

## Article

# An Advanced Optimization Approach for Long-Short Pairs Trading Strategy Based on Correlation Coefficients and Bollinger Bands

Chun-Hao Chen <sup>1</sup>, Wei-Hsun Lai <sup>2</sup>, Shih-Ting Hung <sup>1</sup> and Tzung-Pei Hong <sup>2,3,\*</sup> 

<sup>1</sup> Department of Information and Finance Management, National Taipei University of Technology, Taipei 106, Taiwan; chchen@ntut.edu.tw (C.-H.C.); t109749005@ntut.org.tw (S.-T.H.)

<sup>2</sup> Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung 804, Taiwan; m083040004@student.nsysu.edu.tw

<sup>3</sup> Department of Computer Science and Information Engineering, National University of Kaohsiung, Kaohsiung 811, Taiwan

\* Correspondence: tphong@nuk.edu.tw

**Abstract:** In the financial market, commodity prices change over time, yielding profit opportunities. Various trading strategies have been proposed to yield good earnings. Pairs trading is one such critical, widely-used strategy with good effect. Given two highly correlated paired target stocks, the strategy suggests buying one when its price falls behind, selling it when its stock price converges, and operating the other stock inversely. In the existing approach, the genetic Bollinger Bands and correlation-coefficient-based pairs trading strategy (GBCPT) utilizes optimization technology to determine the parameters for correlation-based candidate pairs and discover Bollinger Bands-based trading signals. The correlation coefficients are used to calculate the relationship between two stocks through their historical stock prices, and the Bollinger Bands are indicators composed of the moving averages and standard deviations of the stocks. In this paper, to achieve more robust and reliable trading performance, AGBCPT, an advanced GBCPT algorithm, is proposed to take into account volatility and more critical parameters that influence profitability. It encodes six critical parameters into a chromosome. To evaluate the fitness of a chromosome, the encoded parameters are utilized to observe the trading pairs and their trading signals generated from Bollinger Bands. The fitness value is then calculated by the average return and volatility of the long and short trading pairs. The genetic process is repeated to find suitable parameters until the termination condition is met. Experiments on 44 stocks selected from the Taiwan 50 Index are conducted, showing the merits and effectiveness of the proposed approach.



**Citation:** Chen, C.-H.; Lai, W.-H.; Hung, S.-T.; Hong, T.-P. An Advanced Optimization Approach for Long-Short Pairs Trading Strategy Based on Correlation Coefficients and Bollinger Bands. *Appl. Sci.* **2022**, *12*, 1052. <https://doi.org/10.3390/app12031052>

Academic Editor: Krzysztof Kozsela

Received: 27 November 2021

Accepted: 17 January 2022

Published: 20 January 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** Bollinger Bands; correlation coefficient; genetic algorithm; pairs trading strategy; trading strategy optimization

## 1. Introduction

In financial markets, investment assets include bonds, funds, stocks, and other derivative financial products, for instance, futures and options. Investors are familiar with the basic principle of profitability: buy an asset at a low price and sell it at a higher price. The difficult part is that appropriate trading signals are hard to find, given the various assets and trends in real financial markets. Because of this phenomenon it is difficult to make a profit. Thus many approaches have been proposed for finding trading strategies that make profits more stable [1–7].

Such trading strategies involve a wide variety of different approaches [8–12], including regression, fuzzy theory, genetic algorithms (GA), artificial neural networks (ANN), memetic algorithms (MA), and support vector machine (SVM), etc. According to the application type, trading strategies in the literature can be divided into two categories:

(1) prediction of financial time series [2,4,6,13–16]; and (2) stock selection, portfolio management, and optimization [7,17–19].

Of these, pairs trading is a critical, widely used trading strategy [20–24] based on a central concept: for two highly-correlated assets, buy when one stock price falls behind and sell when the stock prices converge; this constitutes an arbitrage opportunity [25]. In other words, a profitable pairs trading strategy make take into account how to find a pair of highly correlated stocks and how to generate useful trading signals for buying and selling. Pairs trading can also be applied more widely, e.g., to cryptocurrency and prosumer markets [26,27].

The genetic Bollinger Bands and correlation-coefficient based pairs trading algorithm (GBCPT) was proposed by Huang [28]. It involves an optimization approach to determine parameters for correlation-based candidate pair generation and the Bollinger Bands-based trading signal discovery process. Stock pairs whose correlation coefficients meet the pre-defined threshold are expected to show more discrete trends in the future. In addition, Bollinger Bands are used to determine the rise/fall degrees of the pair. When both conditions are met, the transaction is longed for expected rises and shorted for declining stocks. The pair transaction is closed when the ending conditions of the Bollinger Bands are met. However, there are other parameters in pairs trading that affect the profitability of the strategy; these should be taken into consideration when designing the fitness function.

To solve the above-mentioned problems, we propose the advanced genetic Bollinger Bands and correlation-coefficient based pairs trading algorithm (AGBCPT) to achieve more robust and reliable trading performance. The algorithm encodes six critical parameters into a chromosome: the correlation coefficient threshold, the entry width of the Bollinger Bands, the out width of the Bollinger Bands, the correlation coefficient calculation days, the moving average calculation days, and the forward observation days. When evaluating fitness using such a chromosome, the encoded parameters are utilized to observe the trading pairs and their trading signals generated from the Bollinger Bands, after which the fitness value is calculated by the average return and volatility of long and short trading pairs. The genetic process is repeated to find suitable parameters until the termination conditions are satisfied. Experiments conducted on 44 stocks selected from the Taiwan 50 Index show the merits and effectiveness of the proposed approach.

This paper is organized as follows. Related work is described in Section 2 and the details of the proposed AGBCPT method are stated in Section 3. The experimental results are discussed in Section 4, and Section 5 concludes and outlines future work.

## 2. Related Work

### 2.1. Review of Pairs Trading Strategies

Pairs trading is a neutral trading strategy that investors utilize to yield profits from changing market situations [1,20,21,23,29]. Based on the historical performance of high correlated commodities, a pairs trading strategy focuses on how to observe the trading pair as a target and achieve profit from it [21]. When the correlation weakens, for instance, one stock rises and the other falls. Such a temporary discrete situation can be caused by changes in supply and demand, a sudden large number of transactions by a securities firm, or major news. These factors cause stock fluctuations. A pairs trading strategy then shorts the rising stock and longs the falling one at the same time because investors expect the price difference between the two to converge in the future [23,29]. Krauss classifies pairs trading strategies into distance methods, cointegration methods, time series methods, stochastic control methods, and other methods [23]. In recent years, abundant related research has been produced [25,30–36]. Below, we introduce approaches related to pair trading.

