Kent Academic Repository

Full text document (pdf)

Citation for published version

Adegboye, Adesola and Kampouridis, Michael and Johnson, Colin G. (2018) Regression genetic programming for estimating trend end in foreign exchange market. In: IEEE Symposium Series on Computational Intelligence, 27 Nov - 01 Dec 2017, Hawaii, USA.

DOI

https://doi.org/10.1109/SSCI.2017.8280833

Link to record in KAR

http://kar.kent.ac.uk/63368/

Document Version

Author's Accepted Manuscript

Copyright & reuse
Content in the Kent Academic Repository is made available for research purposes. Unless otherwise stated all content is protected by copyright and in the absence of an open licence (eg Creative Commons), permissions for further reuse of content should be sought from the publisher, author or other copyright holder.

Versions of research
The version in the Kent Academic Repository may differ from the final published version.
Users are advised to check **http://kar.kent.ac.uk** for the status of the paper. **Users should always cite the published version of record.**

Enquiries
For any further enquiries regarding the licence status of this document, please contact:
researchsupport@kent.ac.uk

If you believe this document infringes copyright then please contact the KAR admin team with the take-down information provided at **http://kar.kent.ac.uk/contact.html**

KAR Kent Academic Repository

University of Kent

Regression genetic programming for estimating trend end in foreign exchange market

Adesola Adegboye, Michael Kampouridis and Colin G. Johnson
School of Computing
University of Kent, Medway, uk
Email: {atna3, M.Kampouridis, C.G.Johnson}@kent.ac.uk

***Abstract*—Most forecasting algorithms use a physical time scale for studying price movement in financial markets, making the flow of physical time discontinuous. The use of a physical time scale can make companies oblivious to significant activities in the market, which poses a risk. Directional changes is a different and newer approach, which uses an event-based time scale. This approach summarises data into alternating trends called upward directional change and downward directional change. Each of these trends are further dismembered into directional change (DC) event and overshoot (OS) event. We present a genetic programming (GP) algorithm that evolves equations that express linear and non-linear relationships between the length of DC and OS events in a given dataset. This allows us to have an expectation when a trend will reverse, which can lead to increased profitability. This novel trend reversal estimation approach is then used as part of a DC-based trading strategy. We aim to appraise whether the new knowledge can lead to greater excess return. We assess the efficiency of the modified trading strategy on 250 different datasets from five different currency pairs, consisting of intraday data from the foreign exchange (Forex) spot market. Results show that our algorithm is able to return profitable trading strategies and statistically outperform state-of-the-art financial trading strategies, such as technical analysis, buy and hold and other DC-based trading strategies.**

1. Introduction

The Foreign exchange (Forex) market is the biggest of all financial markets in terms of volume and liquidity [1], [2]. It is a decentralised market that is opened 24 hours a day, 7 days a week throughout the year across the globe. Some of the players in the market include corporations, central banks, retail traders and money remittance companies. Majority of these players buy and sell Forex in responses to economical, political and psychological news. Interaction between these players leads to highly frequent changes in price within a second. This frequent movement in price makes the nature of market data chaotic, noisy and non-stationary, hence making estimation of trend reversal challenging.

Traditionally, price movement is observed by sampling data on a physical time scale. Physical time scale data is generated by first deciding on a sampling interval, then successive data points at the decided interval are captured. Captured data points are used in creating price summary. Sampling data at constant intervals has the possibility of omitting important details between adjacent data points. This is because of the assumption that important market events occur constantly in time which is not always the case. For instance, assuming it is decided to sample price using daily closing price, the flash crash which occurred across US stock indexes on the 6th of May 2010 from 2:32 pm EDT till 3:08 pm EDT would be ignored as prices rebounded shortly afterwards.

Directional Changes (DC) is an alternative approach for capturing price movement. It is based on the idea of transforming physical time scaled data into intrinsic time scale data. In DC framework, the focal point is the measure of price change, while time is a varying factor. This is different from physical time data, where time is always fixed. This approach summarises data into alternating trends called upward directional change and downward directional change. Each of these trends are further dismembered into directional change (DC) event and overshoot (OS) event. DC event is an event that caused a change in price either upward or downward. OS event can be described as movement in price where impact of DC event is still felt even after the DC event has ended i.e after-effect. Therefore an upward DC event is followed by an upward OS event and downward DC event is followed by a downward OS event. The occurrence of DC event is known in hindsight whilst in the trend, therefore the length of DC event can be deduced easily. However, how long the OS event would last is unknown whilst in the trend. The challenge in trend reversal in DC perspective is the ability to determine the length of OS event.

Capturing price movement using DC is characterised by a scalable threshold (usually expressed in percentage) by which price needs to exceed in order to be considered significant. First, a trader according to his belief of what he or she considers a significant event in the market specifies a threshold. Then, the market is summarised by traversing physical time price curve capturing alternating changes in price equal or greater than the specified threshold. By focusing on the important events in the market, DC obfuscates noise, enabling traders focus on important price events.

