World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:13, No:4, 2019

A Prediction Model Using the Price Cyclicality Function Optimized for Algorithmic Trading in Financial Market

Cristian Păuna

***Abstract*—**After the widespread release of electronic trading, automated trading systems have become a significant part of the business intelligence system of any modern financial investment company. An important part of the trades is made completely automatically today by computers using mathematical algorithms. The trading decisions are taken almost instantly by logical models and the orders are sent by low-latency automatic systems. This paper will present a real-time price prediction methodology designed especially for algorithmic trading. Based on the price cyclicality function, the methodology revealed will generate price cyclicality bands to predict the optimal levels for the entries and exits. In order to automate the trading decisions, the cyclicality bands will generate automated trading signals. We have found that the model can be used with good results to predict the changes in market behavior. Using these predictions, the model can automatically adapt the trading signals in real-time to maximize the trading results. The paper will reveal the methodology to optimize and implement this model in automated trading systems. After tests, it is proved that this methodology can be applied with good efficiency in different timeframes. Real trading results will be also displayed and analyzed in order to qualify the methodology and to compare it with other models. As a conclusion, it was found that the price prediction model using the price cyclicality function is a reliable trading methodology for algorithmic trading in the financial market.

***Keywords*—**Algorithmic trading, automated trading systems, financial markets, high-frequency trading, price prediction.

I. Introduction

TRADING and investing in financial markets is a common activity today. An important number of market participants are buying and selling every day in the free markets. Private and public investors, different types of companies and traders are continuously speculating the markets in order to make a profit. The main objective is to catch the price difference in time, to buy cheap and to sell more expensive using different trading and investing strategies.

"Nowadays, in the current challenging economic environment, businesses have changed their models in order to be more service oriented and serve a broader and global audience." [1] The trading and investment domain is one on the top in this long list. "The increases in complexity of the phenomena that characterize a firm's activity in general and of financial aspects in particular, have led to an exponential increase in the volume of data and information operated from any field of financial activity." [2] All of these issues involve new aspects to organize the activity. All are competing to new directions for the business environment in financial trading.

After the widespread release of electronic trading, "the role of the automated trading software in the business intelligence systems of any financial or investment company became significant." [3] An important part of the trades is set up completely automatically today by computers using advanced mathematical algorithms. The low-latency real-time automated systems are used today to build and send trading orders without any human intervention. Designing, testing and developing automated software for trading decisions has become a sustained activity nowadays and this is the field this paper is addressed for. In this article, a computational model will be revealed to predict the price evolution using the real-time price series. Being exclusively a mathematical algorithm, the methodology presented can be applied in any algorithmic trading system to automate the trading decisions. In the first part of the paper, the prediction model will be described and explained. Computing a price prediction line is the core of the model based on the price cyclicality function [4]. The prediction model is presented in the form of two bands, one for the stop loss and one for the take profit price levels. Some clear trading strategies will be developed based on this model together with the implementation steps into an automated trading system.

In the second part of this paper, the main direction will be revealed to integrate the developed model into an algorithmic trading system. The general logical scheme for automated trading software will be presented together with the steps to compute the Price Cyclicality Function. The automated trading signals are built based on the Price Prediction Line. The article will also include code examples to compute these functions. In order to present the reliability of the developed model, in the last part, real trading results obtained with this methodology will be included. A comparative study is also presented in order to compare the developed trading model with other known trading strategies. The study will permit to highlight the advantages of the price prediction bands methodology and its place in the algorithmic trading domain. In the last chapter, different practical conclusions will be presented. It was found that the methodology presented in this paper can obtain good results to predict the changes in price

C. Păuna is with the Economic Informatics Doctoral School, Bucharest University of Economic Studies, 11th Tache Ionescu str. 010352 Bucharest Romania (phone: +407.4003.0000; e-mail: cristian.pauna@ie.ase.ro).

This paper was co-financed by the Bucharest University of Economic Studies during the Ph.D. program and Algorithm Invest company (https://algoinvest.biz).

behavior. This article will conclude that the price prediction model built with the price cyclicality bands is a reliable method which can be applied for algorithmic trading in a wide range of capital markets.

II. The Price Prediction Model

In order to answer the questions "when to buy?" and "when to sell?" on financial markets to make profits, a mathematical model will be developed in this chapter. Based only on the price action, this model will permit to set up the buy and sell orders by specialized software in order to automate the trading process. The concepts used in this chapter are not new. The Price Cyclicality (PCY) function was presented for the first time in [4] and represents a function describing the cyclical behavior of the price movement and the intervals when the price is approaching to change its direction. The PCY function can be used in order to set limit conditions to entry and to exit the markets. The Price Prediction Line (PPL) was developed and explained in [5]. It represents an accurate trend line that describes and predicts the price evolution with good results. What is new in this paper is assembling all of these tools into a model building price bands that will give a clear indication for the entry and exit points, a reliable mathematical that can be adapted into automated software.

A. PCY Function

Being given a data price series of i intervals, the Price Cyclicality Function (*PCY*) is defined by:

$$PCY_i = \alpha\left(\Delta_i - PCY_{i-1}\right) + PCY_{i-1} \text{ with } PCY_0 = 0 \tag{1}$$

where

$$\Delta_i = \frac{\max_i - \xi_i}{\max_i - \min_i} \text{ with } \xi_i = MA_i - ma_i \tag{2}$$

and

$$\max_i = \max_{k=i,i-N}\left(MA_k - ma_k\right) \tag{3}$$

$$\min_i = \min_{k=i,i-N}\left(MA_k - ma_k\right) \tag{4}$$

in which *MA* and *ma* represent the moving averages [6] with two different periods. In (3) and (4), the term *N* is the period of the cyclicality function and represents the number of the time intervals taken into account to build the *PCY* function; α is a functional parameter that can be optimized for each financial market in order to obtain the best results.

