An Overview of an Automated Trading Software System

Ventsislav Nikolov

Abstract: *The paper illustrates the architecture and strategies used in an automated trading software system. Such a system can automatically buy, hold or sell shares at specific time moments according to the set strategy in order to obtain some profit. The architecture of the system is described together with the communication, decision making and strategy module.*

Keywords: *Automated trading, Decision making, Prediction, Financial information exchange*

Преглед на система за автоматична търговия

Венцислав Николов

Резюме: *В статията се представят архитектурата и стратегиите за действие в автоматизирана система за търговия във финансовата област. Такава система може автоматично да купува, задържа и продава акции в определени времеви моменти, в зависимост от зададената стратегия, с цел постигане на максимална печалба. Към модулната архитектура на системата са описани комуникацията, модулите за вземане на решения и стратегията за действие.*

Ключови думи: *Автоматична търговия, Вземане на решения, Прогнозиране, Обмен на финансова информация.*

1. Introduction

The automation of the human activities in different fields is increasing too fast today. It also tends to involve the decision making which naturally leads to using of intelligence computation software systems. Here an automation of a trading system is described as a software implementation which together with some modules can realize appropriate trading strategies and prediction of the market indicators. The system automatically generates orders to buy, sell or hold shares with given price and quantity at specific time moments according to the parameters and constraints of the chosen algorithm [2]. In order to do that information from the market and other sources should be obtained real-time. In this way no human intervention is needed for the system which relies on a software program which works infinitely. In addition, it should allow monitoring of the statistics, current status and other additional information. Generally, the technical design of such a system is not standardized [1] and the system should also be open for new options.
The technical details of the algorithmic trading system are shown in fig.1.

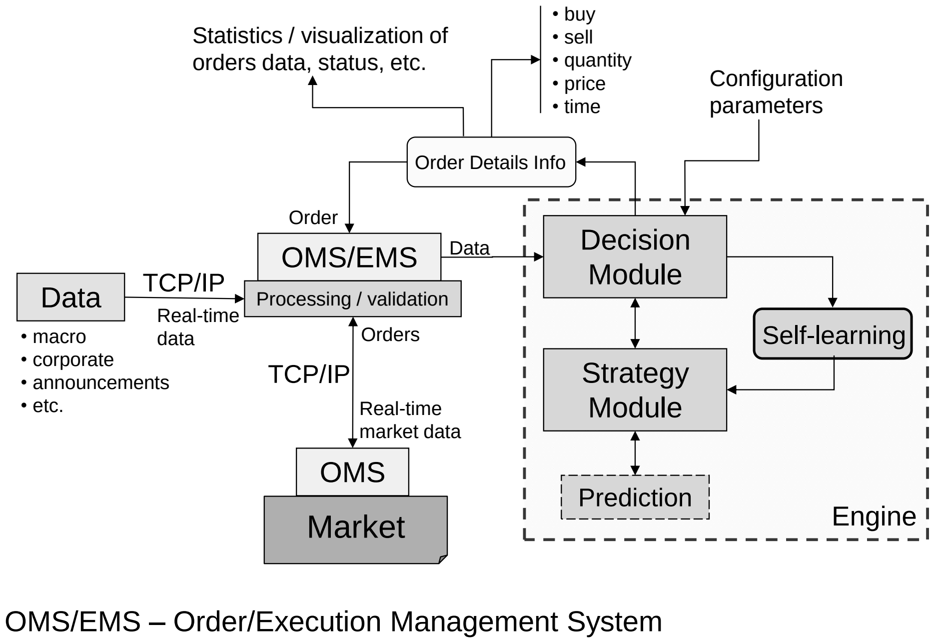

Fig. 1. Description (title) of the figure

The communication module connects the trading system with the rest of the world. It receives information and sends orders about the trades. If the system uses some intelligent approaches of analysing additional information like corporate data, companies' announcements, etc. it should be able to receive that data from certain reliable and authentic data sources. There are some important requirements about the communication which are as follows.

- High speed connection. The connection of the system should be very fast because the difference between successful and unsuccessful trade may be a matter of milliseconds.
- Low latency. The data should be obtained straight from the source providing very low latency.
- Capacity. The system should allow transfer of a huge amount of incoming and outgoing messages. If the capacity of the communication does not withstand to the loading, then it could lead to system algorithms frustration.
- Reliability and recoverability. This requirement concerns not only the communication infrastructure but also the ability of the software to recover after a communication crush.

The most important components of the trading system are described below.

2. Communication

One commonly used approach for data exchange is by using of the Financial Information eXchange (FIX) protocol. It is an open and free XML protocol for real-time communication using trade-related messages allowing sending and receiving orders as well as messages processing and validation. The data exchange is managed by an engine called FIX Engine which controls the session and application layers of the TCP/IP protocol stack. It also manages the network connections, creates and parses the information messages and recovers the system when an error occurs.