There have been empirical studies to investigate regularities in DC summarised data [3]. One such regularity is the existence of a linear relationship between the length of a DC event and its corresponding OS event. It was observed that, on average, if DC event takes t amount of physical time, its corresponding overshoot (OS) will take twice the amount ($2t$). This regularity provides traders with greater understanding of price evolution, which can be leveraged for trend[1] reversal estimation. This is because if one can anticipate how long an OS event will last, it also means that they will know when a directional change takes place, and a new event of the opposite direction starts. Thus, with this OS length regularity, traders are able to devise new strategies to maximise return on their investments. Nevertheless, a drawback of the above regularity is that it presents a simple, linear DC-OS relationship, which was based on averaged observations.

The above has motivated us to look for equations that uncover richer relationships that are not covered by the linear relationship discovered in [3]. To this end, we develop a tailored GP that finds equations to describe the DC-OS relationship. Our goal is twofold: (i) demonstrate that GP can be used to anticipate trend reversal in Forex data, and (ii) demonstrate that the GP derived equations can improve the trading profitability. To this end, we use the GP derived equations as part of a DC-based trading strategy [4]. We compare our results to other DC-based trading strategies and also to two popular physical-time based trading strategies, namely technical analysis and buy and hold.

The rest of this paper is organised as follows: Section 2 presents background information on the DC approach. Section 3 presents the methodology of this paper, and Section 4 presents our experimental setup. In addition, Section 5 presents and discusses our results. Finally, Section 6 concludes the paper and discusses directions for future work.

2. DC Background

2.1. Overview

A directional change (DC) event is characterised by a displacement in the price of a given security (in our case a currency pair) that is considered to be significant. The minimum displacement size is defined by a threshold value, which is in advance decided by a trader. DC event can either be an upturn or a downturn event. After the confirmation of a DC event, an overshoot (OS) event follows. This OS event finishes once an opposite DC event takes place.

The combination of a downturn DC event and a downward OS event represents a downward run and, the combination of an upturn DC event and an upturn OS event represents an upward run. Combination of series of alternating downward runs and upward runs form DC price curve. Figure 1 presents an example of how a physical-time price curve is transformed to the so-called intrinsic time

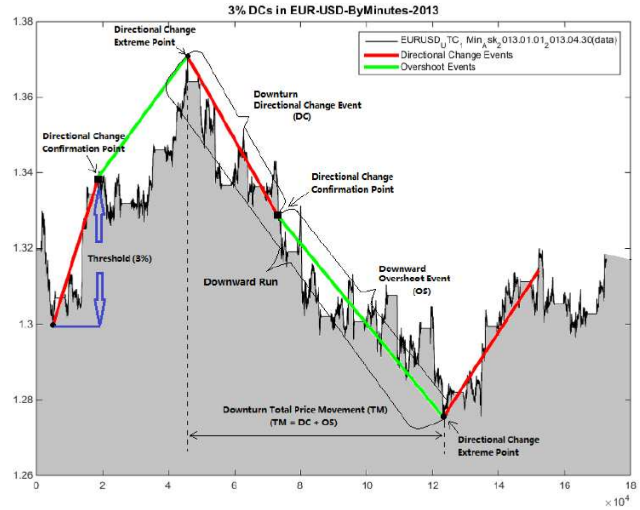


Figure 1: Projection of a DC events defined by a threshold $\theta = 3.0\%$. Source: [5]

[6] and dissected into DC and OS events. Furthermore, we present Algorithm 1, a high-level pseudocode for transforming physical-time price curve to a DC price curve.

In comparison to technical analysis, directional changes is a relatively new way of sampling data. Most DC works focused on either finding additional DC indicators or developing trading strategies with existing indicators. We now present a literature review, of works in DC.

2.2. Review of DC literature

Guillaume et al. [6] carried out the first work that used a DC dataset as an alternative to a physical time dataset. Glattfelder et al. [3] introduced 12 new scaling laws. These laws exhibited quantifiable relationships between the size of price evolution and market transactions that occurred in a given period. These laws were established after empirical studies using Forex data of 13 major currencies. One of the most interesting regularities that was discovered, was the observation that if on average a DC takes t amount of physical time to complete, the OS event will take an amount of $2t$. Aloud et al. [7] extended the work of Glattfelder and introduced four additional scaling laws, these new scaling laws were successfully applied to investigate the impact of different strategies on trading activities in high-frequency Forex market. The catalogue of DC indicators was further extended by Aloud [8] who introduced 5 additional laws to the ones already discovered in [3], [7]. Bisig et al. [9] reported the Scale of Market Quakes (SMQ). SMQ is a way of sizing the impact of economic or political development and other major breaking news on price movement within the Forex market. Their goal was to set the foundation for creating a metric that can be used to measure price change vis-a-vis major world events.

Aloud and Fasli [10] developed agents that model trader's behaviour in Forex market. Their work focused

1. In the context of directional changes, trend can be described as sum of DC event length and OS event length. At the end of a trend, another one begins in the opposite direction. Just before the start of the new trend, is the trend reversal point of the trend that just ended.