More details about how *PCY* function was created, developed and optimized are presented in [4] together with the influence of the functional parameter α. It is also presented a study which reveals how the *PCY* function can be used in order to build a trading and limit conditions for financial markets. The *PCY* function can be applied in any timeframe and is a part of the trading model developed later.

B. Price Prediction Line

The Price Prediction Line (*PPL*) is a trend line obtained using a transformation function of the *PCY* function back into the price space using the formula:

$$PPL_i = PCY_i\left(P\max_i - P\min_i\right)/100 + P\min_i \tag{5}$$

where $P\max_i$ and $P\min_i$ represent the maximum and minimum price values in the current monotony interval of the PCY function given by (1). The PCY function is limited in the interval [0; 100]. Meanwhile, the PPL function is defined into the price interval and predicts the price evolution with good accuracy, as proved in [5]. For some financial markets with high price volatility, the PPL function defined by (5) needs an attenuation process in order to have a smooth evolution. For this, any known methods can be applied as smoothing with Spline line interpolations [7], polynomial or trigonometric interpolations [8] or just a simple, exponential or weighted moving averages [6] with a small period.

PCY and PPL functions for a daily price series of Frankfurt Stock Exchange Deutscher Aktienindex DAX30 financial market [9] are presented in Fig. 1. The functions can be used in order to trade the markets. When PCY and PPL functions are starting to increase, a buy condition is met because the price will increase in the next time intervals. In [4] and [5] are presented different types of trading signals developed with this idea. As we can see in Fig. 1, the information about the entry point in the market can be easily found. For long term or investment trading systems, this point is a good trading opportunity. Even so, the exit point is not so obvious, once the market is bouncing up and down many times until the direction changes. For high-frequency trading systems, with small profit targets, to know the trend direction and the starting point of the trend is not enough. For this kind of algorithms, a take profit value and a stop loss level are required in order to have a complete trading strategy. For this purpose a methodology will be developed in the next sections, using the Parabolic Stop and Reverse (*PSAR*) indicator developed by Wilder [10].

C. Price Prediction Bands

In this section we will define the Price Prediction Bands (PPB) using for ascending trend (current price higher than PSAR):

$$StopLossBand_i = PSAR_i \tag{6}$$

$$TakeProfitOne_i = PPL_i + \left(PPL_i - PSAR_i\right) \tag{7}$$

$$TakeProfitTwo_i = PPL_i + \delta * \left(PPL_i - PSAR_i\right) \tag{8}$$

and for the descending part of the trend (current price lower than PSAR) we have similarly:

$$StopLossBand_i = PSAR_i \tag{9}$$

$$TakeProfitOne_i = PPL_i - (PSAR_i - PPL_i) \quad (10)$$

$$TakeProfitTwo_i = PPL_i - \delta * (PSAR_i - PPL_i) \quad (11)$$

D. How to Use the PPB

As we can see in Fig. 2, once the price touched the PSAR level above PPL, the new PASR level will be calculated under the PPL and the current price. This is the moment when the uptrend begins. There are many trading strategies using this point as an entry point, but the PSAR methodology [10] is limited when it is about the exit point. The usual exit point is located when the price level is equal with PSAR when a new downtrend is starting. This idea is not productive at all, because many times the profit level is too low. It was found that exit the trade will produce a higher profit when the price level is equal with PPB, once the profit is taken near a local maximum point.

The first take profit level predicted by the PPB is calculated in direct relation with the distance between PPL and PSAR. For each time interval the values for PPL and PSAR are different, consequently, the level for the take profit band (TPB) is different. Because the values of the PPL depend on the price evolution and PSAR is updated in strong correlating with the Average True Range (ATR) developed also by Wilder in [11], the take profit level given by PPB is in direct correlation with the price behavior and ATR evolution. The price band formed with the PSAR values levels will be noted here as SLB (stop-loss band). These values are used as stop-loss level in our model.

As we can see in Fig. 2, the distance between TPB and SLB is variable. We will call this distance as to be a safe trade range (STR), because under SLB a stop loss is touched and for values higher than TPB the price is too high for a new entry. There are intervals where the STR distance is increasing. In this case, we will say that we have a price expansion, these are the cases associated with a strong trend when the price is making new highs. If the STR is decreasing, we will say we have a price contraction. If we have an uptrend, this is the case when the trend is preparing to reverse or to slow down the price motion. Starting from the analysis of STR we will develop a trading strategy as it is presented in the next section. As it can be seen in Fig. 2, sometimes the price goes higher the TPB levels. For these cases, the second TPB is included using (8) and (11). We will note the second TPB with TP2B. In Fig. 2, the TP2B was plotted using the gold ratio (δ=1.618). This band is used in case of powerful trends to know where to close the trade with good profit, before the price turning point.

