# An agent strategy for automated stock market trading combining price and order book information

**Conference Paper** · January 2005

DOI: 10.1109/CIMA.2005.1662356 · Source: IEEE Xplore



2 authors:


Gheorghe Cosmin Silaghi
Babeş-Bolyai University



Valentin Robu
Heriot-Watt University

# An Agent Strategy for Automated Stock Market Trading Combining Price and Order Book Information

Gheorghe Cosmin Silaghi
Babeş–Bolyai University
Faculty of Economic Sciences
Cluj-Napoca, Romania
gsilaghi@econ.ubbcluj.ro

Valentin Robu
Centrum voor Wiskunde en Informatica
Kruislaan 413, NL-1098 SJ
Amsterdam, The Netherlands
robu@cwi.nl

## Abstract

*This paper proposes a novel automated agent strategy for stock market trading, developed in the context of the Penn-Lehman Automated Trading (PLAT) simulation platform [1]. We provide a comprehensive experimental validation of our strategy using historic order book data from the NASDAQ market.*

## 1. Introduction

Designing automated trading strategies has received an increasing amount of attention, as trading in many stock markets (e.g. NASDAQ in the US) has now become fully electronic. NASDAQ, for example, allows any individual or private company to place orders on the market, orders which are then matched through the so called Electronic Crossing Networks (ECNs), which are independent firms that provide essentially competing markets for NASDAQ stock [1].

Although the ECNs are easy to use and open to any investor - they are obviously **not** risk-free. For this reason, several simulation platforms have emerged to enable investors and interested researchers to test automated trading strategies, without the attached risks.

The algorithms and heuristics reported in this paper (like those reported in [1, 3, 4, 2], among others) are tested on the PLAT (Penn-Lehman Automating Trading Platform) (see [1] for an overview).

The Penn Exchange Simulator (PXS) (which is at the core of the PLAT) is a (simulated) trading platform that matches orders from two sources: orders from automated agents executing strategies on the PLAT server, as well as orders from the real stock market, which are made available (at 3 second intervals) from Island, one of the largest independent ECNs on the NASDAQ market. Hence, as shown in [1, 3], PLAT is among the most realistic simulators, because it includes real-time, real world data, as well as providing a "safe" test environment where researchers can test the impact different automated agent strategies may have on the market.

An innovative feature of PLAT is the availability of order book data. Order books (which list the buy/sell order queues at each point - see Sect. 2) have been part of stock markets since their beginning. However, only recently has order book information become public on the NASDAQ market (and it's still unavailable in other markets). Chiefly for this reason, most of the backtesting environments used by financial institutions today, include only price information and do not (yet) include order book data. PXS is one of the first platforms for simulated trading employing real-world order book data (cf. [1]).

The goal of this paper is to investigate the potential of combining more traditional price-based strategies (used for a long time in financial econometrics) with strategies based on order book information. Although several strategies of both types have already been implemented and tested in the context of PLAT [1, 3, 2], we feel there is considerable scope for designing strategies that include a combination of these several types of signals.

The strategy proposed in this work-in-progress report has been tested in real, historic order-book information from the NASDAQ (as provided by the PLAT server) - and the results have been compared with other strategies proposed in the literature on this data set. However it has yet to be tested in a direct competition with other strategies (competitions are organized on average twice a year). This is also the approach taken in other papers [3, 2], which first present an evaluation of their approach on historic data, and use this as a guide to the performance of their proposed strategies in a real competition. However, until the proposed strategies are evaluated in a direct competition with other strategies, we position this paper as a "work in progress" report.

The rest of this paper is organized as follows. In Sect. 2, we present a brief overview of the market microstructure and the functioning of the PXS platform, without which, we argue it may be difficult for a casual reader to understand the algorithms presented here. In section 3, we present the basic strategies used for automated trading and present our approach to combining them. Finally, Section 4 presents our experimental validation results and concludes.