3. Decision making module

At the core of the trading system engine works a decision-making module. It is responsible for choosing and processing of assets, order type, price, quantity and time of order execution according to the configuration parameters. The orders are technically executed by the Order Management System (OMS) which receives all needed market and corporate data and sends the trading orders.
The module should be designed in a way allowing continuously improvement according to any additional client-specific requirements. Moreover, it should be adaptable, self-learning and flexible withstanding to the huge range of possible market scenarios. Thus, some soft computing techniques are appropriate to be applied to the decision-making module.

4. Strategy module

The strategy module performs the operations required by the strategies chosen by the Decision Making Module. There are many known strategies that could be used. Some of the most commonly used are shown below.

- Volume Weighted Average Price (VWAP). Parameters of this strategy are: start time, end time, max participation rate, level of aggression. In a given moment if the price is lower than intra-day VWAP average the corresponding share should be bought, and when the current price is higher than the average price it should be sold – fig.2. Thus, in short term there will not be loses because the profit range is kept by buying under the average and selling above the average.

Time	Bid price	Volume	VWAP
9:00	70	200	70.00
9:05	75	150	72.14
9:10	80	250	75.42
9:15	65	200	72.81
9:20	65	100	71.94

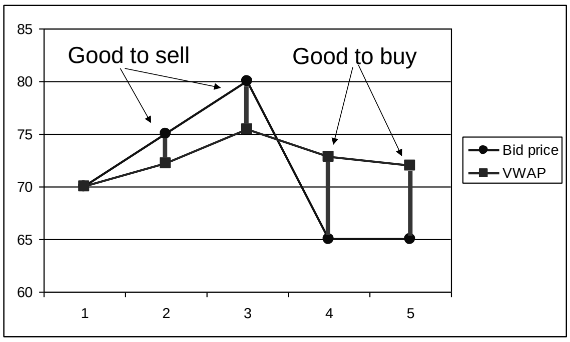

Fig.2. The VWAP approach used by the strategy module

- The VWAP is calculated in this strategy as shown in (1).

$$P_{vwap} = \frac{\sum_j P_j Q_j}{\sum_j Q_j} \tag{1}$$

where P_{vwap} is volume weighted average price
P_j – price of trade j
Q_j – quantity of trade j
j – individual trade that

VWAP is often used as a trading benchmark by investors that aim to be as passive as possible in their execution.

- If current price is below the VWAP benchmark up to the end of the chosen time horizon, the current bid price is determined as good to buying in, but bad for selling out;
- If current price is above the VWAP benchmark up to the end of the chosen time horizon, the current bid price is determined as good to selling out but bad for buying in.

Thus, there will not be loses in short term as a profit range is kept by buying under the benchmark and selling above the benchmark (e.g. based on intra-day VWAP).

- Mean reversion could be used similarly to the VWAP: the average price is computed for a specified period and when the current price is less than the average, the stock is considered as attractive for purchasing with the expectation that the price will rise, and vice versa, if the current price is expected to fall and it is good for selling if it is above the average price.
- Time weighted average price (TWAP). Parameters of this strategy are: start time, end time, max participation rate, level of aggression. This strategy is used to execute orders over a specific time so to keep the price close to that one which reflects the true market price, i.e. the objective is to minimize the market impact.
- Implementation shortfall. Parameters here are: start time, end time, reference price, participation rate aggressiveness, participation rate tolerance, limit price. This strategy is based on the terms:
  - Paper portfolio: represents the ideal case where the transaction costs, commission, etc. do not happen.
  - Actual portfolio: reflects the real markets situation where commissions, etc. are taken into account.
  - Implementation shortfall = Paper portfolio – Actual portfolio
    that is also: theoretical portfolio – implemented portfolio
- Volume participation. Parameters of this strategy are: start time, end time, participation rate and maximum tolerance from participation rate. The objective is to participate in volume as it arises. For example, if the participation rate is 10% and other market participants transacted 10000 shares then 1000 shared have to be executed.

Some other commonly used strategies are: Guerilla, Liquidity, Trend Following, Arbitrage, Mean Reversion, etc. These strategies have some specific parameters: start time, end time, max participation rate, level of aggression, participation rate tolerance and others.

The strategy module may refer to some specific computational modules [3, 4]. For example, as it is shown on fig.1, the prediction module is called from the strategy module. These specific mathematical computations are needed to realize the specific strategy.

The prediction could be performed for:

- Stock prices. The prediction is able to spot trend and patterns more quickly than any human could
- Volumes – in case VWAP is used as a benchmark. The daily volume is not known at the beginning of the day but could be predicted. Some techniques also include principal component analysis on the volumes beforehand.

The prediction generally includes data pre-processing, solving of matrix equations (in batch mode or iteratively) and data post-processing. Also because of the self-learning approach some new strategies may also be added and some of them need to use uni-variate or multi-variate prediction from the prediction sub-module in the engine.

When prediction is performed some indicators for the prediction quality should be calculated for preliminary separated test values. Usually additional confidence bounds are used as shown in grey colour around the predicted values in fig.3 where an example of historical values of a time series and its prediction for a time horizon of ten future values are shown. Usually the confidence levels are 95% or 97.5%.