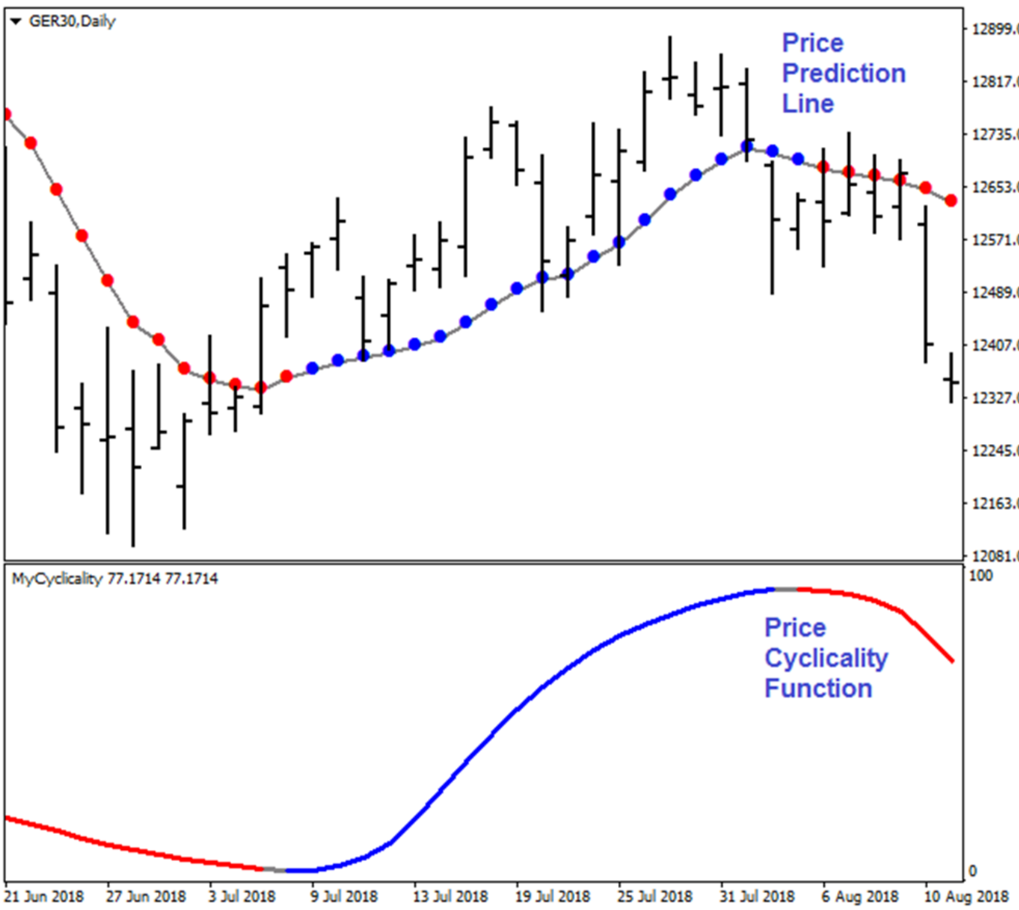

Fig. 1 PCY Function and PPL

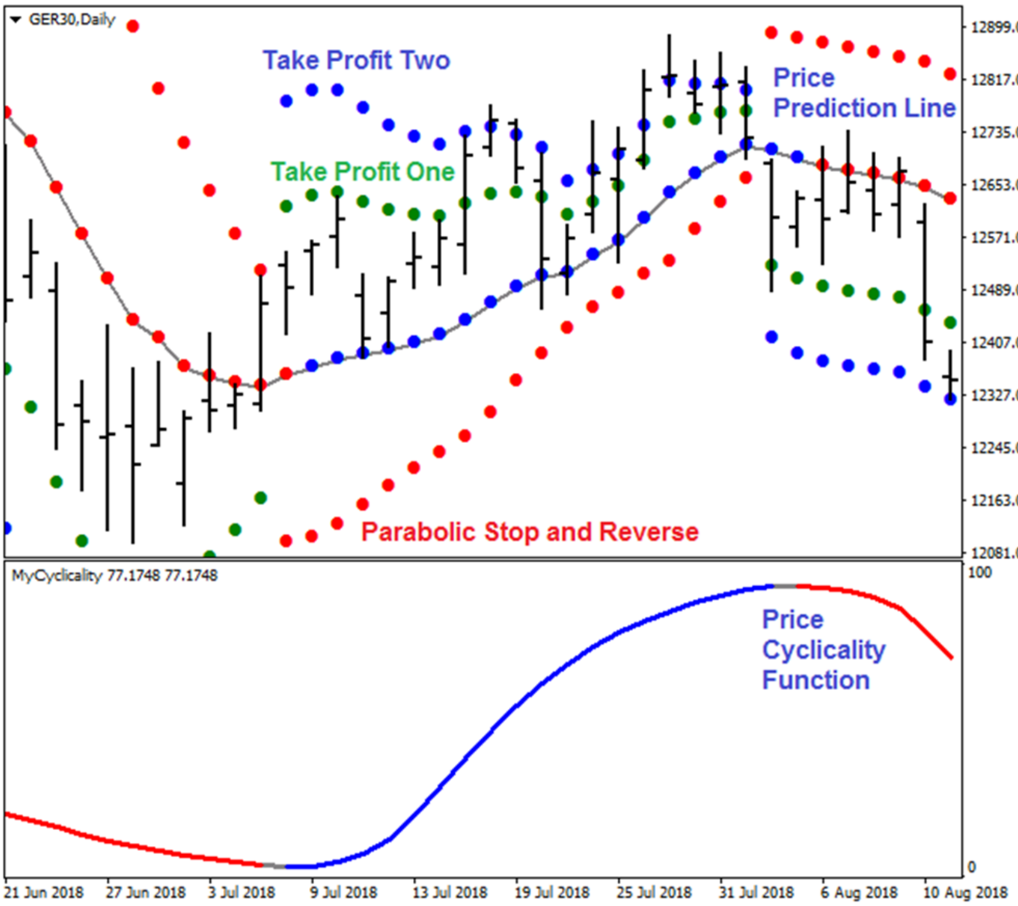

Fig. 2 PPB defined with PPL and PSAR

In the majority of the financial market, after the price touched the TPB or TP2B, a down movement is present before the next up movement. This is the interval when a new entry in the trade is possible in order to maximize the profit. Once the PPL is increasing and the PCY function is increasing too, the uptrend is still present and a new entry neat the PPL is a good opportunity. The PCY values give us good information about the proximity of the price turning point when the trend is reversing. In addition, large values of the STR involve good profit expectation meanwhile lower values for the STR can be a good indication to stay away from the market risk.