In 2006, Gatev et al. published a well-known pairs trading paper. Their proposed GGR (Gatev, Goetzmann and Rouwenhorst) pairs trading method [25] used six-month trading periods from 1962 to 1997 on a large sample of the U.S. equities. After testing the profitability of several trading rules, they observed that their strategy yielded annualized excess returns of up to eleven percent at low exposure to systematic sources of risk. Do

et al. indicate paired transactions that can still make stable profits given market and trade costs [31,32]. Their study extends the GGR method, comparing the test data over different years and different industries and confirming that the declining profitability in pairs trading is mainly due to an increasing share of non-converging pairs. One experimental result also shows that more industrially matched portfolios yield more substantial profits than portfolios selected from the whole market. They thus reduce the convergence failure of the selected stock portfolio.

For various situations and purposes, pairs trading also works with other methods that improve the performance of the pairs trading strategy [37]. For example, Rende et al. experiment with the persistence-based decomposition (PBD) model in a large-scale high-frequency pairs trading application [38]. Their study provides empirical evidence to show that the model is well-suited to noisy high-frequency data in terms of model fitting and prediction. Stäbinger et al. develop a pairs trading framework based on a mean-reverting jump-diffusion model [39]. Their results show that the method performs well in terms of risk-reward characteristics. To find an optimized pairs trading strategy, Fallahpour et al. propose pairs trading strategy optimization based on reinforcement learning [40]. Results on S&P500 constituent stocks confirm the efficiency of the proposed method and show that their approach is superior to existing approaches.

In addition to the stock market, pairs trading strategies are also used in other financial fields. For example, Fil et al. propose the use of paired trading for the cryptocurrency market to find profit space [26]. In experiments, they shift the standard pairs trading from finance to cryptocurrency. The experimental results of the same use of paired trading show that the trading portfolio in the cryptocurrency market does not converge, and profitability is improved when using higher-frequency trading. In addition, Lintilhac et al. state that historically, pairs trading in bitcoin markets have been possible [41]. Due to the increasing needs for distributed energy trading, Oh et al. propose two pair-matching strategies for distributed prosumer energy trading that consider the properties of the trading rules and the statistical characteristics of participants [27]. The literature shows that pairs trading is an effective trading strategy used by investors to yield profits from different market situations in various financial fields.

## 2.2. Review of Optimization Approaches in Financial Applications

Genetic algorithms (GAs) are optimization algorithms widely used for solving complex problems in a variety of fields [42,43]. In the financial field, many applications utilize GAs to improve and search for near-optimal solutions in limited time [44]. For example, Chen et al. propose an optimization algorithm to address the diverse group stock portfolio optimization problem to obtain a diverse group stock portfolio using the grouping genetic algorithm (GGA) [18]. To identify good group trading strategy portfolios, Chen et al. propose an algorithm to not only obtain a reliable group trading strategy portfolio but also to find appropriate stop-loss and take-profit points based on the GGA [17]. Huang proposes a methodology for effective stock selection using support vector regression (SVR) and a GA [45]. He was first to use the SVR to generate surrogates for actual stock returns that, in turn, serve to provide reliable stock rankings. The GA is then used to optimize the parameters for the proposed model.

Chen et al. propose an approach for feature selection utilizing the GA, and use the selected features to construct a long short-term memory (LSTM) neural network model for stock prediction [13]. The results showed that the GA-LSTM model outperforms all baseline models for time series prediction. Cheong et al. propose a spatiotemporal convolutional neural network-based relational network (STCNN-RN) model for stock anomaly detection [46]. To improve the accuracy of the STCNN-RN model, the GA is then employed to identify outlier time points for use in the model to identify abnormal behaviors. They indicate that the model is effective on a multiple financial time series dataset for finding anomalous situations.

For pairs trading optimization, Sermpinis et al. propose a pairs trading structure based on deep reinforcement learning (DRL) and a GA [47]. They first apply the distance method (DM) and the cointegration approach (CA) to generate trading pairs from the given pair pool, after which trading actions are determined using the simple thresholds (ST) strategy, the GA, and DRL. They propose five pairs trading strategies for trading, including the DM-ST, CA-DM-ST, and CA-ST benchmark strategies, and the improved strategies CA-GA-ST and CA-DRL. In CA-GA-ST, the GA is utilized to find appropriate parameter settings, and in CA-DRL, DRL is employed to construct an agent using pairs trading rules and the differences between the two assets. They indicate that CA-DRL is superior to other strategies. Goldkamp et al. propose an intelligent system using mixed integer programming (MIP) and the multi-objective genetic algorithm (NSGA-II) for multivariate pairs trading [48]. It uses MIP to generate trading pairs. The risk and return are used as two conflicting objective functions when finding Pareto solutions using NSGA-II. The results indicate that multi-objective multivariate pairs trading outperforms traditional approaches.

Huang et al. propose an intelligent model for pairs trading based on GA [49]. In their approach, the GA is utilized to find the parameters of moving averages, Bollinger Bands, and stock weight coefficients for the model. Experimental results indicate that GA-based pairs trading effectively improves the performance of pairs trading and outperforms the benchmark in terms of return.

In addition, Huang proposes the genetic Bollinger Band and correlation-coefficient based pairs trading algorithm (GBCPT), using a GA for pairs trading [28]. GBCPT encodes the parameters into a chromosome, including the correlation coefficient threshold, the entry width of the Bollinger channel, and the exit width of the Bollinger channel. The last two parameters are used to determine the width of the Bollinger Bands. To evaluate the chromosome, they first use the correlation coefficient between companies to determine a suitable candidate combination with a correlation coefficient for purchase, after which the Bollinger channels are used as a reference indicator to find the buying and selling signals for the target pair. The average return is then calculated and set as the fitness of a chromosome. The genetic operators are utilized to generate new solutions. The selection operator is used to generate the next population. The genetic process is repeated until the termination condition is met.