Algorithm 1 Pseudocode for generating directional changes events given threshold Δx_{dc}. (source: [3])

Require: Initialise variables (event is Upturn event, $p^h = p^l$ = p(t_0), $\Delta x_{dc}(Fixed) \geq 0, t_0^{dc} = t_1^{dc} = t_0^{os} = t_1^{os} == t_0$)

if event is Upturn Event **then**
    if $p(t) \leq p^h \times (1 - \Delta x_{dc})$ **then**
        $event \leftarrow Downturn\ Event$
        $P^l \leftarrow p(t)$ //Price at end time for a Downturn Event
        $t_1^{dc} \leftarrow t$ //End time for a Downturn Event
        $t_0^{os} \leftarrow t{+}1$ //Start time for a Downward Overshoot Event
    else
        if $p^h < p(t)$ **then**
            $p^h \leftarrow p(t)$ //Price at start of Downturn event
            $t_0^{dc} \leftarrow t$ //Start time for Downturn event
            $t_1^{os} \leftarrow t{-}1$ //End time for a Upturn Overshoot Event
        end if
    end if
else
    if $p(t) \leq p^l \times (1 + \Delta x_{dc})$ **then**
        $event \leftarrow Upturn\ Event$
        $P^h \leftarrow p(t)$ //Price at end time for upturn event
        $t_1^{dc} \leftarrow t$ //End time for a Upturn Event
        $t_0^{os} \leftarrow t + 1$ //Start time for a Upturn Overshoot Event
    else
        if $p^l > p(t)$ **then**
            $p^l \leftarrow p(t)$ //Price at start time for upturn event
            $t_0^{dc} \leftarrow t$ //Start time for a Upturn Event
            $t_1^{os} \leftarrow t - 1$ //End time for a Downturn Overshoot Event
        end if
    end if
end if

on establishing stylised facts regarding how traders react and adapt to changes in Forex market. Their agents used strategies known as ZI-DC0 developed by combining DC approach with trend following and contrary trading technical indicators. Aloud in [11] proposed a new trading strategy called ZI-DC1 as an improvement to the study in [10], Comparison results between ZI-DC0 and ZI-DC1 showed that ZI-DC1 was more profitable. Aloud in [12] introduced an automated trading strategy (DCT2) that can perceive changes in market conditions and adapt dynamically to remain profitable.

Kablan and Ng [13] developed a neuro-fuzzy logic based trading strategy that captures volatility using DCs within pre-specified thresholds. The system predicted the future price of an asset based on the current price and the immediate past three consecutive observations in the market. Their model outclassed the physical-time scale trading strategies they compared with, in terms of profitable returns. Bakhach et al. [14] transformed DC forecasting task into a classification problem. Their goal was to establish the estimative power in directional changes approach. To do this end, they created three new directional changes indicators from technical indicator which they used in forecasting price value at OS extreme point. In addition, [15] was the first work to use a genetic programming algorithm to generate DC-based trading strategies. Results showed that the new algorithm had the potential to outperform its competitors.

One final work we would like to discuss is [4], as it directly influences our current paper. This work created DC-based trading strategies by taking into advantage the OS length regularity that was first observed and reported in [3]. However, [4] observed that the ratio between DC and OS events length is dependent on the distribution in the dataset under observation. In other words, they did not always observe that on average OS events would last twice the length of DC events. In an attempt to address the issue, they used tailored OS event lengths, which were calculated on the training datasets, for estimating trend reversals. They then tested the effectiveness of their methodology by using a multiple threshold DC trading strategy, which was optimised by a genetic algorithm. Their results showed that the algorithm was able to yield positive returns.

The above two works ([3], [4]) motivate us to continue towards this direction, looking for a better way to describe the relationship between DC and OS events length. As we have already explained, in this paper we will use a GP to uncover this relationship. In the next section, which presents our methodology, we first formalise the DC and OS relationship via a simple mathematical equation, and then we present the GP we used for creating new equation describing the above relationship.

3. Methodology

3.1. Linear and non-linear DC-OS Relationships

As we already mentioned above, there were two works that successfully described relationships between DC and OS event lengths: [3] and [4]. The former found that an OS event lasts on average twice the length of a DC event, and is shown in Equation 1. On the other hand, the latter work used tailored estimates, thus the ratio of OS and DC length varied per dataset. Nevertheless, this was still a linear equation, as we can see from Equation 2.

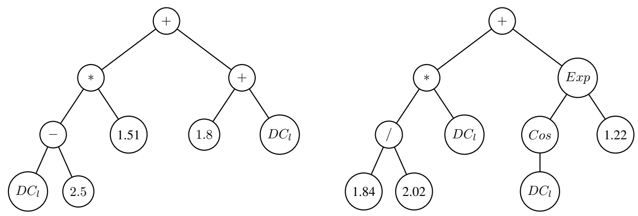
$$OS_l \approx 2 \times DC_l \tag{1}$$

$$OS_l = C \times DC_l; C > 0 \tag{2}$$

where OS_l is the length of an OS event, and DC_l is the length of a DC event. C is a constant, which can take any positive real value.