The PPB can be used for manual trading on daily timeframe or 4 hours (4H) timeframe. The information included in the PPB is good enough for short and medium time trades. Better results are expected using algorithmic trading and even high-frequency trading as we will see in the next section. An automated algorithm will use the PPB values in order to build automated trading signals. For high-frequency trading small profit target range will be used, the TPB and TP2B being used in order to limit the entry into a new position. For these cases, SLB is also used as a stop-loss level.

The PPB can be also used in order to find some cases when the price is oversold. Sometimes the price exceeds TPB and TP2B into downtrends with the STR is in contraction. In these cases, the price can be considered oversold and a buy trade can be a good opportunity until the price is bouncing again in the STR area. A case like this is plotted in Fig. 3. The oversold cases found with PPB will be a subject for additional trading signals included in the next section.

E. Trading Signals Based on the PPB

To automate the trading decisions we have to include the significance of the PPB levels into some Boolean variables called trading signals. These will be the core of the automated trading software presented in the next chapter. The first type of trading signal based on PPB is related to the point when the uptrend is starting. These signals for each time interval are given by:

$$\begin{aligned} Buy_i = &(PPL_i > PPL_{i-1}) \wedge (PPL_i > SLB_i) \wedge \\ &(Ask > SLB_i) \wedge (Ask < TPB_i - \theta) \wedge \\ &(PCY_i > PCY_{i-1}) \wedge (PCY_i < \rho) \end{aligned} \tag{12}$$

where *Ask* is the current ask price for the equity traded, θ is the minimal take profit level and ρ is the maximum PCY value as protection for the trend reversal. The functional parameters θ and ρ can be optimized for each financial market traded in order to maximize the profit. These trading signals can be used with good results for daily and four hours timeframes.

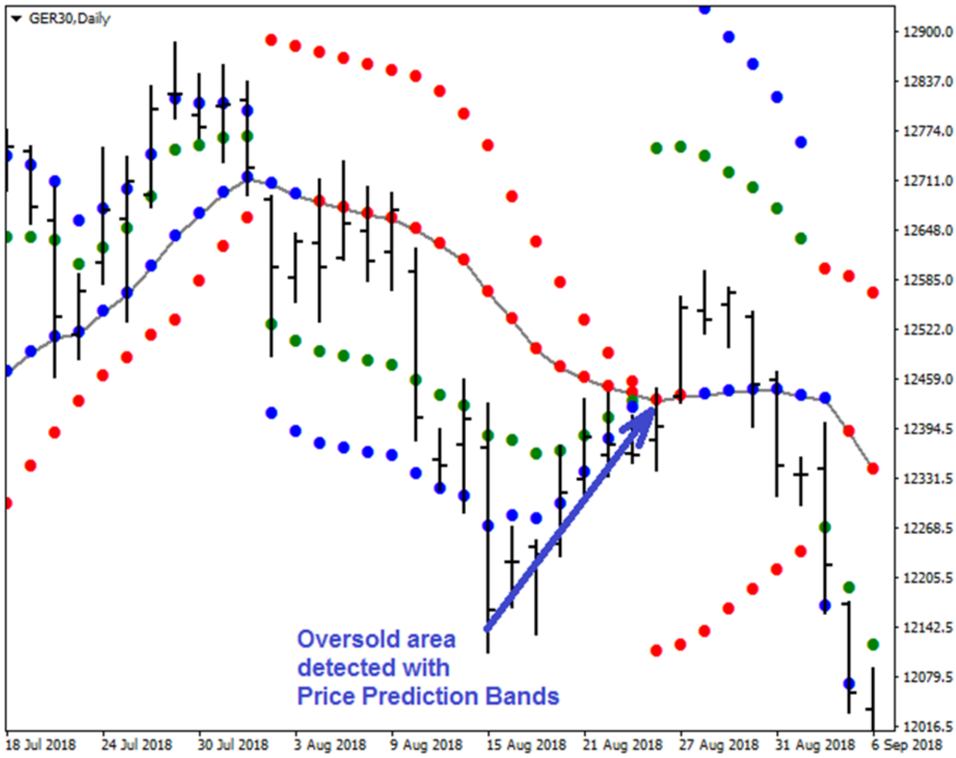

Fig. 3 Oversold price area detected with PPB

For the rest part of the uptrend, on the expansion periods of the STR, the trading signals can be given by:

$$\begin{aligned} Buy_i = {} & Expansion_i \wedge (PPL_i > SLB_i) \wedge \\ & (Ask > SLB_i) \wedge (Ask < TPB_i - \theta) \wedge \\ & (PCY_i > PCY_{i-1}) \wedge (PCY_i < \rho) \end{aligned} \tag{13}$$

where

$$Expansion_i = |TPB_i - SLB_i| > |TPB_{i-1} - SLB_{i-1}| \tag{14}$$

defines the price expansion intervals, where STR is increasing. These types of signals are used with good results for daily and four hours timeframes.