### 2.3. Review of Bollinger Bands

Bollinger Bands are a type of statistical chart that indicates the price volatility of financial commodities over time. The following parameters control typical Bollinger Bands, including the moving average (MA) and the constant  $W$  for controlling the bandwidth. The MA of a trading day  $i$  is the average price from the trading day  $i - mDay$  to  $i - 1$ , where  $mDay$  determines the number of days for calculating MA. The constant  $W$  is used to control the bandwidth. The upper and the lower bands are the components of the Bollinger Bands. The upper and lower bands are calculated using  $(MA + W\sigma)$  and  $(MA - W\sigma)$ , where  $\sigma$  is the standard deviation of the given period. These parameters determine the form of the Bollinger Bands.

In the literature, many approaches take Bollinger Bands into consideration when designing trading strategies. For instance, Windasari et al. propose a technical analysis method that uses historical data and indicators to identify price fluctuations in a specific period [50]. Bollinger Bands and the Williams percent range are indicators used in the research to provide information about stock trends by following a particular pattern of buying/selling. For the dataset, they use the stocks of six companies from the Indonesia Stock Exchange. Their experimental results show that the average return of the companies performs well, which proves that Bollinger Bands are feasible as an indicator for finding trading signals. Prasetijo et al. propose trading strategies employing Bollinger Bands and parabolic SAR indicators [3]. They develop a web-based application by which to evaluate the performance of the proposed strategies.

### 3. Proposed Approach

In this section, we describe the proposed approach in detail. The flowchart of AGBCPT is presented in Section 3.1, and the AGBCPT components are introduced in Section 3.2, including the encoding scheme, the initial population, the fitness function, and the genetic operations. In Section 3.3, the AGBCPT algorithm is presented, followed by an example in Section 3.4.

#### 3.1. AGBCPT Flowchart

The AGBCPT flowchart is shown in Figure 1.

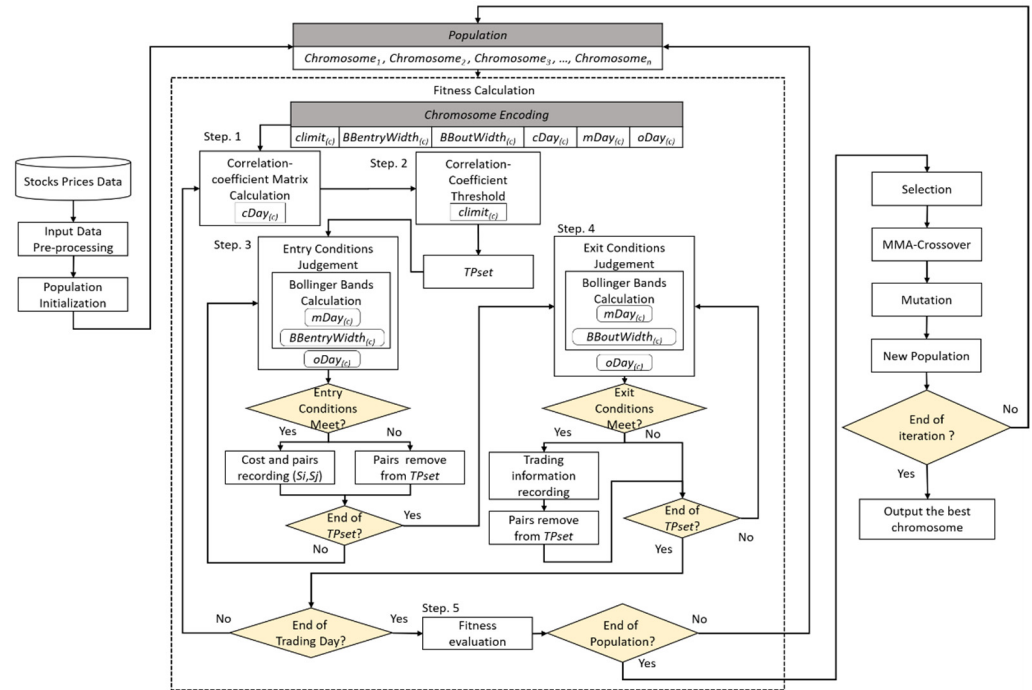


Figure 1. AGBCPT flowchart.

Figure 1 shows that the proposed approach collects the stock price series of the companies and then preprocesses the data, after which the population is randomly initialized according to the encoding scheme and the population size. The fitness calculation process determines the correlation coefficient matrix of all companies in each trading day  $T$  (Step 1). The number of days for the calculation is the  $cDay$  gene. Next, the  $cLimit$  gene is a threshold used to find the qualified stock pairs (Step 2), which are kept in  $TPset$  when their correlation coefficient value is smaller than  $cLimit$ . Then, the Bollinger Band channels for stock pairs are generated using the  $mDay$  and  $BBentryWidth$  genes (Step 3). On each date  $T$ ,  $mDay$  is used to calculate the moving average, and  $BBentryWidth$  is used to calculate the upper and lower channels. The formulas of the upper and lower channels of entering are defined as

$$UB_i(T) = MA_i(T) + BBentryWidth_c * \sqrt{\frac{\sum_{k=T-mDay_c}^{T-1} (cp_k^i - \mu^i)^2}{mDay_c}}, \text{ and} \quad (1)$$

$$LB_i(T) = MA_i(T) - BBentryWidth_c * \sqrt{\frac{\sum_{k=T-mDay_c}^{T-1} (cp_k^i - \mu^i)^2}{mDay_c}}, \quad (2)$$



where  $MA_i(T)$  is the  $i$ -th moving average calculated as

$$MA_i(T) = \frac{\sum_{k=T-mDay_c}^{T-1} cp_k^i}{mDay_c} \tag{3}$$

Then,  $oDay$  is used to check whether stock pair  $(s_i, s_j)$  is satisfied with the entry conditions, including (1)  $cp_{T-oDay}^i > UB_i(T) > cp_T^i$  for stock  $s_i$  and (2)  $cp_{T-oDay}^j < LB_j(T) < cp_T^j$  for stock  $s_j$ , where  $cp_T^h$  is the close price of stock  $s_h$  on date  $T$ . When both conditions are met and  $cp_T^i > cp_T^j$ , the proposed approach sells  $s_i$  and buys  $s_j$ , and the pair  $(s_i, s_j)$  is also recorded. It then continues to judge the entry conditions for the next candidate pair until all pairs are processed.

