Genetic programming for combining directional changes indicators in international stock markets

Xinpeng Long[1 [0000−0002−7506−8371]], Michael Kampouridis[1 [0000−0003−0047−7565]], and Panagiotis Kanellopoulos[1 [0000−0002−8093−1379]]

School of Computer Science and Electronic Engineering,
University of Essex, Wivenhoe Park, UK
`[xl19586, mkampo, panagiotis.kanellopoulos]@essex.ac.uk`

Abstract. The majority of algorithmic trading studies use data under fixed physical time intervals, such as daily closing prices, which makes the flow of time discontinuous. An alternative approach, namely directional changes (DC), is able to convert physical time interval series into event-based series and allows traders to analyse price movement in a novel way. Previous work on DC has focused on proposing new DC-based indicators, similar to indicators derived from technical analysis. However, very little work has been done in combining these indicators under a trading strategy. Meanwhile, genetic programming (GP) has also demonstrated competitiveness in algorithmic trading, but the performance of GP under the DC framework remains largely unexplored. In this paper, we present a novel GP that uses DC-based indicators to form trading strategies, namely GP-DC. We evaluate the cumulative return, rate of return, risk, and Sharpe ratio of the GP-DC trading strategies under 33 datasets from 3 international stock markets, and we compare the GP's performance to strategies derived under physical time, namely GP-PT, and also to a buy and hold trading strategy. Our results show that the GP-DC is able to outperform both GP-PT and the buy and hold strategy, making DC-based trading strategies a powerful complementary approach for algorithmic trading.

Keywords: Directional changes · Genetic programming · Algorithmic trading.

1 Introduction

Algorithmic trading has always been a vibrant research topic of paramount importance within the finance domain [8]. The majority of algorithmic trading research takes place on physical time scale, e.g. using hourly, daily, and weekly data. However, using such fixed time scales has the drawback of making data discontinuous and omitting important information between two data points, e.g. daily data would not have captured the flash crash that occurred across US stock indices on 6 May 2010 from 2:32pm to 3:08pm, as prices rebounded shortly afterwards [2].

An alternative approach is to summarise prices as events. The rationale is to record key events in the market representing significant movements in price, such as a change of, for instance, 5%. Directional changes (DC) is a relatively recent event-based technique, which relies on a threshold θ to detect significant price movements.

It was first proposed in [12] and formally defined in [19]. In the DC framework, a physical time series is divided into upward and downward trends, where each such trend marks a DC event at the moment the price change exceeds θ; the DC event is usually followed by an overshoot (OS) event representing the time interval of price movement along the trend beyond the DC event.

In this work, we are interested in using DC-based indicators to perform algorithmic trading. Indicators are mathematical patterns derived from past data and are used to predict future price trends. They are commonly used in technical analysis, e.g., in the form of moving averages, and trade breakout rules. With the evolution of DC research, new DC-based indicators have been proposed, see e.g.[3][20][21]. Therefore, in this paper we will combine 28 different DC indicators under a genetic programming (GP) algorithm [18], namely GP-DC. We apply the derived trading strategies to 33 different datasets from three international markets, namely the DAX performance index, Nikkei 225, and the Russell 2000 index. Our goal is to show that the DC paradigm is not only competitive compared to the physical time paradigm, but has even the potential to outperform it. To achieve this goal, we benchmark GP-DC with another GP-based physical time trading strategy, namely GP-PT, that uses technical analysis indicators under physical time. We compare the GP-DC's results to results obtained by GP-PT. We compare the two GPs' performance on different financial metrics, such as cumulative returns, average rate of return per trade, risk, and Sharpe ratio. We also compare the GPs' performance against the buy-and-hold strategy, which is a common financial benchmark.

The remainder of this paper is organized as follows. In Section 2, we present background information and discuss the DC-related literature. Section 3 introduces the methodology of our experiments, and then Section 4 presents the experimental set up, as well as the datasets, benchmarks, and parameter tuning process. Section 5 presents the experimental results and finally, Section 6 concludes the paper and discusses future work.

2 Background and Literature Review

2.1 Overview of directional changes

Directional Changes form an event-based approach for summarising market price movements, as opposed to a fixed-interval-based approach. A DC event is identified only when the price movement of the objective financial instrument exceeds a threshold predefined by the trader. Depending on the direction of price movement, such DC events could be either upturn events or downturn events. Frequently, after the confirmation of a DC event, an overshoot (OS) event follows; the OS event ends when a new price movement starts in an opposite trend, eventually leading to a new DC event. Recent studies, however, have pointed out that a DC event is not necessarily followed by an OS one [1].