The signal given by (13) gives us good results for the price expansion intervals as we will see in the last chapter. In the rest of the intervals, when we have not an expansion for the STR, there are also good cases for trading opportunities. We have found that cases are filtered by an additional condition imposed for the TPB. When the TPB is increasing, even STR is decreasing, good trading opportunities can be found. All of these are given by the trading signals assembled with the formula:

$$\begin{aligned} Buy_i = {} & (TPB_i > TPB_{i-1}) \wedge (PPL_i > SLB_i) \wedge \\ & (Ask > SLB_i) \wedge (Ask < TPB_i - \theta) \wedge \\ & (PCY_i > PCY_{i-1}) \wedge (PCY_i < \rho) \end{aligned} \tag{15}$$

It was found that the trading signals given by (15) give us good trading results for the daily and four hours timeframes. To automate the cases for the oversold price intervals detected with PPB, the trading signals can be given by:

$$\begin{aligned} Buy_i = {} & (!Expansion_i) \wedge (PPL_i > TPB_i) \wedge \\ & \wedge (Ask < TPB_i - \theta) \wedge (PCY_i < \mu) \end{aligned} \tag{16}$$

where μ is a minimal value for the PCY function for which long trade is accepted in the oversold area. This functional parameter can be optimized for each financial market in order to improve the results. These signals give us good results for four hours timeframe. For the daily timeframe, the trading signals (16) need to be filtered with an additional condition in order to set only that trades in presence of a strong trend. For this purpose, a simple limit condition for the ATR values is strong enough. In this section, we presented the buy side signals developed with PPB. These are the most used trading signals for the majority of the markets. For those markets where sell trades can be considered, the sell trading signals can be assembled similarly.

III. Informatics for Automatic Trading

The place of the automated trading software in the business intelligence system of a modern investment company is well defined in [3]. “An automated trading system is a software which is receiving the real-time and historical price data of an equity, generates the signals for buying and selling of the equity based on well-determined algorithms, sets the volume of trading based on the capital liquidity and the capital a defined risk level, builds the trading orders and send them to the brokerage account without any human intervention" [3]. A logical scheme for automated trading software is presented in Fig. 4.

There are three data inputs for automated trading software. First is the low-latency real-time price data from the stock exchange. The second is the historical price data coming from a data warehouse. These two data fluxes are managed by two modules for real-time and historical data-mining processes. A low latency data management module will assure the speed for the data processing. The price data series are stored in memory and set up to be ready for the mathematical model.

The trading algorithms use low-latency price data and build trading signals. The third data flux includes real-time capital and liquidity data from the brokerage account. These data are the core of the risk management module. In this module, depending on the liquidity and the risk level established, the volume for the trading orders is set up. A reliable capital and risk management method is presented in [11]. With the signals and trading volume, the orders can be assembled and automatically sent to the brokerage account.

In this paper, details about the integration of the trading model developed will be presented. All technical aspects regarding data acquisition are already solved. There are many trading platforms available that integrate all of these features. One of them is Meta Trader 4 [12] which permits algorithmic trading using a Meta Quotes programming language [13]. This language will be used in the next sections in order to exemplify the codes for different procedures.

A. Integration of PCY Function

The PCY function is the core of the presented trading model. The formulas presented in Section II *A* must be computed with low time consumption for a prompt response, to permit assemblage and sending the trading orders as fast as possible. In Fig. 5 is presented a code sample to compute the PCY function in real time.