The next step is to generate the Bollinger Band channels again for the pairs that have been performed previously (Step 4). According to the  $mDay$  and  $BBoutWidth$ , the exiting channels are calculated as:

$$US_i(T) = MA_i(T) + BBoutWidth_c * \sqrt{\frac{\sum_{k=T-mDay_c}^{T-1} (cp_k^i - \mu^i)^2}{mDay_c}}, \text{ and} \tag{4}$$

$$LS_i(T) = MA_i(T) - BBoutWidth_c * \sqrt{\frac{\sum_{k=T-mDay_c}^{T-1} (cp_k^i - \mu^i)^2}{mDay_c}}. \tag{5}$$

The exiting conditions are (1)  $cp_{T-oDay}^i > LS_i(T) > cp_T^i$  for stock  $s_i$  and (2)  $cp_{T-oDay}^j < US_j(T) < cp_T^j$  for stock  $s_j$ , by which the proposed approach buys  $s_i$  and sells  $s_j$ . When a stock pair trading is complete, it records  $profit(s_i, s_j) = income(s_i, s_j) / cost(s_i, s_j)$  as well as the minimum value of the return, after which the trading pair  $(s_i, s_j)$  is removed from  $TPset$ , and it continues to judge the next pair's exit condition until all pairs have been processed.

Finally, the fitness value of a chromosome, that is, the profit of all trading pairs divided by the minimum value of the return, is evaluated and the genetic operators are executed to generate new offspring. The process is repeated until the termination conditions are met.

### 3.2. AGBCPT Components

In this section, we describe four AGBCPT components: the encoding scheme, the initial population, the fitness function, and the genetic operations.

#### 3.2.1. Encoding Scheme

The parameters used in the pairs trading strategy influence the pairs trading return. Because the strategy described here utilizes the correlation coefficient and Bollinger Bands, it takes into account the six parameters—correlation coefficient threshold ( $cLimit$ ), entry width of the Bollinger Bands ( $BBentryWidth$ ), out width of the Bollinger Bands ( $BBoutWidth$ ), correlation coefficient calculation days ( $cDay$ ), moving averages calculation days ( $mDay$ ), forward observation days ( $oDay$ )—and encodes them into a chromosome with real numbers. The correlation coefficient is applied to find potential stock pairs, and the Bollinger Bands are employed to find pairs trading signals. The encoding scheme of a chromosome is shown in Table 1.

**Table 1.** Encoding scheme of chromosome  $C_q$ .

Chromosome $C_q$					
$cLimit_q$	$BBentryWidth_q$	$BBoutWidth_q$	$mDay_q$	$cDay_q$	$oDay_q$

In Table 1, the genes representing  $cLimit$  and  $cDay$  belong to the correlation coefficient calculation, and  $mDay$ ,  $BBentryWidth$ ,  $BBoutWidth$ , and  $oDay$  belong to the Bollinger Bands.

The *cLimit* value is the threshold of the correlation coefficient set for finding potential stock pairs. *cDay* represents the days for calculating the correlation coefficient of two stocks. The *mDay*, *BBEntryWidth*, *BBoutWidth*, and *oDay* parameters are used in the Bollinger Bands. *mDay* represents the days for calculating the moving averages. The Bollinger Bands width of the up and down channels for the entry and exit signals are represented by *BBEntryWidth* and *BBoutWidth*. *oDay* represents the days of the stock price comparison for a trading signal.

### 3.2.2. Initial Population

According to the predefined ranges of the six parameters, the initial population is generated randomly at the given population size. The parameter ranges are shown in Table 2.

**Table 2.** Predefined ranges of six genes.

Name	Abbreviation	Range
Correlation coefficient threshold	<i>cLimit</i>	$-1 < cLimit < 1$
Entry width of Bollinger Bands	<i>BBEntryWidth</i>	$0 < BBEntryWidth < 2$
Out width of Bollinger Bands	<i>BBoutWidth</i>	$0 < BBoutWidth < BBEntryWidth$
Moving average calculation days	<i>mDay</i>	$5 \leq mDay \leq 20$
Correlation coefficient calculation days	<i>cDay</i>	$5 \leq cDay \leq 20$
Forward observation days	<i>oDay</i>	$1 \leq oDay \leq 3$

### 3.2.3. Fitness Function

Since the goal of the fitness function is to evaluate the quality of the chromosome, it is important to define an appropriate fitness function. In the proposed method, the GA is utilized to find appropriate parameters for the pairs trading strategy; therefore, the fitness value of a chromosome is evaluated by the profit and risk of a pairs trading strategy. Before starting the fitness function, the profit of a stock pair after *n* transactions using the trading strategy is defined as

$$profit_h(s_i, s_j) = \sum_{t=1}^n \frac{tpP_t(s_i, s_j)}{tpC_t(s_i, s_j)}, \tag{6}$$

where  $tpP_t(s_i, s_j)$  and  $tpC_t(s_i, s_j)$  are the income and the cost of the *h*-th stock pair ( $s_i, s_j$ ) in the *t*-th transaction, respectively. The total profit of a chromosome is then defined as

$$totalProfit(C_q) = \sum_{h=1}^{|TPset|} profit_h(s_i, s_j), \tag{7}$$

where *TPset* contains the qualified stock pairs and  $|TPset|$  is the number of stock pairs. The risk of a chromosome is defined as

$$risk(C_q) = \min(profit_1(s_i, s_j), \dots, profit_h(s_i, s_j), \dots, profit_{|TPset|}(s_i, s_j), 1), \tag{8}$$

where the function *min()* is used to find the smallest return from the set of  $profit_h(s_i, s_j)$ ; if all the returns are higher than one, the risk value is one.

According to the total profit and the risk factors, the fitness function of a chromosome is defined as

$$fitness(C_q) = \frac{totalProfit(C_q)}{risk(C_q)}. \tag{9}$$

In other words, the fitness value of a chromosome is evaluated by the sum of the return and the minimum return of all trading pairs.