Figure 1 presents an example of how to convert physical time series to DC and OS events using two different thresholds (see the red and blue lines). Note that thresholds may in principle vary, as traders need not necessarily agree on which price

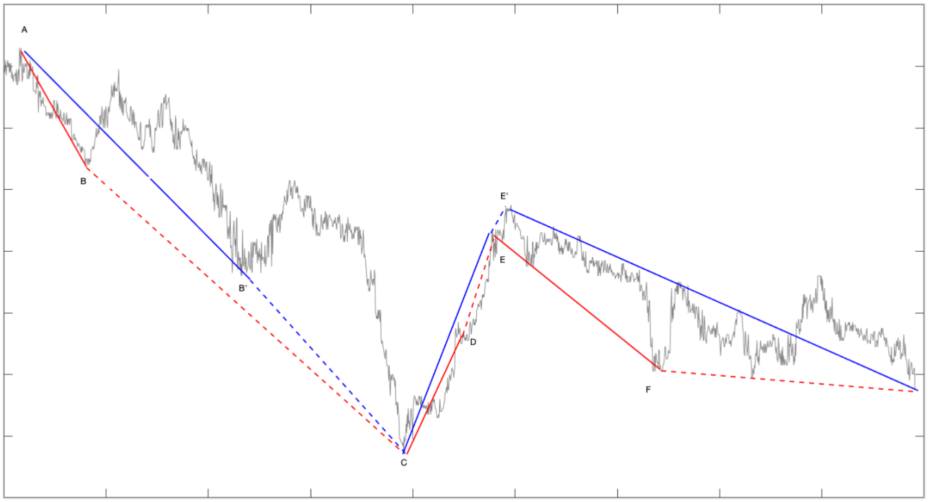

Fig. 1. An example for DC. The grey line indicates the physical time series, the red line denotes a series of DC and OS events as defined by a threshold of 0.01%, while the blue line denotes a series of DC and OS events as defined by a threshold of 0.018%. DC events are depicted with solid lines, while dotted lines denote the OS events.

movement constitutes a significant event; each such threshold leads to a different event series. A smaller threshold leads to the identification of more events and increases the opportunity of trade, while a larger threshold leads to fewer events with greater price movement. Thus, selecting an appropriate threshold is a key challenge.

By looking at the historical daily price movement (grey line) and the events created by the threshold of 0.01% (red lines), there are plenty of price movements that are not classified as events under the DC framework, as these do not exceed the threshold. Only when a price change is larger than the threshold is the time series divided into DC events (solid lines) and OS events (dotted lines). For example, the solid red line from A to B is considered a DC event on a downturn, while an OS event follows (from B to C). Then, a new DC event (in the opposite direction) is detected from C to D and this is followed by an OS event from D to E in an upturn, and so on.

It is worth noting that the change of trend can be confirmed only when the price movement exceeds the threshold. In other words, we do not know when the OS event ends until the next DC event (in the opposite direction) is confirmed. For example, in Figure 1, the point D is a DC event confirmation point. Before point D, the last OS event is considered to be still active, while the trader considers it to have been in a downward event. This leads to a paradox that on the one hand, in order to maximise returns, trades should be closed as near as possible to the endpoint of the OS event, and on the other hand, when the endpoint of the OS event is detected, it is already well beyond that point. Therefore, figuring out the extreme point where direction is reversed, such as point C in Figure 1, is an active research topic on the DC domain. In particular, several scaling laws have been suggested to identify the OS event length.

The advantage of DC is that it offer traders a new perspective on price movements; it allows them to focus on significant events and ignore other price movements that could be considered as noise. Therefore, DC leads to new research directions and challenges that are not relevant under physical time periods; in the following section, we present existing work on DC.

2.2 Related work

DC was first proposed by Guillaume et al. [12] and was formally defined by Tsang [19] as an alternative, event-based method to the traditional physical time model. Since DC is appropriate to handle non-fixed time intervals and high-frequency data, a series of papers applies it on tick data from the Forex market, see e.g. [7][15][14]. There exist two key issues in DC. The first is when do the OS events end; clearly, this has impact on profit maximisation. In other words, we are interested in figuring out the relationship between DC events and OS events. In this direction, Glattfelder et al. [11] introduced 12 new empirical scaling laws to establish quantitative relationships between price movements and transactions in the foreign exchange market. Following along this path, Aloud and Fasli [5] considered four new scaling laws under the DC framework and concluded that these perform successfully on the foreign exchange market. To name an example, one of the most prominent scaling laws states that OS takes, on average, twice as long to reach the same amount of price change as the DC event length. Recently, Adegboye and Kampouridis [1] proposed a novel DC trading strategy which does not assume that a DC event is always followed by an OS event; their results suggest that this strategy outperforms other DC-based trading strategies, as well as the buy and hold strategy, when tested on 20 Forex currency pairs.