## 2. Market Microstructure and Order Matching on PLAT

By market microstructure, we understand the computational and transactional details of the NASDAQ market, and in particular of the Island ECN and PXS simulator [1]. As mentioned in the introduction, the matching between buying and selling orders on the NASDAQ is entirely electronic and is performed through several large independent Electronic Crossing Networks (ECNs). In the case of the PLAT simulator (and for all experiments reported in this paper), the data used is for Microsoft stock (MSFT), as provided by the Island ECN.

There are two types of orders which can be placed on such a market: limit orders and market orders. Limit orders specify a quantity (volume) and a desired price - and therefore are only executed when matching order(s) with the same or more competitive prices are found. Market orders are orders to buy and sell a certain volume of shares at the price available on the market - and are executed as soon as they arrive.

When a buy/sell limit order arrives (at each moment or second in the trading day), they are placed into two queues, called respectively the buy/sell order books. Each of the two order books (buy/sell) consists of a list of orders, giving the volume and the price per shared asked. Orders books are sorted on prices: from the highest to lowest in the buy order book and from lowest to highest in the sell order book.

Orders are matched sequentially as they arrive, up to the available volumes, as long as the buy price is higher than the sell price (if a difference exists, then the matching is performed at the average of the two values). The exchange's last price (also referred to as the "ticker" price) is the price at which the last transaction took place. On Island (and hence the PXS platform), only the top 15 orders from each order book are available for viewing (and automatic querying by the agents). The order books on PXS are updated at intervals of roughly every 3 seconds with fresh orders from the real order books on Island. Simulations performed on the PXS are performed in two main modes:

- The historical mode. In this mode, an automated agent executes its buy/sell strategy against real order book data (from Island ECN), which is archived on the PLAT server.
- The live mode. In this mode, several automated agents (belonging to different participants) execute their strategy on the server by placing buy/sell orders, which are matched both against real orders from Island, as well as orders placed by other automated agents. This mode is used during competitions.

Each agent on PLAT owns a virtual portfolio, consisting both of (virtual) cash and share holdings. Agents start the simulation day with a value of zero for their portfolio. They can borrow any amount of money or shares for the day without interest - but with the essential condition that they liquidate all their share positions by the end of the day This is an important condition to insure that agents make a profit based on their technical trading algorithms and not merely because of positive trends in the market.

## 3. Design of an Automated Trading Strategy

### 3.1. Existing trading strategies: a review

In this section we start by very briefly reviewing some simple automated trading strategies, both from those order book data and those using only price information. The reason for this section is that our own trading algorithm (proposed in the next subsection) basically relies on combining the signals from one or more of these strategies. For reasons of space, our description cannot be very thorough and does not aim to be exhaustive - interested readers are asked to consult [1, 2, 3, 4] for further details.

**3.1.1. Static Order Book Imbalance (SOBI)** This strategy aims to make full use of available order book information. It works by first computing two volume-weighed averages of the prices contained in the buy and sell order books. Two differences are then computed from each average and the last price of the exchange. If the "sell-side" difference is larger than the "buy-side" difference by more than a threshold amount (theta dollars), this is an indication that prices are going to rise and a buy order is placed. Correspondingly, if the “buy side” difference is larger than the “sell side” by this threshold, a sell order is placed.

**3.1.2. Volume Average Weighed Prices (VWAP)** This is the second strategy to use order book information we considered [2]. In our implementation, it first computes a weighed price average for the whole market (a weighed averages of prices weighed by volumes, from both the buy and sell order books). This value is interpreted as a “true” equilibrium price for the market. Next, if the average price for the first orders in the “buy” book is higher than the market average (by a certain threshold), then it places a sell order. If the market average is higher than the price of the first order in the sell book, then it places a buy order.

**3.1.3. Trend Following (TF)** Trend following is a “classical” strategy, which uses only information from price movements. It computes (through linear regression with respect to time) two trend lines from the last “ticker” prices from two different time windows. The first of these is for a larger time window (in our case a 4 hour window) up to the current time, while the second is for a smaller window (of 1 hour). If the slope (or gradient) of the two trend line matches, for example if it’s positive for both, then the agent places a buy order. Once the nearer term trend reverses sign compared to the longer term trend, it starts liquidating its position.