### 3.2.4. Genetic Operations

The crossover and mutation genetic operations are described in this section. First, the max–min–arithmetic (MMA) crossover operator applied to the population in the proposed algorithm. It is executed as follows: (1) two chromosomes *Cq* and *Cp*, randomly selected

from the population, are  $C_q$ : [ $cLimit_q$ ,  $BBentryWidth_q$ ,  $BBoutWidth_q$ ,  $mDay_q$ ,  $cDay_q$ ,  $oDay_q$ ] and  $C_p$ : [ $cLimit_p$ ,  $BBentryWidth_p$ ,  $BBoutWidth_p$ ,  $mDay_p$ ,  $cDay_p$ ,  $oDay_p$ ]; (2) Then, four new chromosomes are generated by the four operators based on a predefined parameter  $d$  as

$$C_{new1}: [\min(cLimit_q, cLimit_p), \dots, \min(oDay_q, oDay_p)];$$

$$C_{new2}: [\max(cLimit_q, cLimit_p), \dots, \max(oDay_q, oDay_p)];$$

$$C_{new3}: [(d \times cLimit_q + (1 - d) \times cLimit_p), \dots, (d \times oDay_q + (1 - d) \times oDay_p)];$$

$$C_{new4}: [((1 - d) \times cLimit_q + d \times cLimit_p), \dots, ((1 - d) \times oDay_q + d \times oDay_p)].$$

A one-point mutation operator is applied to the population to generate new offspring. Every gene is mutated itself according to the mutation rate. Once a gene is selected for mutation, it randomly generates a new value based on the given range (see Table 2).

### 3.3. Proposed AGBCPT

Before describing the proposed AGBCPT, the notation is introduced in Table 3.

**Table 3.** AGBCPT notation.

Notation	Description
$cLimit$	Correlation coefficient threshold
$dTotal$	Final trading day
$mDay$	Moving average calculation days
$cDay$	Correlation coefficient calculation days
$oDay$	Forward observation days
$T$	Trading day
$BBentryWidth$	Entry width of Bollinger Bands
$BBoutWidth$	Out width of Bollinger Bands
$TPset = \emptyset$	Trading pair candidate set
$tp(s_i, s_j) = (s_i, s_j)$	Trading pair set ( $s_i, s_j$ )
$tpC(s_i, s_j)$	Trading pair cost ( $s_i, s_j$ )
$tpP(s_i, s_j)$	Trading pair income ( $s_i, s_j$ )
$Profit(s_i, s_j)$	Trading pair profit ( $s_i, s_j$ )

The proposed AGBCPT is described below:

**Input:** Selected companies:  $S = \{s_1, s_2, \dots, s_i, \dots, s_n\}$ ,  $1 \leq i \leq n$ , where  $n$  is the number of companies, and the closing prices of all the companies, with the  $i$ -th represented as  $CP_i = [cp_1^i, cp_2^i, cp_3^i, \dots, cp_t^i, \dots, cp_{dTotal}^i]$ ,  $1 \leq t \leq dTotal$ ,  $1 \leq i \leq NumCompanies$ , where  $dTotal$  is the last trading day and  $NumCompanies$  is the number of companies.

**Parameters:** Population size  $pSize$ , max generation  $maxGeneration$ , mutation rate  $mRate$ , crossover rate  $cRate$ , and parameter for the max-min arithmetical crossover operator  $d$ .

**Output:** Chromosome with highest fitness value  $bestChro$ .

**STEP 1:** Randomly initialize the population with population size  $pSize$ . Each chromosome has six genes: the correlation coefficient threshold ( $cLimit$ ), the entry width of the Bollinger Bands ( $BBentryWidth$ ), the out width of the Bollinger Bands ( $BBoutWidth$ ), the correlation coefficient calculation days ( $cDay$ ), the moving average calculation days ( $mDay$ ), and the forward observation days ( $oDay$ ).

**STEP 2:** Use the following steps to calculate the correlation coefficient matrix of  $n$  companies  $MT_{(n)}$ .



**STEP 2.1:** Obtain the historical closing prices  $CPs_i$  and  $CPs_j$  of two companies  $s_i$  and  $s_j$  from the trading days  $(T - cDay_q)$  to  $(T - 1)$  according to  $cDay_q$  in chromosome  $C_q$  as

$$CPs_i = \{cp_{T-cDay_q}^{s_i}, cp_{T-cDay_q+1}^{s_i}, \dots, cp_{T-1}^{s_i}\}, \text{ and}$$

$$CPs_j = \{cp_{T-cDay_q}^{s_j}, cp_{T-cDay_q+1}^{s_j}, \dots, cp_{T-1}^{s_j}\}.$$

**STEP 2.2:** Calculate the correlation coefficient of  $s_i$  and  $s_j$  using

$$CC_{s_i s_j} = \frac{\sum_{k=T-cDay_q}^{T-1} (cp_k^{s_i} - \mu_{s_i}) (cp_k^{s_j} - \mu_{s_j})}{\sqrt{\sum_{k=T-cDay_q}^{T-1} (cp_k^{s_i} - \mu_{s_i})^2 (cp_k^{s_j} - \mu_{s_j})^2}}. \tag{10}$$

**STEP 2.3:** Repeat Steps 2.1 and 2.2 to complete the correlation coefficient matrix  $MT_{(n)}$ .

**STEP 3:** Use the following steps to select the stock pairs whose  $CC_{s_i s_j}$  is less than  $cLimit_q$  and then calculate the stock pair's entry and exit bands according to  $BBEntryWidth_q$ ,  $BBoutWidth_q$ , and  $mDay_q$  of chromosome  $C_q$ .

**STEP 3.1:** Generate the trading pair candidate set according to  $TPset = \{tp(s_i, s_j) | CC_{s_i s_j} \leq cLimit_q\}$ , where  $cLimit_q$  is the correlation coefficient threshold from chromosome  $C_q$ .