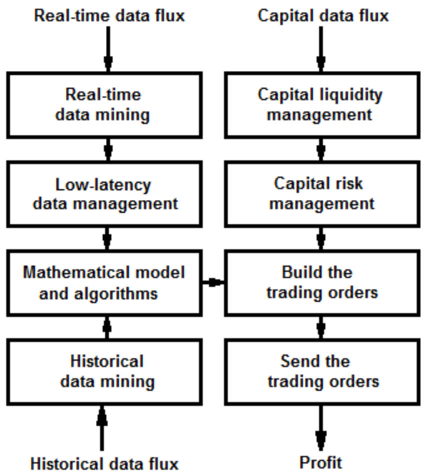

Fig. 4 Data logical scheme of automated trading system

```
//Price cyclicality indicator
input int MAShortPeriod;
input int MALongPeriod;
input int MACyclePeriod;
input double AlphaPower;
input ENUM_MA_METHOD MAMethod=MODE_SMA;
#property indicator_separate_window
#property indicator_minimum 0
#property indicator_maximum 100
#property indicator_buffers 3
#property indicator_color1 Gray
#property indicator_color2 Blue
#property indicator_color3 Red
double Cyclicality[], UP[], DOWN[], DELTA[], PriceCyclePeriod=MACyclePeriod*2;
int init(){IndicatorBuffers(4); SetIndexBuffer(0, Cyclicality);
           SetIndexBuffer(1, UP); SetIndexBuffer(2, DOWN); SetIndexBuffer(3, DELTA);
           SetIndexStyle(0,DRAW_LINE,STYLE_SOLID,3,indicator_color1);
           SetIndexStyle(1,DRAW_LINE,STYLE_SOLID,3,indicator_color2);
           SetIndexStyle(2,DRAW_LINE,STYLE_SOLID,3,indicator_color3); return(0);}
int start(){if(Bars<=MALongPeriod) return(0);
    bool PriceCycleStart=false, MACycleStart=false;
    int CountedBars=IndicatorCounted(), i, j, nPriceCycle=1, nMACycle=1;
    double CurrentDelta, Maxim, Minim, MAShort, MALong, MINMAX[MALongPeriod];
    if(CountedBars<MALongPeriod+MAShortPeriod)
      {for(j=1;j<=MALongPeriod;j++){Cyclicality[Bars-j]=0.0; DELTA[Bars-j]=0.0;}}
    i=Bars-MALongPeriod-1; PriceCycleStart=false; MACycleStart=false;
    while(i>=0){MAShort=iMA(Symbol(),0,MAShortPeriod,0,MAMethod,PRICE_TYPICAL,i);
                MALong=iMA(Symbol(),0,MALongPeriod,0,MAMethod,PRICE_TYPICAL,i);
                CurrentDelta=MAShort-MALong; MINMAX[nPriceCycle]=CurrentDelta;
                if(nPriceCycle>=PriceCyclePeriod){nPriceCycle=1; PriceCycleStart=true;}
                else{nPriceCycle=nPriceCycle+1;}
                if(PriceCycleStart){for(j=1; j<=PriceCyclePeriod; j++)
                  {if(j==1){Maxim=MINMAX[j];} else{if(Maxim>MINMAX[j]) Maxim=MINMAX[j];}
                   if(j==1){Minim=MINMAX[j];} else{if(Minim<MINMAX[j]) Minim=MINMAX[j];}}
                   DELTA[i]=((Maxim-CurrentDelta)/(Maxim-Minim))*100; nMACycle=nMACycle+1;
                   if(nMACycle>=MACyclePeriod){nMACycle=1; MACycleStart=true;}} else {DELTA[i]=0;}
                if(MACycleStart){Cyclicality[i]=AlphaPower*(DELTA[i]-Cyclicality[i+1])+Cyclicality[i+1];} i--;}
    int k=Bars-CountedBars; if(nMACycle==0) k-=1;
    for(j=k;j>=0;j--){if(Cyclicality[j]>=Cyclicality[j-1]){DOWN[j]=Cyclicality[j]; UP[j]=EMPTY_VALUE;}
                                                 else{UP[j]=Cyclicality[j]; DOWN[j]=EMPTY_VALUE;}}
    return(0);}
//Price cyclicality indicator
```

Fig. 5 Code sample to compute the PCY Function [4]

B. Integration of PPL

Once the PCY functions values are computed, the PPL values can be given by a procedure like in Fig. 6.

With PCY and PPL values ready, the trading signal variables can be easily computed. When the signal is confirmed, a trading order is assembled and sent instantly by the trading platform to the broker.

```
int start()
  {for(int i=Bars-MyPeriod;i>=0;i--)
     {Cycle[i]=iCustom(NULL,0,"MyCyclicality",MyPeriod,0,i);
      if(Cycle[i]>=Cycle[i+1]){Pmax=iHigh(NULL,0,i);}
      if(Cycle[i]< Cycle[i+1]){Pmin=iLow(NULL,0,i);}
      Buffer[i]=Cycle[i]*(Pmax-Pmin)/100+Pmin;
      Prediction[i]=iMAOnArray(Buffer,0,MyPeriod/2,0,MODE_EMA,i);
      if(Cycle[i]>=Cycle[i+1]){UpBuffer[i]=Prediction[i]; DnBuffer[i]=0; Signal[i]=1;}
                              else{DnBuffer[i]=Prediction[i]; UpBuffer[i]=0; Signal[i]=0;}}
   return(0);}
```

Fig. 6 Code sample to compute the PPL

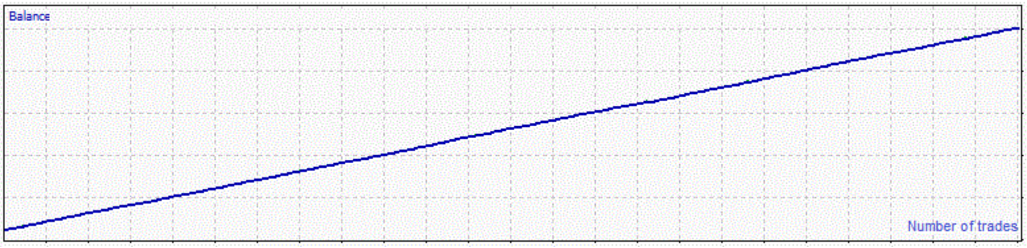

Fig. 7 Capital evolution due to the trades made by PPB signals

IV. Trading Results

In this chapter are displayed results obtained with all trading signal types presented above.

A. Trading Results Using PPB Methodology

The results presented below were obtained using TheDaxTrader [14], an automated trading system that uses the PPB trading signals in order to generate buy side trades for DAX30 [9].

The results presented in Table I were obtained with a high-frequency trading algorithm applied for DAX30 the period 01.06.2015 – 30.09.2018 using a fixed target of 10 points for each trade. The DAX30 index market was traded as a contract for difference (CFD) with a spread of 1 point. The exposed capital involved and the risk management for the high-frequency trading procedures was made using the “Global Slot Loss Method” presented in [11].

The PPB trading signals were built for daily and four hours timeframe interval. An additional condition was imposed regarding the hourly intervals of the executed trades between 8:00 and 20:00 coordinated universal time (UTC) in order to ensure the liquidity on the market. In Table I are presented the trading results obtained for the PCY limit $\rho = 99.95$, for the minimal take profit distance $\theta=10$ and for $\alpha=0.33$, the functional parameter of the PCY function. For the signals given by (16) the value $\mu = 1$ was used for the maximal level of the PCY into the oversold intervals. The moving averages used to build the PCY function were computed for 20 and 50-time intervals. The signals were computed by the trading software one time per minute.

All trading signals assembled with the PPB values generate a significant number of trades and good values for the risk and reward ratio (RRR). For the signals (12), (13) and (15) traded together; the RRR obtained is 1:8.22, a very good value compared with other signals as we will see in the next section. The lowest capital exposure is obtained for these signals made in direction of the main trend. For all PPB signals traded together, the RRR is 1:6.53, also a good value.