However, there might be cases that the above equation does not hold, and this was something that was confirmed from preliminary empirical studies. For example, there are cases where a DC event has a zero overshoot therefore there is no OS event. In this case, both of the above equations are unsuitable for estimating trend reversal. In addition, the relationship between DC and OS event length might not always follow a linear pattern. Using a simple linear equation will

Figure 2: Sample GP individual trees: internal nodes are represented by arithmetic functions. The leaf nodes are represent by numeric constants and the DC length, denoted as DC_l. Given a DC event length the tree estimates the corresponding OS length.



(a) Sample tree 1 (b) Sample tree 2

not take into account these dynamics. A limitation of this approach is that it could lead to hit-and-miss trend reversal estimation in some cases like the two we mentioned above.

In this study we are looking for both linear and non-linear relationships between DC and OS event lengths. Thus, instead of assuming a linear relationship, we propose re-writing Equations 1 and 2 in the more general form of Equation 3. As we can observe, Equation 3 suggests that there is indeed a relationship between OS and DC, but the underlying function is unknown. In order to discover this underlying function, we use a genetic programming algorithm, which is very popular with this type of symbolic regression problems, and is presented next.

$$OS_l = f(DC_l) \tag{3}$$

3.2. GP - Symbolic Regression

We use a tree-based GP [16], and our GP estimates OS length given a DC length. In this section we present the composition and evaluation of our GP.

First, a threshold is used in transforming physical time based dataset into DC based dataset. Then, the DC dataset is split into separate upward DC dataset and downward DC dataset. Finally we develop a GP that searches a solution space for an equation $f(DC)$ which maps DC event length to OS event length. The sum of the DC event length observed in the dataset and our estimated OS event length will give us the estimated point in time when trend reversal will happen. In our approach there will be a separate equation tailored for upward DC dataset and downward DC dataset respectively. For the equations to be valid, the expression must have at least one DC event length as its term.

3.2.1. Model representation. We represent our evolved GP individuals using tree structures. The inner nodes of our trees were composed of linear and non-linear functions. We utilised 2-arity functions: {addition, subtraction, division, multiplication, power} and 1-arity functions: {sine, cosine, power, log, exponential}. Division function was protected. Log, exp and power functions were also protected because depending on the input parameter to the functions, it was possible for trees to return NaN (not a number) and Inf (infinity). The terminal nodes consisted of an ephemeral random constant (ERC) and an external input which represented DC event length. We selected an ERC with a probability Pr, alternatively the DC event length is selected with probability $1 - Pr$. All our functions and terminals are presented in Table 1. To initialise our population we used ramped half-and-half technique. Figures 2a and 2b show sample trees our GP can construct. Figure 2a and 2b are trees which represent equations that calculates OS length as $((DC_l - 2.5) \times 1.51) + (1.8 + DC_l)$ and $((\frac{1.84}{2.02}) \times DC_l) + (e^{\cos(DC_l)/1.22})$ respectively and the DC_l in both equations represent the length of DC event.

TABLE 1: Configuration of the proposed GP algorithm

Configuraton	Value
Function set	addition, subtraction, division, multiplication,sine, cosine, power, log, exponential
Terminal set	input variable (i.e DC events length) and ephemeral random constant.
Genetic operation	subtree mutation and subtree crossover

3.2.2. Model evaluation. To evaluate our trees we measure error between actual OS length ($Actual_{OS_l}$) and OS length that the GP estimates ($Estimated_{OS_l}$). The estimation error $Error_{OS_l}$ was calculated using root mean square error, which we show in Equation 4.

$$Error_{OS_l} = \sqrt{\frac{\sum_{i=1}^{N}(Actual_{OS_l} - Estimated_{OS_l})^2}{n}} \tag{4}$$

During evolution, we penalised some trees constructs, namely trees that have only constants as terminal nodes, trees that estimate a negative value and trees that evaluate fitness to NaN or infinity. We performed tournament selection and selected parents based on fitness level. If there are more than one candidate trees, we evaluate tree size to break the tie and the tree with the shortest depth is selected.

3.2.3. Genetic operators. We used standard genetic operators to evolve most of the trees. The operators we used are subtree mutation and subtree crossover (see Table 1). Our evolution was controlled by a crossover ratio P_e. At P_e we select subtree crossover operator and at $1 - P_e$ we select subtree mutation operator. We introduced an elitism ratio that determines the number of best performing trees that should be carried forward from one generation to another. To control growth, we use hard limits on the depth of offspring programs generated. Maximum_depth is used for controlling mutation operation.

3.2.4. Problem specific GP parameters. We introduced a parameter called wrapper to replace wrong estimates with 0. This value was necessary because it was possible for GP to estimate OS length value as negative, NaN or infinite during out-of-sample testing. We chose the value of 0 because after empirical observation, we realised that there were cases where a DC event is directly followed by another DC event of the opposite direction, hence the OS length of the preceeding DC event was 0.