**STEP 3.2:** Obtain the closing prices  $CPs_i$  and  $CPs_j$  from trading days  $(T - mDay_q)$  to  $(T - 1)$  of both  $s_i$  and  $s_j$  of  $tp(s_i, s_j)$  in  $TPset$  as

$$CPs_i = \{cp_{T-mDay_q}^{s_i}, cp_{T-mDay_q+1}^{s_i}, \dots, cp_{T-1}^{s_i}\}, \text{ and}$$

$$CPs_j = \{cp_{T-mDay_q}^{s_j}, cp_{T-mDay_q+1}^{s_j}, \dots, cp_{T-1}^{s_j}\}.$$

**STEP 3.3:** Calculate the moving average values  $MA_i(T)$  and  $MA_j(T)$  of  $s_i$  and  $s_j$  using the closing prices generated in the previous step as

$$MA_i(T) = \frac{\sum_{k=T-mDay_q}^{T-1} cp_k^{s_i}}{mDay_q}, \text{ and } MA_j(T) = \frac{\sum_{k=T-mDay_q}^{T-1} cp_k^{s_j}}{mDay_q}.$$

**STEP 3.4:** Use the moving average value and  $BBEntryWidth_q$  to calculate the entry upper and lower bands of  $s_i$  and  $s_j$  on day  $T$  based on Formulas (1) and (2) as

$$UB_i(T) = MA_i(T) + BBEntryWidth_q * \sqrt{\frac{\sum_{k=T-mDay_q}^{T-1} (cp_k^{s_i} - \mu)^2}{mDay_q}}, \tag{11}$$

$$LB_i(T) = MA_i(T) - BBEntryWidth_q * \sqrt{\frac{\sum_{k=T-mDay_q}^{T-1} (cp_k^{s_i} - \mu)^2}{mDay_q}}, \tag{12}$$

$$UB_j(T) = MA_j(T) + BBEntryWidth_q * \sqrt{\frac{\sum_{k=T-mDay_q}^{T-1} (cp_k^{s_j} - \mu)^2}{mDay_q}}, \text{ and } \tag{13}$$

$$LB_j(T) = MA_j(T) - BBEntryWidth_q * \sqrt{\frac{\sum_{k=T-mDay_q}^{T-1} (cp_k^{s_j} - \mu)^2}{mDay_q}}. \tag{14}$$

**STEP 3.5:** Use the moving average value and  $BBoutWidth_q$  to calculate the exit upper and lower bands of  $s_i$  and  $s_j$  on day  $T$  based on Formulas (3) and (4) as

$$US_i(T) = MA_i(T) + BBoutWidth_q * \sqrt{\frac{\sum_{k=T-mDay_q}^{T-1} (cp_k^{s_i} - \mu)^2}{mDay_q}}, \quad (15)$$

$$LS_i(T) = MA_i(T) - BBoutWidth_q * \sqrt{\frac{\sum_{k=T-mDay_q}^{T-1} (cp_k^{s_i} - \mu)^2}{mDay_q}}, \quad (16)$$

$$US_j(T) = MA_j(T) + BBoutWidth_q * \sqrt{\frac{\sum_{k=T-mDay_q}^{T-1} (cp_k^{s_j} - \mu)^2}{mDay_q}}, \text{ and } (17)$$

$$LS_j(T) = MA_j(T) - BBoutWidth_q * \sqrt{\frac{\sum_{k=T-mDay_q}^{T-1} (cp_k^{s_j} - \mu)^2}{mDay_q}}. \quad (18)$$

**STEP 4:** Use the following steps to determine whether to start pairs trading.

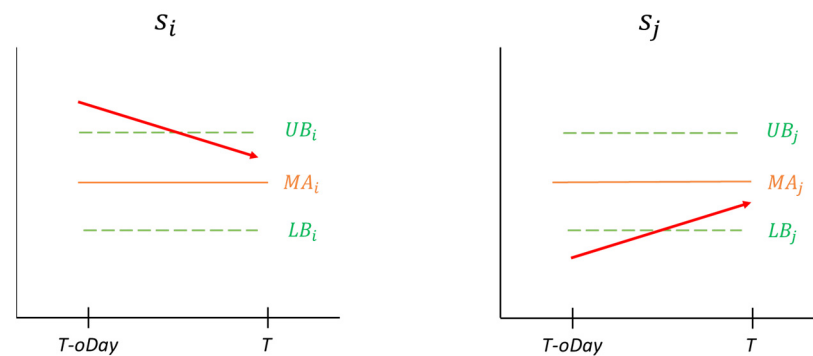
**STEP 4.1:** Determine whether all trading pairs  $tp(s_i, s_j)$  in  $TPset$  meet the following two entry conditions: Condition 1: The  $(T - oDay_q)$  closing price of the stock  $s_i$  crosses the *upper band of buy (UB)* downward on  $T$  day:

$$cp_{T-oDay_q}^{s_i} > UB_i(T) > cp_T^{s_i}.$$

Condition 2: The  $(T - oDay_q)$  closing price of the stock  $s_j$  crosses the *lower band of buy (LB)* upward on  $T$  day:

$$cp_{T-oDay_q}^{s_j} < LB_j(T) < cp_T^{s_j}.$$

when the two entry conditions are met, as shown in Figure 2, it is expected that  $s_i$  will continue to fall, and  $s_j$  will continue to rise. Hence, short  $s_i$  and long  $s_j$ .



**Figure 2.** The two entry conditions.

**STEP 4.2:** Short one unit of stock  $s_i$  and buy an integer number of stock  $s_j$  at the same cost to stock  $s_i$  when  $cp_T^{s_i} > cp_T^{s_j}$ ; otherwise, buy one unit of stock  $s_i$  and short an integer number of stock  $s_j$  at the same cost as stock  $s_j$ .

