading agents. The identified trading strategies have been tested in real continuous market data, and presented promising performance from not only technical but also business perspectives.

Acknowledgement

This work is sponsored by Australian Research Council Discovery Grant (DP0773412, LP0775041, DP0667060, DP0449535), and UTS internal grants.

References

[1] Arthur, W.B., Durlauf, S.N., Lane, D.A. *The Economy as an Evolving Complex System II*, Santa Fe Institute, Addison-Wesley, Volume 27 (1997), 583.

[2] Cao, L., Wang, J., Lin, L. and Zhang, C. Agent Services-Based Infrastructure for Online Assessment of Trading Strategies. *Proc of IAT2004*, IEEE press, 345-349.

[3] Cao, L., Ni, J., Wang, J. and Zhang, C. Agent Services-Driven Plug and Play in the F-TRADE. 17th Australian Joint Conference on Artificial Intelligence, *LNAI 3339* (2004), 917–922.

[4] Cao, L., Zhang, C., Two-Way Significance of Knowledge Actionability, *Int. J. of Business Intelligence and Data Mining*, 4, 2007.

[5] Cao, L.B., Zhang, C., F-Trade: An Agent-Mining Symbiont for Financial Services, *AAMAS2007.*

[6] Chan, T. *Artificial Markets and Intelligent Agents*. PhD thesis, Massachusetts Institute of Technology, 2001.

[7] Cheng, S-F, Leung, E., Lochner, K. M., O'Malley, K., Reeves, D. M., Schvartzman, L. J. and Wellman, M. P. Walverine: A Walrasian Trading Agent. *Proc. of AAMAS2003*, ACM, 465-472.

[8] Trading Agent Competition: http://www.sics.se/tac/

[9] Ioannis, A., Vetsikas, and Selman, B. A principled study of the design tradeoffs for autonomous trading agents. *Proc. of AAMAS2003*, 473-480.

[10] Dasgupta, P. and Hashimoto, Y. Multi-attribute dynamic pricing for online markets using intelligent agents, *Proc of AAMAS04*, ACM, 277-284.

[11] David, E., Azoulay-Schwartz, R. and Kraus, S. Protocols and strategies for automated multi-attribute auctions, *Proc. of AAMAS02*, 77-85.

[12] David, E., Azoulay-Schwartz, R. and Kraus, S. Bidders' strategy for multi-attribute sequential English auction with deadline, *Proc. of AAMAS2003*, 457-464.

[13] Lin, L., Cao, L.B., Mining in-depth patterns in stock market, *Int. J. of Intelligent Systems Technologies and Applications*, 2007

[14] Madhavan, A. Market Microstructure: A Survey. *Journal of Financial Markets*, (2000), 205–258.

[15] Wan, H.A., Hunter, A. *On Artificial Adaptive Agents Models of Stock Markets*, Simulation 68:5, 279-289.

[16] Wellman, M. P., et al. Designing the Market Game for a Trading Agent Competition. *IEEE Internet Computing*, 5(2) (2001), 43-51.

[17] Cao, LB., *Multi-Strategy Integration for Actionable Trading Agents*, Int. Workshop on Agent and Data Mining Interaction joint with WI-IAT2007, IEEE Computer Society Press, 2007.

[18] http://www.cis.upenn.edu/~mkearns/projects/pat.html

[19] www.f-trade.info.