4. Experimental setup

In this section, we present the experimental setup to accomplish our two goals. As a reminder, our first goal is to demonstrate that the GP derived equations can be used to anticipate trend reversal; the second goal is to demonstrate that the GP derived equations can improve the trading profitability. As we have previously explained, having good estimates of trend reversal allows traders to take informed decisions and increase their margin for profit. To achieve our goals, our experiments are divided into two parts. In the first part, we will use the GP for symbolic regression in DC summaries to uncover hidden DC-OS relationships. We will then compare GP's regression error with previous works in this field, i.e. Equation 1 and Equation 2. In the second step of our experiments, we will embed the equations returned by our GP in an existing DC trading strategy, which was first presented in [4]. The goal here is to demonstrate that by uncovering better DC-OS relationships, we are able to better estimate trend reversals, which leads to improved trading results. We will compare the trading results' returns to other DC trading strategies, and also to popular financial benchmarks, such as technical analysis, and buy and hold.

The rest of this section is organised as follows: first, in Section 4.1 we present the data we use for our experiments. Then, in Section 4.2, we present the parameter tuning for our proposed GP, which as we have explained is part of the first step of our experiments. Lastly, Section 4.3 presents the experimental setup for the second step of the experiments, i.e. the actual trading results.

4.1. Data

We used 10-minute interval high frequency data of the following currency pairs: EUR/GBP (Euro and British Pound), EUR/USD (Euro and US dollar), EUR/JPY (Euro and Japanese Yen), GBP/CHF (British Pound and Swiss Franc), and GBP/USD (British Pound and US dollar) from June 2013 to May 2014. We considered each month in the period as a separate physical-time dataset. Datasets from the months of June 2013 and July 2013 were used for tuning OS length estimation and trading strategy algorithms. The rest of the dataset from the month of August 2013 to May 2014 were used for testing the tuned algorithm. We chose a ratio of 70:30 for training and testing sets.

4.2. GP parameter tuning

In total, we used 50 DC datasets for tuning. They were derived using thresholds 0.010%, 0.013%, 0.015%, 0.018% and 0.020% to generate DC price curves from Forex data of June and July 2013 across 5 currency pairs: EUR/GBP, EUR/USD, EUR/JPY, GBP/CHF and GBP/USD.

We tuned GP population size, number of generations, tournament size, crossover probability and maximum depth parameters using the I/F-Race package [17]. I/F-Race package is based on an iterated racing procedure, which is an extension of the Iterated F-race procedure. It implements racing methods for the selection of the best configuration for an optimisation algorithm by empirically selecting the most appropriate settings from a set of instances of an optimisation problem [18]. Table 2 presents our GP parameter configuration determined using I/F-Race.

TABLE 2: Regression GP experimental parameters for detecting DC-OS relationship, determined using I/F-Race.

Parameter	
Population	500
Generation	37
Tournament size	3
Crossover probability	0.98
Mutation probability	0.02
Maximum depth	3
Elitism	0.10

4.3. Trading algorithm experimental setup

As we have explained, in the second step of our experiments, we will embed the equations returned by our GP into an existing DC-based trading strategy, which was previously presented in [4]. This trading strategy, called DC+GA, used a genetic algorithm to optimise the recommendations of multiple DC thresholds; it also used Equation 2, as its trend reversal estimator. We present this trading strategy in more detail in the next section. In order to test the effect of our GP equations, we replace Equation 2 from DC+GA with the best performing equation returned by our GP. We refer to this trading algorithm as GP+DC+GA. In addition, we also run tests with Equation 1 as the trend reversal estimator; we refer to this as O+GA. Lastly, we also run experiments with a technical analysis algorithm, and also buy and hold. All these algorithms are summarised below.

4.3.1. Technical analysis. There is a large number of technical indicators that are used in algorithmic trading. In order to combine different indicators to formulate trading strategies, we use a GP algorithm called EDDIE [19], [20], [21], [22]. This algorithm has demonstrated in all of the above works its ability to generate profitable strategies. The technical indicators EDDIE utilised in our experiments are moving average, trade break out, filter, volatility, momentum and momentum moving average.

4.3.2. Buy and hold. Buy and hold is a common benchmark for trading algorithms. A trading strategy that buys a financial product at the start of a period with the expectation that price will appreciate over time, then sell at the end of the period.

4.3.3. DC+GA.

This is the original DC work presented in [4], and is using Equation 2. The idea behind this trading strategy is to use multiple DC thresholds. This is because different thresholds provide different perspective of the data under observation. Smaller threshold sizes are used in detecting more events, and this allows traders react more promptly to price movement. However, this might not be an optimum strategy because there is a transaction cost[2] associated with trading. On the other hand, with a larger threshold, fewer events are detected, providing opportunities of taking action when price change is more sizeable. Selecting a threshold that is too large can lead to inaction or opportunity loss[3]. Thus, this trading strategy combines the use of different threshold values in an attempt to take advantage of the different characteristics of smaller and larger thresholds. At any point in time, each threshold will be recommending an action: buy, hold, or sell. As there are multiple thresholds, each threshold might recommend a different action. In order to decide which action to take, we use a weight system, where each DC threshold is assigned a weight. We then undertake a majority vote, where the recommendation with the highest sum of weights, wins. For example, if we have 5 DC thresholds, and the first three recommend buy and the remaining two sell, we would then sum up the weight values of the first three thresholds, and compare it to the sum of the weights of the last two thresholds; depending on which sum is higher, we would follow the respective action. The weights are not fixed, but are evolved by a genetic algorithm. More information about this trading strategy can be found in [4].