**STEP 4.3:** Record the cost  $tpC(s_i, s_j)$  of this trading pair  $tp(s_i, s_j)$ .

**STEP 4.4:** Remove  $tp(s_i, s_j)$  from  $TPset$  if the entry conditions are not met.

**STEP 4.5:** Go to Step 4.1 to determine the entry conditions of the next pair in  $TPset$ .

**STEP 5:** Use the following steps to determine whether to finish every trading pair in  $TPset$ .

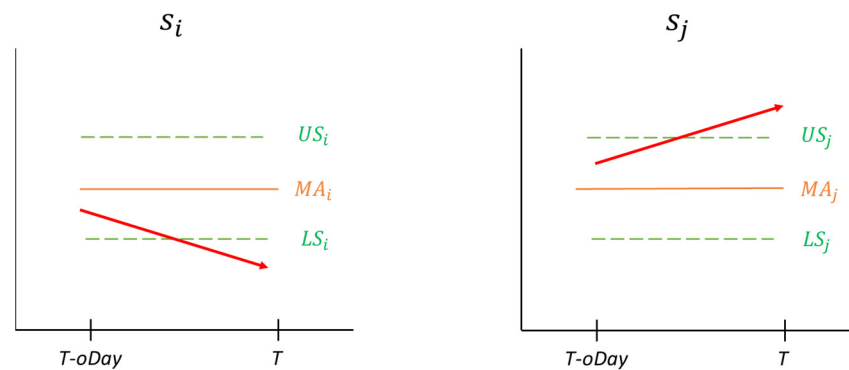
**STEP 5.1:** Determine whether all trading pairs  $tp(s_i, s_j)$  in  $TPset$  meet the following exit conditions: Condition 3: The  $(T - oDay_q)$  closing price of the stock  $s_i$  crosses the *lower band of sell (LS)* downward on the closing price at day  $T$ :

$$cp_{T-oDay_q}^{s_i} > LS_i(T) > cp_T^{s_i}.$$

Condition 4: The  $(T - oDay_q)$  closing price of the stock  $s_j$  crosses the *upper band of sell (US)* upward on the closing price at day  $T$ :

$$cp_{T-oDay_q}^{s_j} < US_j(T) < cp_T^{s_j}.$$

when the above exit conditions are satisfied, as shown in Figure 3, the transaction is closed. If it enters the market by shorting  $s_i$  and longing  $s_j$ , then it closes  $tp(s_i, s_j)$  by longing  $s_i$  and shorting  $s_j$ . Likewise, if it enters the market by longing  $s_i$  and shorting  $s_j$ , then it closes the trading pair by shorting  $s_i$  and longing  $s_j$ .



**Figure 3.** Close conditions.

**STEP 5.2:** Record the income  $tpP(s_i, s_j)$  of  $tp(s_i, s_j)$ .

**STEP 5.3:** Record the profit  $Profit_h(s_i, s_j)$  of  $tp(s_i, s_j)$ .

**STEP 5.4:** Remove  $tp(s_i, s_j)$  from  $TPset$ .

**STEP 5.5:** Record the transaction frequency of the trading pair as  $totalSell = totalSell + 1$ .

**STEP 5.6:** Go to Step 5.1 to determine the exit conditions of the next trading pair in  $TPset$ .

**STEP 6:** If the stop conditions are not met ( $T + 1 < dTotal$ ), set  $T = T + 1$  and go to Step 2 to continue the entry and exit judgment. Otherwise, go to Step 7.

**STEP 7:** Evaluate the fitness value of a chromosome by the average return and the risk of all trading pairs, as mentioned in the previous section.

**STEP 8:** Repeat Steps 2 to 7 until the fitness value of every chromosome in the population is calculated.

**STEP 9:** If the stop condition  $Generation = maxGeneration$  is met, then terminate the evolution process and goes to Step 14. Otherwise, set  $Generation = Generation + 1$  and go to Step 10.

**STEP 10:** Execute tournament selection to generate the next population.

**STEP 10.1:** Select two chromosomes randomly from the population and compare their fitness values. The chromosome with the higher fitness value is kept for the next population.

**STEP 10.2:** Repeat Step 10.1 until  $pSize$  chromosomes have been generated.

**STEP 11:** Execute MMA crossover operator with parameter  $d$  and crossover rate  $cRate$ .

**STEP 12:** Execute mutation operator to generate a new offspring with mutation rate  $mRate$ .

**STEP 13:** Go to Step 2 to evaluate the fitness of new chromosomes.

**STEP 14:** Output the chromosome with the highest fitness value as the best chromosome *bestChro*.

### 3.4. AGBCPT Example

In this section, the stock price series of the six companies in Table 4 are used as the input dataset to demonstrate AGBCPT. Each stock price series contains thirteen stock prices.

**Table 4.** Stock prices of six companies.

Stock ID	Stock Price Series
$S_{1101}$	10.5, 11, 11.25, 11.5, 12, 12.25, 13, 13.5, 13.75, 14.25, 15, 15.5, 17
$S_{1102}$	21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 27, 24, 23
$S_{2317}$	45, 44.5, 44.75, 45.5, 46, 44.5, 45.15, 45.5, 45.5, 45.25, 45.55, 45.4, 48
$S_{2412}$	71.5, 68, 66.5, 65, 63, 60, 57, 58, 56, 54, 59, 68, 69
$S_{2474}$	138.5, 135, 130, 120, 121, 115, 107, 100, 103, 98, 105, 114, 123
$S_{6505}$	10.5, 10.25, 10, 9.75, 9.5, 9, 8.5, 9, 8.5, 8, 7.75, 7.5, 7.5

The parameters used in this example are stated as follows. The population size was set at 5, the parameter for the MMA crossover operator was set at 0.7, and the crossover and mutation rates were set at 0.8 and 0.1. Below, the example is given and explained step-by-step.

**STEP 1:** The population is initialized. Since *pSize* is 5, the initial population can be randomly generated according to the encoding schema and the predefined ranges of parameters. Take  $C_1$  as an example. The six parameters are generated as  $[-0.98, 1.0, 0.5, 10, 10, 1]$ . In the same way, the initial population is formed and shown in Table 5.

**Table 5.** Initial population.

Chromosome	<i>cLimit</i>	<i>BBentryWidth</i>	<i>BBoutWidth</i>	<i>mDay</i>	<i>cDay</i>	<i>oDay</i>
$C_1$	-0.98	1.0	0.5	10	10	1
$C_2$	-0.5	1.0	0.5	8	5	1
$C_3$	-0.38	1.32	0.64	10	14	3
$C_4$	-0.7	0.71	0.52	18	17	2
$C_5$	-0.15	0.84	0.13	16	11	2

**STEP 2:** For every chromosome, the correlation coefficient matrix  $MT_{(n)}$  of the six companies is calculated by gene *cDay<sub>q</sub>* of chromosome  $C_q$ .