TABLE I
Trading Results Obtained with PPB Signals

Trading signals and timeframe	Number of trades	Profit	Draw-down	RRR
(12) 4 Hours	53	7,410	5,350	1:1.39
(12) Daily	117	16,224	5,353	1:3.03
(13) 4 Hours	72	10,292	5,123	1:2.01
(13) Daily	119	16,069	5,688	1:2.83
(15) 4 Hours	136	19,570	5,131	1:3.81
(15) Daily	180	25,701	5,687	1:4.52
All above together	330	46,970	5,716	1:8.22
(16) 4 Hours	179	25,274	9,337	1:2.71
All signals together	430	60,969	9,337	1:6.53

The longest trade period for these signals obtained in the study above was 218 hours. The Quality Trading Coefficient (QTC) [16] obtained was between 0.52 and 0.87. All these values indicate that PPB trading methodology is a reliable one. Even the oversold PPB trading signals used a double capital exposure, the returns from this type of trades is significant. The capital evolution in the time interval for all signals assembled together is presented in Fig. 7.

B. Comparative Trading Results

In order to have a clear image of the PPB trading methodology, in this section will be presented comparative trading results made in the same market conditions with different and known trading methodologies. In order to compare the same type of trading methodologies, we will compare the results obtained with (12), (13) and (15) with results made with other trading methodologies in the direction of the main trend. To obtain the results in Tables II and III, TheDaxTrader [14] automated trading system was used for DAX30 Index [9] for the period 01.06.2015 – 30.09.2018 using a high-frequency trading methodology with a fixed profit target of 10 points for all signals presented.

TABLE II
COMPARATIVE TRADING RESULTS

Trading signals and timeframe	Number of trades	Profit	Draw-down	RRR
Perfect Order [16]	107	20,107	8,712	1:2.31
Fischer Signals [17]	177	47,212	6,321	1:7.46
Turtle Signals [18]	162	24,160	3,693	1:6.54
PPB (12)+(13)+(16)	330	46,970	5,716	1:8.22

As we can see in Table II, the results made with PPB trading methodology have the highest RRR value. The PPN trading signals made almost double trades than other trading strategies in the same trading conditions. These results are an additional confirmation that the PPB trading methodology is a reliable one. The results made with (16) will be compared with other results made also with an oversold price trading methodology in Table III.

TABLE III
COMPARATIVE TRADING RESULTS OBTAINED WITH PRICE OVERSOLD METHODS

Trading signals and timeframe	Number of trades	Profit	Draw-down	RRR
RSI Oversold [19]	76	18,243	5,901	1:3.09
PPB (16)	179	25,274	9,337	1:2.71

As we can see, the PPB trading signals made a significant number of trades with a comfortable RRR value. The PPB results are perfect comparable with the RSI oversold trading method presented in [19]. Both methods included in Table III trade the cases when the rice is oversold but they rarely intersect. Using both methods into a trading system will generate additives profits with the same capital exposure.

V. CONCLUSIONS

The PPB presented in this paper can be used to develop a reliable trading methodology. Based only on the price action, the trading method presented here can be easily applied for algorithmic trading and high-frequency trading in automated trading systems.

The core of this trading model is the PCY function. It is a mathematical transformation of the price into a subspace defined in the [0; 100] interval. Based on the minimal and maximal price values on a time interval, the PCY function has an asymptotic behavior and can be used in order to impose limit conditions in order to avoid initiating trades in the interval when the main trend is preparing to reverse.

The PPL [5] was considered as the core for this trading model. It is a reversed transformation of the PCY Function into the price space. This trend line is considered as the main price for the prediction bands. In order to find the main trend, the method uses the *PSAR* [10] function which defines the current SLB. To have a complete trading model, TPBs are defined using (7), (8), (10) and (11). All of these together define the PPB.

The methodology presented in this paper can be easily used for manual trading and investment. The PPB levels can be followed in order to set up the trades. The best results are obtained with daily and four hours timeframes. This methodology was tested with good results for the next financial markets: Deutscher Aktienindex (DAX30), Dow Jones Industrial Average (DJIA30), Financial Times London Stock Exchange (FTSE100), Cotation Assistée en Continue Paris (CAC40), Swiss Stock Exchange Market Index (SMI20), Australian Securities Exchange Sydney Index (ASX200), Tokyo stock Exchange Nikkei Index (Nikkei225), NASDAQ100 Index, Standard & Poor's Index (S&P500) and Small Capitalization US Index (Russell2000). Also with good and stable results, the PPB methodology presented in this paper was applied for Gold and Bent Crude Oil financial markets. For the currency markets, the method can be also applied using additional conditions regarding the price volatility level and the power of the trend. The cases when the market is not in a major trend must be avoided.

Being exclusively a mathematical model based on the price action, the PPB method can be adapted for algorithmic trading and can be easily included in an automated trading system. Sample codes about how the PCY Function, the PPL and the PPB can be automated are also included in this paper. The simplicity of this method and the reduced number of functional parameters made this methodology to be one of the easiest integrated ones into a trading system. All parameters for the PCY function can be set up and used with the same valued for all financial markets. The parameters for the trading signals presented can be optimized for each financial market but the values do not differ in time; once optimized for a long period of time, they can be used with good results for the next period.

Looking at the results presented in the last chapter, the significant number of the trades set up by the presented method and the good values for the risk and reward ratio recommend the PPB as to be a reliable trading methodology.