The second key issue is the application of technical analysis under a DC framework; technical analysis has been frequently used on physical time by capturing features of markets, namely technical indicators. Aloud [3] converted physical time data into event-based data and introduced a first set of indicators tailored for the DC framework. Further DC indicators were suggested in [20] and [21]. These DC indicators were applied to summarise price changes in the Saudi Stock Market with the aim to help investors discover and capture valuable information. Furthermore, Ao and Tsang [6] proposed two DC-based trading strategies, namely TA1 and TA2, derived from the Average Overshoot Length scaling law. Their results indicated a positive return for most cases in FTSE 100, Hang Seng, NASDAQ 100, Nikkei 225, and S&P 500 stock market indices. Very recently, a combination of DC with reinforcement learning, trained by the Q-learning algorithm, was proposed by Aloud and Alkhamees [4] on S&P500, NASDAQ, and Dow Jones stock market. Their results showcase substantial return and an increase in the Sharpe ratio.

The above discussion reveals a relative scarcity of DC studies on the stock market. Moreover, using DC-based indicators to derive trading strategies is still in its infancy compared to the, well-established, technical analysis under physical time. We remark that GP has been very effective in the past in combining different (technical) indicators to derive profitable trading strategies, see e.g. [9] [10][13]. This naturally begs the question of how effective GP would be when combined with DC-based indicators, and, hence, motivates us to compare such an approach with a physical time model. Next, we introduce the GP methodology while also presenting the GP-DC trading strategy we used.

3 Methodology

This section presents GP-DC, a genetic programming approach using indicators suggested for the DC framework.

3.1 Genetic programming model

Terminal set After obtaining the daily closing prices for a dataset, we apply the DC framework to summarise the prices as events. Then, from the event series, we calculate the values of 28 indicators specific to the DC framework, much alike technical indicators being derived from technical analysis in physical time [16]. These 28 DC indicators have been introduced and discussed in [3] and, together with an Ephemeral Random Constant (ERC), form the terminal set. Whenever ERC is called, it returns a random number following the uniform distribution and ranging between -1 and 1. In order to fit the range of ERC, the DC indicators have been normalised.

Table 1 lists the DC indicators. In particular, there is a collection of 11 indicators, some of which are calculated over a certain period (e.g. the total number of DC events N_{DC} can be calculated over a period of 10, 20, 30, 40, or 50 days), thus leading to a total of 28 indicators. The third column in Table 1 takes the value N/A for indicators not requiring a period length (namely OSV, TMV, T_{DC}, and R_{DC}).

Table 1. DC indicators

Indicator	Description	Periods (days)
TMV	TMV is the price movement between the extreme point at the beginning and end of a trend, normalised by the threshold θ.	N/A
OSV	OSV is calculated by the percentage difference between the current price with the last directional change confirmation price divided by the threshold θ.	N/A
Average OSV	This is the average value of the OSV over the selected period.	3, 5, 10
R_{DC}	R_{DC} represents the time-adjusted return of DC. It could be calculated by the TMV times threshold θ divided by the time intervals between each extreme point.	N/A
Average R_{DC}	This is the average value of the R_{DC} over the selected period.	3, 5, 10
T_{DC}	This is the time spent on a trend.	N/A
Average T_{DC}	This is the average value of T_{DC} over the selected period.	3, 5, 10
N_{DC}	N_{DC} is the total number of DC events over the selected period.	10, 20, 30, 40, 50
C_{DC}	C_{DC} is defined as the sum of the absolute value of the TMV over the selected period.	10, 20, 30, 40, 50
A_T	A_T represents the difference between the time DC spends on the up trends and down trends over the selected period.	10, 20, 30, 40, 50

Function set The function set includes two logical operators, namely AND and OR, and two logical expressions, namely less than (<) and greater than (>).