4.3.4. O+GA.

This is a modified version of DC+GA. Equation 2 in DC+GA is replaced with Equation 1 proposed by [3].

4.3.5. GP+DC+GA (proposed algorithm).

This is also a modified version of DC+GA. Equation 2 in DC+GA is replaced with the equations returned by our GP.

Finally, except for buy and hold trading strategy, the rest of the trading strategies we tested are evolutionary based and required parameter tuning. These tuned parameters are population size, number of generations, tournament size, crossover probability, mutation probability, and number of thresholds (for the multi-threshold DC strategies). Table 3 presents the value for each parameter. Please note that we used the same parameter configuration for all DC based trading strategies to avoid bias. In addition, it should be mentioned that for all evolutionary algorithms the experiments are run 50 times on each dataset and the results presented correspond to the average value over the 50 executions. On the other hand, the buy and hold strategy is run just once per dataset, since it represents a deterministic strategy.

2. Transaction cost: The cost incurred for trading a currency with another

3. Opportunity loss: Getting locked-in to a position because the event direction never changed by the defined threshold along the rest of the price coast line

TABLE 3: Trading strategy experimental parameters determined using I/F-Race.

Parameter	EDDIE	DC+GA	O+GA	GP+DC+GA
Population	500	1000	1000	1000
Generation	30	35	35	35
Tournament size	2	7	7	7
Crossover probability	0.9	0.9	0.9	0.9
Mutation probability	0.1	0.1	0.1	0.1

5. Results and analysis

This section is divided in three parts. In the first part (Section 5.1), we present the results from the first step of our experiments, where we compare the regression performance of our proposed GP to Equations 1 and 2. In the second part (Section 5.2), we present trading results from the DC based trading strategies O+GA, DC+GA and GP+DC+GA, as well as EDDIE and buy and hold. Finally, we summarise and discuss our findings in Section 5.3.

5.1. Regression results

To evaluate our algorithm, we used 250 DC datasets from the month of August 2013 to May 2014. Table 4 presents the average of root mean squared error for these datasets. From the table we see that the GP consistently outperformed both Equation 1 and Equation 2 at estimating OS length in all currency pairs. To confirm the above results, we applied Friedman's non-parametric statistical test. Our null hypothesis is that the algorithms come from the same continuous distribution. The result of the statistical test presented in Table 5 shows Equation 3 ranking the highest. The adjusted p-value at the α = 0.05 level, according to the Hommel post-hoc, shows the differences in ranks of our results to be statistically significant.

The above results demonstrate that the GP was not only able to create new equations describing the relationship between DC and OS length, but was also able to do this with significantly smaller error, when compared to Equations 1 and 2. Our interest now shifts to using these new equations as part of a DC-based trading strategy, to investigate whether these new and improved equations can lead to increased profit margins.

5.2. Comparison among trading algorithms

Table 6 presents the average (over 50 runs) daily returns after trading 5 currency pairs from August 2013 to May 2014. We observe that GP+DC+GA has higher mean return than DC+GA and O+GA (0.01896%, compared to 0.01125% and -0.0093%). One interesting thing to note is that the positive returns were obtained when EUR is the base currency[4]. GP+DC+GA was outperformed by all the other DC trading strategies when GBP is the base currency.

4. The based currency is the currency against which exchange rates is quoted.

TABLE 4: Estimation results by OS length estimator algorithms. 10-minute interval data. Results show RMSE value. They are averaged over 5 different thresholds (0.010%, 0.013%, 0.015%, 0.018%, 0.020%), 5 different currency pairs and 10 different datasets (August 2013 to May 2014).

Algorithm	GP	Equation 2	Equation 1
EUR/GBP	**6.77462**	7.15624	6.82228
EUR/JPY	**4.08026**	4.42172	4.70026
EUR/USD	**5.77218**	6.27207	6.41959
GBP/CHF	**5.86789**	6.36392	6.33501
GBP/USD	**6.09010**	6.66513	6.42833
Mean	**5.71024**	6.16841	6.13421

TABLE 5: : Statistical test results of OS length estimation according to the non-parametric Friedman test with the Hommel post-hoc test. Significant differences at the α = 0.05 level

Algorithm	Average Rank	$Adjust_{pHomm}$
GP (c)	1.272	-
Equation 1	**2.332**	2.12E-32
Equation 2	**2.396**	6.44E-36

Nevertheless, the profit margin obtained was quite significant that it offset the losses. It is also important to note that $O + GA$, which was using Equation 1 had on average negative returns (-0.0093%).