As it was presented, the method shows when a new trend begins. For these particular moments, trading signals can be built with (12). The PPB also reveals the time intervals when the price is in an expansion. For these cases when the STR is increasing, good trading opportunities can be found using (13). For the intervals when the price is contracting, sustained trading signals are also built with (15). All of these signals traded together obtained a risk to reward ratio value equal with 1:8.22 for the study case took as an example. This value is a very good one for a single strategy computed with algorithmic trading into an automated trading system.

As it was found in this paper, using the price level related to the PPB values, good opportunities for oversold buy trades can be found using (16). These additional trading signals offer us a significant number of trades even if the risk and reward ratio is higher. The profit made by these types of signals is a significant one and the methodology is preferred by many investors. The signals build with the PPB can be also included in automated investment systems with very good efficiency.

Taking in consideration the results presented, the simplicity of the method, all the advantages regarding the trend detection, stop loss and take profit levels, detection of the price expansion and contraction intervals, defining the STR in different timeframe make the PPB be a reliable and sustained trading methodology.

The reduced number of the functional parameters and the simple integration into any trading software recommend the method presented to be considered for any automated trading and investment system.

The PPB trading methodology improves the spectrum of the trading strategies and can also be considered as a data-mining filter in addition to any other trading methodology in order to improve the trading efficiency.

References

[1] R.D. Zota, L. Ciovica, Designing software solutions using business processes, 7th International Conference on Globalization and Higher Education in Economics and Business Administration, GEBA 2013, DOI: 10.1016/S2212-5671(15)00125-2

[2] V.D. Păvăloaia, Methodological Approaches to Computer Modeling Possibilities in Financial Analysis, Scientific Annals of Economics and Business, January 2012, ISSN: 2501-3165, DOI: 10.2478/v10316-012-0026-5, Available at: https://www.researchgate.net/publication/267202159

[3] C. Păuna, Automated Trading Software. Design and Integration in Business Intelligence Systems. Bucharest, Romania: Database Systems Journal, vol. IX/2018, ISSN: 2069-3230. Academy of Economic Studies. Available at: http://dbjournal.ro/archive /29/29_3.pdf

[4] C. Păuna, I. Lungu, Price Cyclicality Model for Financial Markets. Reliable Limit Conditions for Algorithmic Trading, Journal of Studies and Researches of Economic Calculation and Economic Cybernetics, 4/2018, ISSN: 0585-7511. Academy of Economic Studies DOI: 10.24818/18423264/ 52.4.18.10 Available at: http://ecocyb.ase.ro

[5] C. Păuna, A Price Prediction Model for Algorithmic Trading, under final review at Romanian Journal for Information Science and Technology, ISSN: 1453-8245. Romanian Academy.

[6] Cox, D.R. Sir, Prediction by Exponentially Weighted Moving Averages and Related Methods, 1961, Journal of the royal Statistical Society, Series B, Vol. 23, No. 2, pp. 414-422.

[7] C. Reinsch, Smoothing by Spline functions, Numerische Mathematik, Volume 10, Issue 3, ISSN 0945-3245, 1967, DOI https://doi.org/10.1007/BF02162161. pp. 177-183.

[8] C. Berbente, S. Mitran, S. Zancu, Metode Numerice. Bucharest, Romania: Editura Tehnica, 1997, ISBN 973-31-1135-X, pp. 15-20.

[9] Börse, Frankfurt, Frankfurt Stock Exchange Deutscher Aktienindex DAX30 Components, 2018. Available on: http://www.boerse-frankfurt.de/index/dax

[10] J.W. Wilder, Jr., New Concepts in Technical Trading Systems. Greensboro, NC Trend Research, 1978. ISBN 978-0-89459-027-6.

[11] C. Păuna, Capital and risk management for automated trading systems, Iași, Romania: Proceedings of the 17th International Conference on Information in Economy, May 2018, Alexandru Ioan Cuza Academy. Available at: https://pauna.biz/ideas

[12] Meta Trader 4, 2018. Available at: https://www.metatrader4.com

[13] Meta Quotes Language 4, 2018 Available at: https://www.metatrader4.com/en/automated-trading/mql4-programming

[14] C. Păuna, TheDaxTrader. Automated trading system, 2010. Online software presentation. Available at: https://pauna.biz/ thedaxtrader

[15] C. Păuna, The Quality Trading Coefficient. General Formula to Qualify a Trade and a Trading Methodology, Bucharest, Romania: Economic Informatics Journal, vol. 22, no.3/2018. ISSN: 1453-1305, Academy of Economic Studies. DOI: 10.12948/issn14531305/ 22.3.2018.09. Available at: http://revistaie. ase.ro/content/87/09%20-%20pauna.pdf

[16] K. Lien, Day Trading & Swing Trading the Currency Market – Technical and Fundamental strategies to Profit from Market Moves, John Wiley & Sons, 2009, pp. 138-140

[17] C. Păuna, Reliable Signals Based on Fisher Transform for Algorithmic Trading, Timișoara, Romania: Timisoara Journal of Economic and Business, Volume 11, Issue 1/2018. ISSN: 2286-0991 DOI: 10.2478/tjeb-2018-0006 Available at: https://www.tjeb.ro

[18] C. Păuna, A Different Approach for the Turtle Strategy in Algorithmic Trading, under final review for INEKA 2019 at University of Verona, Italy. Available at: https://pauna.biz/ideas

[19] L. Connors, C. Alvarez, Short Term Trading Strategies That Work – A Quantified Guide to Trading Stocks and ETFs, TradingMarkets Publishing Group, 2009. pp. 53-74