Model representation The GP evolves logical expressions, where the root is one of AND, OR, $<$, or $>$. These expressions are then integrated as the first branch of an If-Then-Else (ITE) statement; see Part 1 of Figure 2. The rest of the ITE tree contains a 'Then' and an 'Else' branch; the former represents a buy action, and always returns a leaf node with a value of 1. The latter represents a hold action, and always returns a leaf node with a value of 0. Note that there is no sell action during this structure; we will discuss the part of sell action in Section 3.2. We did not include Part 2 in the GP is as its values are constants, either 0 or 1; there was thus no need to evolve them.

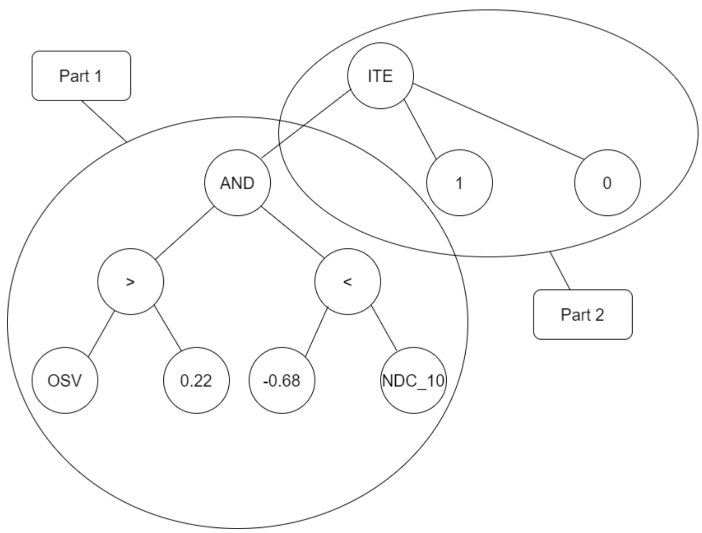

Fig. 2. An example of the GP tree and the If-Then-Else structures. If OSV is greater than 0.22 and N_{DC} for 10 days is greater than -0.68, then we get a signal for a buy action; otherwise, we hold.

Fitness function We use the Sharpe ratio as the fitness function of the GP trading strategies. The advantage of using the Sharpe ratio is that it takes into account both returns and risk:

$$\mathsf{SharpeRatio} = \frac{\mathsf{E}(R) - R_f}{\sqrt{\mathsf{Var}(R)}}, \tag{1}$$

where E and Var stand for the sample mean value and the sample variance, R stands for the rate of returns and R_f is the risk-free rate. The data used, i.e., the returns, for computing the Sharpe ratio were obtained by the trading algorithm outlined in Section 3.2, which indicates when the selling of the stocks will take place.

Selection method and operators We use elitism, sub-tree crossover and point mutation. We also use tournament selection to choose individuals as parents for the above operators.

A summary of the GP configuration is presented in Table 2.

Table 2. Configuration of the GP algorithm

Configuration	Value
Function set	AND, OR, >, <
Terminal set	28 DC indicators and ERC
Genetic operators	Elitism, subtree crossover and point mutation
Selection	Tournament

3.2 Trading strategy

The goal of the GP tree, which corresponds to our trading strategy, is to answer the question: "Is the stock price going to increase by r% within the next n days?". If the GP tree returns True, we buy one amount of stock, unless we already own the stock. If the GP tree returns False, we take no action (hold). When we already own a stock, and the price increases by r% within the next n days, we sell the stock on the given day this happens. If the price does not increase by r% within the next n days, we sell the stock on the n-th day. Note that short-selling is not allowed in this trading strategy. At the end of each sell action, we calculate and record the resulting profit. All positions take transaction costs into account; the transaction cost is 0.025% per trade. The above trading strategy is summarised in Algorithm 1.

Algorithm 1 Our trading strategy given threshold r% and duration n days

Require: Initialise variables (O represents the prediction of the GP tree, while $index$ indicates whether the stock is held)

if $O = 1$ and $index = 0$ **then**
  Buy one amount of stock
  $index \leftarrow 1$
  $N \leftarrow i$ //Starting time for trade: i is always the current time
  $K \leftarrow p$ //Stock price when buying: p is always the current price
else
  if ($index = 1$ and $p > (1 + r/100) \times K$) OR $(i - K) > n$ **then**
    Sell the stock
    $index \leftarrow 0$
    Calculate and record profit
  end if
end if