**3.1.4. Reverse Strategy** Consider the following straightforward, trend following strategy: buy when the price is rising and sell when it is falling. The reverse strategy does exactly the opposite: it sells when the price is rising and buys when it is falling. Although this strategy appears counter-intuitive [3], it works by exploiting the price micro-movements (small price spikes in both direction), which make up the evolution of the price of a stock during each trading day.

### 3.2. Our approach: Towards a combined-weight decision strategy

Our approach to this problem was to investigate whether a trading strategy that combines signals from more than one source could perform better than strategies using only one criteria. In particular we wanted to see whether supporting the buy/sell decision from a price based strategy with that from an order book-based strategy can lead to good experimental performance.

We start by considering the 4 basic criteria described above. Each of these returns, for a given state of the market at any time point, a recommendation which is assigned a numerical value. In our current implementation, we assign 1 to a buy recommendation, 0 to neutral and -1 to sell (future work may consider more finely-grained strategies, where the buy/sell volumes recommended by each strategy are also taken into consideration, not just the direction).

Next, each criteria is assigned a weight, such that the weights add up to 1. By making a weighed sum from the 4 criteria (or 2 or 3 of them - depending how many we wish to consider), a number between -1 and 1 is obtained, which gives the “aggregate recommendation” from the 4 strategies. This is compared to two thresholds: if it is above a certain “buy” threshold, a buy order is placed, if it is below some “sell” threshold, a sell order is placed.

For clarity we illustrate this through an example. Suppose we consider a mixed decision strategy from 3 of the criteria described above: SOBI, reverse and trend follower (TF). We assign each of these criteria equal weights (namely 0.33). If the buy threshold is set to 0.66, then at least two out of the three strategies must give a “buy” signal in order for the agent to place a buy order. The sell threshold can be set at -0.66, to achieve the opposite effect. One can already see that, for equal weights, it is easy to transform our mechanism into a “majority voting” rule: if a majority of strategies give a buy or sell signal, that decision is taken.

## 4. Experimental results and discussion

The strategies described above were tested using historical order book data for Microsoft (MSFT) stock, for a consecutive 15-day period from 05.01.2004 to 23.01.2004. We first tested each of the individual 4 strategies on this data set and then we considered several mixed strategies, which combine signals from 2, 3 or all of the 4 criteria.

There are two important issues to be discussed when evaluing the performance of a trading strategy. One is the liquidation strategy, which all agents must implement due to the requirement that they liquidate their share positions at the end of each day (see Sec. 2 for a discussion of this issue). In our case, liquidation was simply implemented as a requirement to sell (or buy back) any deficit of shares in the last hour of trading (that is from 15.00 to 16.00). All strategies in our simulations used the same liquidation policy, to minimize the distortions between them, but further work is needed to determine if this policy affects some strategies more than others.

| Strategy | Profit | Sharpe ratio |
|---|---|---|
| SOBI only | 22357 | 0.2 |
| VWAP only | -35905 | -0.823 |
| Reverse only | 18014 | 0.92 |
| TF only | 14746 | 0.14 |
| All 4, Decision: 4/4 | 9281 | 0.3797 |
| SOBI, Reverse, Decis: 2/2 | 15111 | 0.261 |
| SOBI, Rev&TF, Decis: 2/3 | 37785 | 0.65 |

**Table 1. Experimental results for MSFT data from 05.01.2004 to 23.01.2004**

Another important issue is the performance criteria. Using only raw profit over the 15 days is not a sufficient criteria, since in real-life a trade-off between risk and return of a strategy must also be taken into account. A commonly used measure of this is the Sharpe ratio, defined as the empirical daily average of returns divided by the standard deviation [1]. A higher Sharpe ratio shows consistently high earnings, with small spreads in returns. Figure 1 gives the result graphs for one of the days of the simulation, while Table 1 summarizes the results over the whole test period. In Table 1, "decision: 2/3" means 2 out of the 3 criteria considered have to signal a "buy/sell" in order for the buy, respectively sell decision to be actually taken (for experiments reported here, all criteria were assigned equal weights). Several combined strategies were tested, using different combinations of criteria and "majority voting" rules (partial results are reported in Table 1).