We also compared profitability between GP+DC+GA and physical time based trading strategies. Results showed that returns obtained by GP+DC+GA exceeded that of EDDIE and buy and hold. In fact, EDDIE experienced negative returns in 4 out of the 5 currency pairs (EUR/JPY, EUR/USD, GBP/CHF, GBP/USD), which also resulted in a negative mean return of -0.00076%. The return for buy and hold strategy is not included in the table because it is run one time per dataset. On average over all currency pairs analysed, buy and hold strategy made a return of 0.01274% which is less than 0.01896% made by GP+DC+GA.

Additionally, we performed Friedman's non-parametric statistical test to evaluate the statistical significance of the returns by the evolutionary algorithm based trading strategies. The null hypothesis is that the trading strategies come from the same continuous distribution. From the result presented in Table 7, we observed that returns from GP+DC+GA is ranked the highest and they statistically outperformed DC+GA and $EDDIE$ at α = 0.05 level. There was no statistical significance between GP+DC+GA and O+GA; nevertheless, GP+DC+GA was ranked higher than O+GA.

Table 8 details the number of positive trading days over which the excess return in Table 6 was made. This information further shows the consistency of the trading strategies over the trading period. GP+DC+GA is ranked second, it achieve 50% positive trading days across all currencies. O+GA had the highest number of positive trading days (58%) across all currency pairs analysed, notwithstanding, result indicates that of the 5 currency pairs, GP+DC+GA had more profitable days than O+GA in 3 (EUR/GBP, EUR/JPY, GBP/USD). O+GA incurred high loss during the non-profitable days and this was evident in the negative excess return of -0.0093%. Similarly, EDDIE had the same number of profitable days as GP+DC+GA but loss incurred during the non-profitable days was high and it led to a negative excess return. Finally DC+GA had the least number of profitable days (36%), the excess return was large enough to remain profitable after trading across all currency pairs, nevertheless it was not enough to surpass GP+DC+GA due to the difference in the number of profitable days. This result is a reinforcement that GP+DC+GA is better than all other trading strategies benchmarked.

TABLE 6: Mean return results for all trading strategies. DC strategies using 5 thresholds. 10-minute interval data. 5 different currency pairs and 10 different datasets (August 2013 to May 2014). Results shown in % values.

Algorithm	GP+DC+GA	DC+GA	O+GA	EDDIE
EUR/GBP	0.00703	0.00058	0.00341	0.00001
EUR/JPY	0.11600	0.06327	-0.07723	-0.00378
EUR/USD	0.00733	0.00055	0.02455	-0.00002
GBP/CHF	-0.0198	-0.00357	0.00903	0.00004
GBP/USD	-0.00629	-0.00045	-0.00580	0.00001
Mean	**0.01896**	**0.01125**	**-0.0093**	**-0.00076**

TABLE 7: Statistical test of trading returns according to the non-parametric Friedman test with Homel post-hoc test. 10-minute interval data. Significant differences at the α = 0.05 level

Algorithm	Average ranking	$Adjust_{pHomm}$
GP+DC+GA (c)	1.64	-
O+GA	1.76	0.64
DC+GA	**2.78**	2.01E-17
EDDIE	**3.82**	9.27E-17

TABLE 8: Total number of positive months per currency pair in 10 months in % values

Algorithm	GP+DC+GA	DC+GA	O+GA	EDDIE
EUR/GBP	60%	50%	50%	60%
EUR/JPY	60%	40%	50%	20%
EUR/USD	50%	30%	80%	40%
GBP/CHF	30%	20%	70%	70%
GBP/USD	50%	40%	40%	60%
Total	**50%**	**36%**	**58%**	**50%**

5.3. Summary

Based on our experimentations and results, we achieved our goal. Greater insight into trend reversal estimation in DC can be achieved if we can correctly estimate OS event length. Our out-of-sample experimentation results show that our algorithm was able to estimate OS length better than the other two algorithm currently in use. To evaluate the performance of our algorithm at trading, we embedded it in an existing trading strategy. On the whole, test results showed that our version of the trading strategy called GP+DC+GA outperformed other DC based and physical time based trading strategies. We also observed that there were some base currencies that our algorithm did not achieve the performance we had expected. In these cases we suspect that there might be some peculiarity in the datasets of these base currencies that our DC based trading strategy needed to learn to be profitable.

6. Conclusion

In conclusion, anticipating trend reversal is crucial for profitable trading. By estimating OS length, we are able to estimate trend reversal in DC once a directional change is confirmed. We used GP to find equations that improved estimation of OS event length. We showed that our GP is able to estimate OS length given a DC length with less error than other known OS length estimation techniques. We showed that we are able to outperform two intrinsic-time and two physical-time approaches (technical analysis, buy and hold). We traded in 5 Forex spot markets and used 10 months data captured in 10-minute interval.