The rate of return from each trade is computed based on the price P_b we bought and the price P we sold the stock; see Equation (2). These returns are saved as a list and, eventually, we compute the sample mean of that list, which gives the overall rate of return; this is the input to Equation (1) to determine the Sharpe ratio. The *risk*, as seen in Equation 3, is the standard deviation of that list.

$$R = \left\{ \frac{0.99975 \cdot P - 1.00025 \cdot P_b}{1.00025 \cdot P_b} \right\} \cdot 100\% \tag{2}$$

$$\mathsf{Risk} = \sqrt{\mathsf{Var}(R)} \tag{3}$$

4 Experimental set up

4.1 Data

Recall that, as discussed in Section 1, our goal is to evaluate GP-DC algorithm on the stock market. We use data from three international markets, namely the DAX performance index, Nikkei 225, and the Russell 2000 index. From each index, we downloaded 10 stocks from Yahoo! Finance, as well as the data for the index themselves. Therefore in total, we use 33 datasets (3 markets × 10 stocks + 3 indices). Each dataset consists of daily closing prices for the period 2015 to 2020 and was split into three parts, namely training, validation, and test, as follows: 60%:20%:20%. All data were then converted into DC indicators (see Table 1), and normalised, as explained in Section 3.1.

4.2 Benchmarks

We compare the performance of the GP-DC trading strategy against GP-PT as well as buy-and-hold, a typical financial benchmark. For the GP-PT algorithm, we use the same GP as the one described above in Section 3. The only difference is that its terminals are now based on technical analysis (physical time), rather than directional changes. To make the comparison fairer, the number of technical indicators in the GP-PT algorithm is equal to that of the DC indicators in the GP-DC algorithm. These indicators are: each of Moving Average, Commodity Channel Index, Relative Strength Index, and William's %R with periods of 10, 20, 30, 40, and 50 days, each of Average True Range, and Exponential Moving Average with periods of 3, 5, 10 days, and finally, On Balance Volume and parabolic SAR without periods [17]; hence, we obtain 28 technical indicators.

4.3 Parameter tuning for GP

We performed a grid search to decide on the optimal GP parameters for both the GP-DC and GP-PT algorithms, and tuning took place by using the validation set. Based on [18], we adopted the most common values for each parameter, namely 4, 6, 8 (max depth); 100, 300, 500 (population size); 0.75, 0.85, 0.95 (crossover probability); 2, 4, 6 (tournament size); and 25, 35, 50 (number of generations). Mutation probability is equal to (1-crossover probability), so we did not need to separately tune this parameter. Table 3 shows the selected parameters and their value after tuning.

Table 3. Parameters of the GP algorithm

Parameters	Value
Max depth	6
Population size	500
Crossover probability	0.95
Tournament size	2
Numbers of generation	50

4.4 Parameter tuning for trading strategy

Recall that there are 3 parameters on our trading strategy, 2 parameters derived for the question "whether the stock price will increase by r% during the next n days ?" and one parameter is the threshold on DC. Rather than tuning the above parameters and then selecting the best set across all datasets (which is what we did for the GP), we decided to allow for tailored values for each dataset. The configuration space for these three parameters is presented in Table 4.

Buy and hold is also a useful benchmark, as it compares the GPs' performance against the market performance. We will thus also report the buy and hold performance of each dataset.

Table 4. Configuration space for the trading strategy

Parameters	Configuration space
n (days-ahead of prediction)	1, 5, 15
r (percentage of price movement)	1%, 5%, 10%, 20%
Threshold of DC	0.001, 0.002, 0.005, 0.01, 0.02

5 Result and analysis

In this section, we present our results for the DC model, the physical time model and the traditional benchmark of buy and hold. Our aim is to study the competitiveness of the DC-based indicators and whether the resulting trading strategies can outperform the traditional technical analysis (GP-PT) trading strategies.

5.1 Comparison between GP-DC and GP-PT

Table 5 presents summary statistics across all 33 datasets under rate of return (ROR), risk, and Sharpe ratio (SR). As we can observe, the GP-DC algorithm outperformed GP-PT algorithm in terms of average, median, and maximum results for ROR and SR. On the other hand, GP-PT algorithm did better in terms of average, median, and maximum risk.

Figure 3 presents the box plots of the above results, and we can reach similar conclusions as from Table 5. Furthermore, not only the values but also the overall box plot of GP-DC algorithm is higher in terms of ROR and SR, when compared to the GP-PT algorithm. When arguing about risk, the GP-DC's plot is higher than the GP-PT's one, indicating more risky behavior by GP-DC. Furthermore, the ROR for each trade of DC is concentrated above zero. In contrast, the results of the GP-PT algorithm have many negative values, which indicate that GP-DC algorithm is more competitive than the GP-PT algorithm in terms of rate of return.