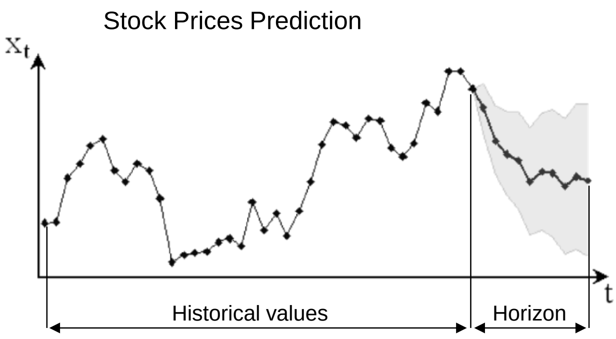

Fig.3 Example of time series and prediction for a given time horizon

Often only correct moving direction of the prediction is enough because if only 60-65% of the trades are successfully performed based on the predictions then it could be enough for profit to happen.
In Fig. 4 one of the most commonly used approach for time series prediction is shown using the sliding window approach.
There are different mathematical models which work using this approach. The Box-Jenkins approach, autoregressive methods as well as neural networks are amongst them. The work is separated in two stages: model identification and prediction. The first one is based on historical values analysis and the second uses the built model from the first stage in order to generate the predictions. The model identification is in fact an optimization problem and the prediction is filtering the input through the model.

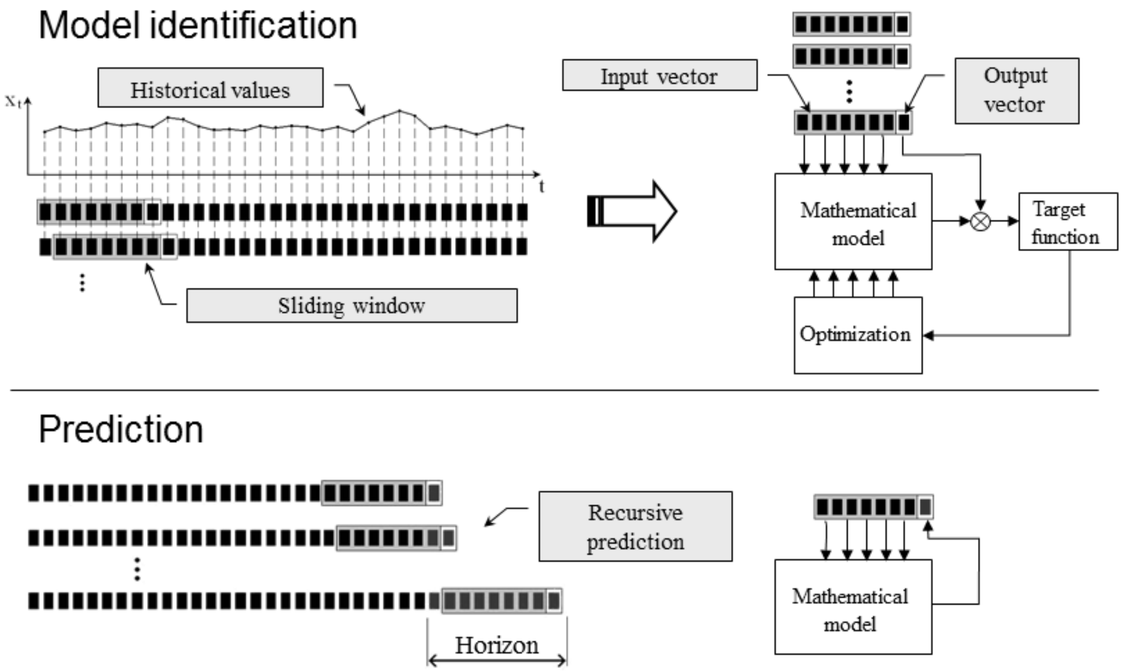

Fig.4. Fitting a mathematical model to historical values and prediction

Another approach for the prediction is performing of fundamental analysis in addition to the technical analysis.

5. Conclusions and future work

The implementation of the automated trading system is in progress and from scientific point of view the results of the trading simulations should be analyzed. They should be done without human intervention for long enough time periods. The results will generally depend on the strategies set in the system as well as on the properties and behaviour of the traded shares. The market situation may result in series of data with different variations and thus the different strategies may be appropriate or inappropriate in a given time moments.

References

[1]. Kim, Kendall. Electronic and algorithmic trading technology. The complete guide. Elsevier, 2007.

[2]. Algorithmic trading. An overview. Instinet, 2005

[3]. Baker, G., S. Tiwari. Algorithmic trading: perceptions and challenges. Edhec Risk Advisory, 2004.

[4]. Ward, S., Sherald, M. Successful Trading Using Artificial Intelligence. Ward Systems Group, 2006.

[5]. Rao, S. Algorithmic Trading: Pros and Cons. Tata Consultancy Services Limited, 2007.

For contacts:
Dr. Ventsislav Nikolov
Senior Software Developer
Eurorisk Systems Ltd.
31, General Kiselov Str., 9002 Varna, Bulgaria
E-mail: vnikolov at euroriskssystems dot com