As a motivation for future work, our GP evolved Equation 3 using subtree mutation and subtree crossover. It might be beneficial to investigate introduction of additional genetic operators to discover even richer OS length estimators to achieve improved trend reversal estimation. More work needs to be done to test our methodology in more Forex markets, using data captured with higher frequency (e.g. tick data) and over longer periods (e.g. 5 years). Finally, further work can also be done to combine DC based trading strategy and physical time based strategy. This is because we observed that in currency pairs where our trading strategy performed badly, technical indicator based strategy had positive returns and vice-versa. Nevertheless, it is yet to be seen whether this strategy can improve trading returns even further.

References

[1] C. Evans, K. Pappas, and F. Xhafa, "Utilizing artificial neural networks and genetic algorithms to build an algo-trading model for intra-day foreign exchange speculation," *Mathematical and Computer Modelling*, vol. 58, no. 5, pp. 1249–1266, 2013.

[2] A. Bakhach, E. P. K. Tsang, and H. Jalalian, "Forecasting directional changes in fx markets," in *IEEE Symposium on Computational Intelligence for Financial Engineering & Economics (IEEE CIFEr'16)*. Athens Greece: IEEE, December 2016, pp. 6–9.

[3] J. Glattfelder, A. Dupuis, and R. Olsen, "Patterns in high-frequency fx data: discovery of 12 empirical scaling laws," *Quantitative Finance*, vol. 11, no. 4, pp. 599–614, 2011.

[4] M. Kampouridis and F. E. Otero, "Evolving trading strategies using directional changes," *Expert Systems with Applications*, vol. 73, pp. 145–160, 2017.

[5] E. P. Tsang, R. Tao, A. Serguieva, and S. Ma, "Profiling high-frequency equity price movements in directional changes," *Quantitative Finance*, pp. 1–9, 2016.

[6] D. M. Guillaume, M. M. Dacorogna, R. R. Davé, U. A. Müller, R. B. Olsen, and O. V. Pictet, "From the bird's eye to the microscope: A survey of new stylized facts of the intra-daily foreign exchange markets," *Finance and stochastics*, vol. 1, no. 2, pp. 95–129, 1997.

[7] M. Aloud and M. Fasli, "The impact of strategies on the stylized facts in the fx market," 2013.

[8] M. E. Aloud, "Time series analysis indicators under directional changes: The case of saudi stock market," *International Journal of Economics and Financial Issues*, vol. 6, no. 1, 2016.

[9] T. Bisig, A. Dupuis, V. Impagliazzo, and R. Olsen, "The scale of market quakes," *Quantitative Finance*, vol. 12, no. 4, pp. 501–508, 2012.

[10] M. Aloud, E. Tsang, and R. Olsen, "Modeling the fx market traders behavior: An agent-based approach," *Banking, Finance, and Accounting: Concepts, Methodologies, Tools, and Applications: Concepts, Methodologies, Tools, and Applications*, p. 350, 2014.

[11] M. Aloud, "Directional-change event trading strategy: Profit-maximizing learning strategy," *The Seventh International Conference on Advanced Cognitive Technology and Applications*, 2016.

[12] M. E. Aloud, "Profitability of directional change based trading strategies: The case of saudi stock market," *International Journal of Economics and Financial Issues*, vol. 6, no. 1, 2016.

[13] A. Kablan and W. L. Ng, "Intraday high-frequency fx trading with adaptive neuro-fuzzy inference systems," *International Journal of Financial Markets and Derivatives*, vol. 2, no. 1-2, pp. 68–87, 2011.

[14] A. Bakhach, E. P. Tsang, and W. L. Ng, "Forecasting directional changes in financial markets," 2015.

[15] J. Gypteau, F. E. Otero, and M. Kampouridis, "Generating directional change based trading strategies with genetic programming," in *European Conference on the Applications of Evolutionary Computation*. Springer, 2015, pp. 267–278.

[16] J. R. Koza, *Genetic programming: on the programming of computers by means of natural selection*. MIT press, 1992, vol. 1.

[17] M. López-Ibánez, J. Dubois-Lacoste, T. Stützle, and M. Birattari, "The irace package, iterated race for automatic algorithm configuration," Citeseer, Tech. Rep., 2011.

[18] M. Birattari, Z. Yuan, P. Balaprakash, and T. Stützle, "F-race and iterated f-race: An overview," in *Experimental methods for the analysis of optimization algorithms*. Springer, 2010, pp. 311–336.

[19] M. Kampouridis and E. Tsang, "Investment opportunities forecasting: Extending the grammar of a gp-based tool," *International Journal of Computational Intelligence Systems*, vol. 5, no. 3, pp. 530–541, 2012.

[20] M. Kampouridis, A. Alsheddy, and E. Tsang, "On the investigation of hyper-heuristics on a financial forecasting problem," *Annals of Mathematics and Artificial Intelligence*, vol. 68 (4), pp. 225–246, 2013.

[21] F. E. B. Otero and M. Kampouridis, *A Comparative Study on the Use of Classification Algorithms in Financial Forecasting*. Springer Berlin Heidelberg, 2014, pp. 276–287.

[22] M. Kampouridis and F. E. Otero, "Heuristic procedures for improving the predictability of a genetic programming financial forecasting algorithm," *Soft Computing*, pp. 1–16, 2015.