To confirm the above results, we performed the non-parametric Kolmogorov-Smirnov test between the GP-DC and GP-PT results distributions. We ran the test for each metric (ROR, risk, and SR). The p-value for each test was 0.0082, 0.8107, and 0.6015, respectively. As the p-value for ROR was below 0.05, it denotes that the null hypothesis is rejected at the 5% significance level, thus making the differences in rate of return between GP-DC and GP-PT statistically significant. On the other hand, even though GP-PT algorithm had a lower risk, the differences were not statistically significant. Similarly, even though GP-DC algorithm outperformed GP-PT algorithm in terms of SR, their difference was not statistically significant.

Table 5. Summary statistics of the GP-DC and GP-PT algorithm. The best values per metric appear in boldface.

Measurement	Rate of return		Risk		Sharpe ratio (SR)	
Algorithms	GP-DC	GP-PT	GP-DC	GP-PT	GP-DC	GP-PT
Average	**1.4949**%	-0.0566%	0.1062	**0.0898**	**0.3403**	0.2919
Median	**1.7943**%	-0.2495%	0.0814	**0.0757**	**0.2985**	0.1207
Maximum	**9.7798**%	7.9318%	0.3273	**0.2340**	**1.3688**	1.3382
Minimum	-7.5580%	**-4.7622**%	0.0280	0.0280	-0.5037	**-0.2604**

These results show the potential of the DC approach to act as a complementary approach to the physical time one, as it can yield statistically higher returns than physical time technical analysis indicators. However, it should also be noted that this happened at the expense of a slightly higher risk. Therefore, it deserves further study whether more fine-tuned DC strategies can also lead to lower risk, or, perhaps, whether a mix of DC and physical time strategies is to be suggested.

5.2 Buy and hold

We now compare the performance of the GP-DC and GP-PT algorithms with the buy-and-hold strategy, where one unit of stock is bought on the first day of trading and sold on the last day. Because of the nature of buy-and-hold, the standard deviation cannot be calculated since there is only a single buy-sell action and thus a single profit value; similarly, we cannot calculate risk and SR. Besides, rate of return is not a very meaningful metric for comparison, as both GP-DC and GP-PT algorithms have a high number of trades, while buy-and-hold has a single trade. To make a fairer comparison, we instead use the cumulative returns over the test set.

As we can observe in Table 6, the GP-DC algorithm has a significantly higher average and median values compared to the GP-PT algorithm and buy-and-hold (GP-