The best combined strategy we found is the one between SOBI, reverse and TF, with a 2/3 majority rule. However, the simple reverse ("contrarian") strategy is still the overall winner in terms of Sharpe ratio, although our combined strategy is the overall winner in terms of raw profit. On some days of the 15-day interval the combined strategies performed considerably better, though: one such example is 12.01.2004 (shown in Fig. 2), a day when all strategies lost money, except the combination between SOBI, reverse and TF[1].

We conclude that the preliminary experimental results are encouraging, but further work is needed to devise better ways to combine these strategies as well as to investigate, at a more fundamental level, the underlying reasons behind the observed empirical results.

1 The considerable negative result for the VWAP strategy (implemented by us in a simplified form) may simply be an indication that this strategy needs reimplementation. This does not affect the validity of other results, which do not use VWAP.

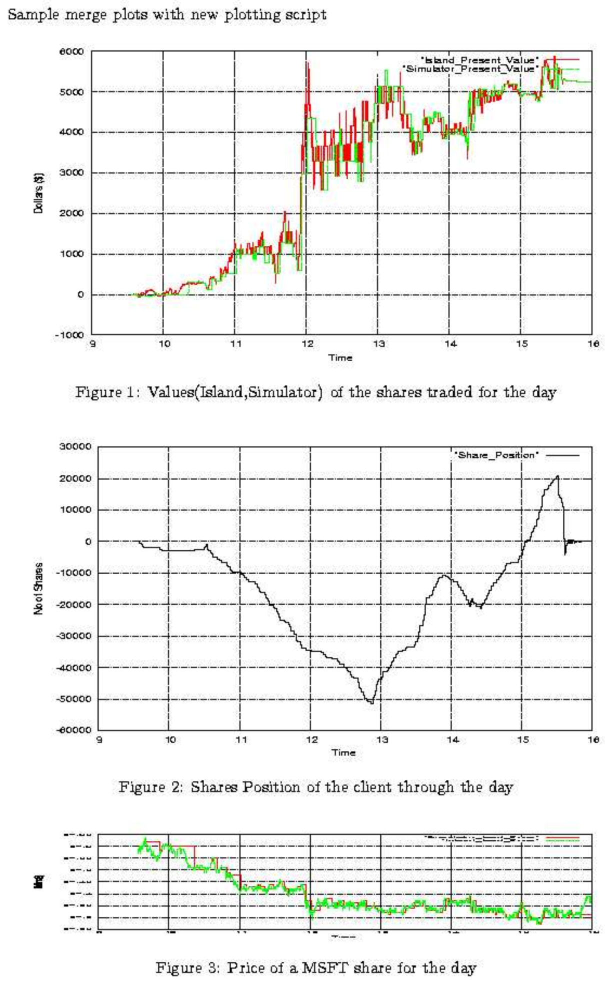


**Figure 1. Results for trading using the MSFT stock data for 12th January 2004**

# References

[1] Kearns, M., Ortiz, L. - "The Penn-Lehman Automated Trading Project", *IEEE Intelligent Systems*, No. 6, pp. 22–31, Nov/Dec 2003.

[2] Kakade, S., Kearns, M., Mansour, Y., Ortiz, L. "Competitive Algorithms for VWAP and Limit Order Trading". In *Proc. ACM Conf. Electronic Commerce*, 2004.

[3] Feng, Y., Yu, R. Stone, P., "Two Stock-Trading Agents: Market Making and Technical Analysis", *Proc of AMEC V Workshop*, Springer LNAI, pp. 18-36, 2004.

[4] Sherstov, A., Stone, P. "Three Automated Stock-Trading Agents: A Comparative Study" In *Proc. of the AMEC VI Workshop*, New York, USA, July 2004.

[5] Bouchard, J., Mezard, M., Potters, M. "Statistical Properties of Stock Order Books: Empirical Results and Models", *Quantitative Finance*, vol. 2, no. 4, Aug. 2002.