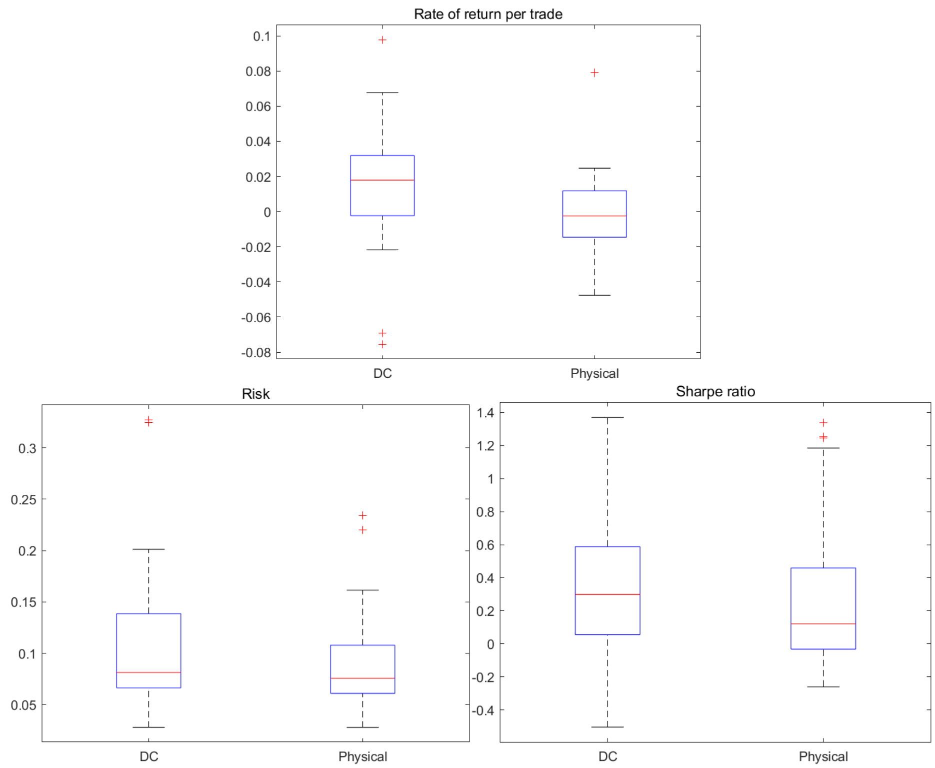

Fig. 3. Box plot of DC and physical time

DC average: 13.85%; median: 11.94%. GP-PT average: -1.53%; median: -2.73%. Buy-and-hold average: -4.08%; median: -10.81%). On the other hand, the highest cumulative returns is observed for buy-and-hold (around 135%), and the lowest for GP-DC (around -33%). It is also worth noting that the markets tested in this article are predominately bear markets, as it is also evident by the negative average and median cumulative returns of the buy and hold strategy. Therefore the fact that GP-DC algorithm has achieved strong average and median cumulative return performance indicates its high potential as a profitable trading paradigm.

The above results are also confirmed by looking at the distribution of results presented in Figure 4. As we can observe, the majority of the values presented in the box plot for GP-DC algorithm have higher values (i.e. cumulative returns) than the other two approaches. These results are supported by the Kolmogorov-Smirnov tests, which returned a p-value of 0.0082 in the comparison of GP-DC and GP-PT algorithms, and a p-value of 4.83E-04 for the comparison of DC and buy-and-hold. It should be noted that statistical significance in this case at the 5% level is for p-values below 0.025, after taking into account the Bonferroni correction for the (two) multiple comparisons.

Table 6. Cumulative returns of GP-DC, GP-PT, and buy and hold. Best values denoted in boldface.

Model	Average	Median	Maximum	Minimum
GP-DC	**13.8498%**	**11.9383%**	83.2537%	**-33.0880%**
GP-PT	-1.5341%	-2.7340%	59.4906%	-33.5400%
Buy and hold	-4.0821%	-10.8100%	**135.9218%**	-42.7290%

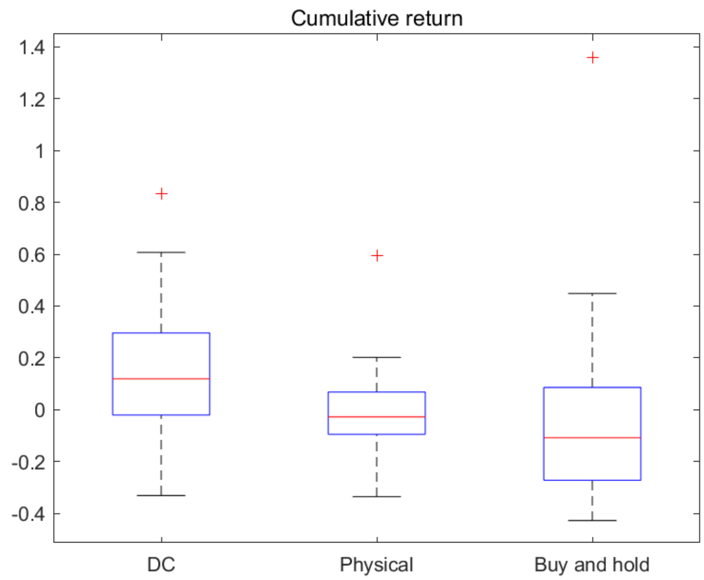

Fig. 4. Box plot of cumulative returns for GP-DC, GP-PT, and buy-and-hold

6 Conclusion

We have explored the benefit of combining genetic programming with indicators tailored for a directional changes framework. Our main contribution is to provide evidence for the effectiveness of this approach in the stock market. To do so, we conducted experiments on 33 datasets from 3 different international stock markets. Over these datasets, our approach (GP-DC) statistically outperformed the GP-PT algorithm, that combines genetic programming with technical indicators based on physical time, as well as the buy and hold strategy, in terms of cumulative return, rate of return, and Sharpe ratio. On the other hand, GP-PT algorithm had lower risk than GP-DC, although this finding is not statistically significant. The above results demonstrate that GP-DC is competitive against these two benchmarks in the stock market and can also be considered as a complementary technique to physical time.

Future work will thus focus on creating new trading strategies that combine technical analysis (physical time) and DC indicators. We believe that such strategies have the potential to bring in further improvements in profitability and risk and outperform the standalone strategies from technical analysis and directional changes.

References

1. Adegboye, A., Kampouridis, M.: Machine learning classification and regression models for predicting directional changes trend reversal in fx markets. Expert Systems with Applications **173**, 114645 (2021)
2. Adegboye, A., Kampouridis, M., Johnson, C.G.: Regression genetic programming for estimating trend end in foreign exchange market. In: 2017 IEEE Symposium Series on Computational Intelligence (SSCI). pp. 1–8. IEEE (2017)
3. Aloud, M.E.: Time series analysis indicators under directional changes: The case of saudi stock market. International Journal of Economics and Financial Issues **6**(1), 55–64 (2016)
4. Aloud, M.E., Alkhamees, N.: Intelligent algorithmic trading strategy using reinforcement learning and directional change. IEEE Access **9**, 114659–114671 (2021)
5. Aloud, M.E., Fasli, M.: Exploring trading strategies and their effects in the foreign exchange market. Computational Intelligence **33**(2) (2016)
6. Ao, H., Tsang, E.: Trading algorithms built with directional changes. In: 2019 IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFEr). pp. 1–7. IEEE (2019)
7. Bakhach, A., Tsang, E., Ng, W.L., Chinthalapati, V.R.: Backlash agent: A trading strategy based on directional change. In: 2016 IEEE Symposium Series on Computational Intelligence (SSCI). pp. 1–9. IEEE (2016)
8. Brabazon, A., Kampouridis, M., O'Neill, M.: Applications of genetic programming to finance and economics: past, present, future. Genetic Programming and Evolvable Machines **21**(1), 33–53 (2020)
9. Christodoulaki, E., Kampouridis, M., Kanellopoulos, P.: Technical and sentiment analysis in financial forecasting with genetic programming. IEEE Symposium on Computational Intelligence for Financial Engineering and Economics (CIFEr) (2022)
10. Claveria, O., Monte, E., Torra, S.: Evolutionary computation for macroeconomic forecasting. Computational Economics **53**(2), 833–849 (2019)
11. Glattfelder, J.B., Dupuis, A., Olsen, R.B.: Patterns in high-frequency fx data: discovery of 12 empirical scaling laws. Quantitative Finance **11**(4), 599–614 (2011)
12. Guillaume, D.M., Dacorogna, M.M., Davé, R.R., Müller, U.A., Olsen, R.B., Pictet, O.V.: From the bird's eye to the microscope: A survey of new stylized facts of the intra-daily foreign exchange markets. Finance and stochastics **1**(2), 95–129 (1997)
13. Hamida, S.B., Abdelmalek, W., Abid, F.: Applying dynamic training-subset selection methods using genetic programming for forecasting implied volatility. arXiv preprint arXiv:2007.07207 (2020)
14. Hussein, S.M.: Event-based microscopic analysis of the fx market. Ph.D. thesis, University of Essex (2013)
15. Kampouridis, M., Adegboye, A., Johnson, C.: Evolving directional changes trading strategies with a new event-based indicator. In: Asia-Pacific Conference on Simulated Evolution and Learning. pp. 727–738. Springer (2017)
16. Kampouridis, M., Tsang, E.: Investment opportunities forecasting: Extending the grammar of a gp-based tool. International Journal of Computational Intelligence Systems **5**(3), 530–541 (2012)
17. Kelotra, A., Pandey, P.: Stock market prediction using optimized deep-convlstm model. Big Data **8**(1), 5–24 (2020)
18. Poli, R., Langdon, W.B., McPhee, N.F.: A field guide to genetic programming. Published via http://lulu. com and freely available at http://www. gp-field-guide. org. uk.(With contributions by JR Koza) (2008)

19. Tsang, E.: Directional changes, definitions. Working Paper WP050-10 Centre for Computational Finance and Economic Agents (CCFEA), University of Essex Revised 1, Tech. Rep. (2010)
20. Tsang, E.P., Tao, R., Ma, S.: Profiling financial market dynamics under directional changes. Quantitative finance, http://www. tandfonline. com/doi/abs/10.1080/14697688.2016 **1164887** (2015)
21. Tsang, E.P., Tao, R., Serguieva, A., Ma, S.: Profiling high-frequency equity price movements in directional changes. Quantitative finance **17**(2), 217–225 (2017)