**STEP 2.1:** Take  $C_1$  as an example. Because the value of *cDay<sub>1</sub>* is 10, *T* starts from 11. The historical closing prices of  $S_{1101}$  and  $S_{1102}$  from trading days  $(T - 10)$  to  $(T - 1)$  are shown as

$$CPS_{1101} = \{10.5, 11, 11.25, 11.5, 12, 12.25, 13, 13.5, 13.75, 14.25\}, \text{ and}$$

$$CPS_{1102} = \{21, 22, 23, 24, 25, 26, 27, 28, 29, 30\}.$$

**STEP 2.2:** The correlation coefficient of the two companies  $CCS_{1101, S_{1102}}$  is then calculated as 0.9941.

**STEP 2.3:** Steps 2.1 and 2.2 are repeated to generate the correlation coefficient of any two companies. The resultant matrix  $MT_{(n)}$  is shown in Table 6.

**Table 6.** Correlation coefficient matrix of six companies.

Company	1101	1102	2317	2412	2474	6505
1101	1	0.9941	0.3869	−0.9772	−0.9771	−0.9598
1102		1	0.3946	−0.9861	−0.9819	−0.9708
2317			1	−0.3244	−0.423	−0.2942
2412				1	0.9723	0.9881
2474					1	0.9466
6505						1

**STEP 3:** The  $cLimit_q$  value is used to find the qualified stock pairs and  $BBentryWidth_q$ ,  $BBoutWidth_q$ , and  $mDay_q$  are used to generate the entry and exit bands.

**STEP 3.1:** Take  $C_1$  as an example. Because  $cLimit_q$  is  $-0.98$  and the  $CCS_{1102, S_{2412}}$  and  $CCS_{1102, S_{2474}}$  are  $-0.9861$  and  $-0.9819$ , meeting the condition, they are inserted into the trading pair candidate set  $TP_{set} = \{tp(S_{1102}, S_{2412}), tp(S_{1102}, S_{2474})\}$ .

**STEP 3.2:** Since  $mDay_1$  of  $C_1$  is 10, the stock price series from Day 1 ( $=11 - 10$ ) to 10 ( $=11 - 1$ ) of companies  $S_{1102}$  and  $S_{2412}$  of  $tp(S_{1102}, S_{2412})$  in  $TP_{set}$  are generated as

$$S_{1102}: \{21, 22, 23, 24, 25, 26, 27, 28, 29, 30\}, \text{ and}$$

$$S_{2412}: \{71.5, 68, 66.5, 65, 63, 60, 57, 58, 56, 54\}.$$

**STEP 3.3:** Using the 10-day moving average of  $S_{1102}$  and  $S_{2412}$  as examples, the values of  $MA_{1102}(11)$  and  $MA_{2412}(11)$  are calculated as 25.5 and 61.9.

**STEP 3.4:** The moving average value and  $BBentryWidth_1$  are used to calculate the entry upper and lower bands of  $S_{1102}$  and  $S_{2412}$  as

$$UB_{1102}(11) = 25.5 + 1 \times 3.02 = 28.52,$$

$$LB_{1102}(11) = 25.5 - 1 \times 3.02 = 22.47,$$

$$UB_{2412}(11) = 61.9 + 1 \times 5.78 = 67.68, \text{ and}$$

$$LB_{2412}(11) = 61.9 - 1 \times 5.78 = 56.11.$$

**STEP 3.5:** The moving average value and  $BBoutWidth_1$  are used to calculate the exit upper and lower bands of  $S_{1102}$  and  $S_{2412}$  as

$$US_{1102}(11) = 25.5 + 0.5 \times 3.02 = 27.01,$$

$$US_{2412}(11) = 61.9 + 0.5 \times 5.78 = 64.79,$$

$$LS_{1102}(11) = 25.5 - 0.5 \times 3.02 = 23.99, \text{ and}$$

$$LS_{2412}(11) = 61.9 - 0.5 \times 5.78 = 59.01.$$

**STEP 4:** The  $oDay_q$ , entry upper and lower bands are used to determine whether trading pair  $tp(s_i, s_j)$  in  $TP_{set}$  meet the conditions to enter the market.

**STEP 4.1:** Take trading pair  $tp(S_{1102}, S_{2412})$  as an example. It is checked to determine whether it meets the following entry conditions at trading Day  $T (=11)$ . Since  $oDay_1$  of  $C_1$  is 1, according to the stock prices on Day 10 ( $=11 - 1$ ) of  $S_{1102}$  and  $S_{2412}$ , the two conditions are shown as

$$\text{Condition 1 : } (cp_{T-oDay_1}^{S_{1102}} = 30) > (UB_{1102}(T) 28.52) > (cp_T^{S_{1102}} = 27), \text{ and}$$

$$\text{Condition 2 : } (cp_{T-oDay_1}^{S_{2412}} = 54) < (LB_{2412}(T) = 56.11) < (cp_T^{S_{2412}} = 59).$$

Since the above entry conditions are met, it is expected that  $S_{1102}$  will continue to fall and  $S_{2412}$  will continue to rise; the pairs trading strategy then shorts  $S_{1102}$  and longs  $S_{2412}$ . The conditions are shown in Figure 4.

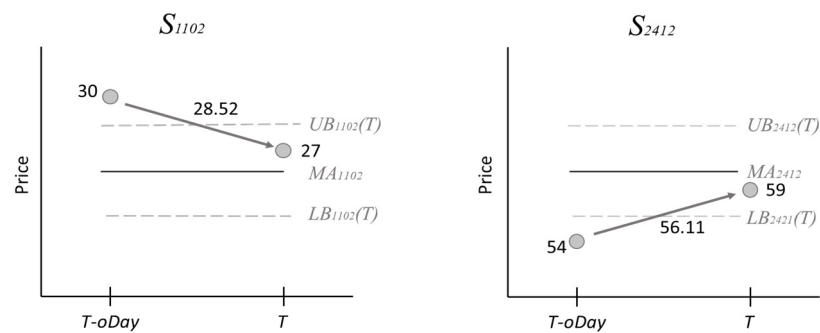


Figure 4. Entry conditions of  $S_{1102}$  and  $S_{2412}$ .

**STEP 4.2:** Because the closing price on trading Day 11 of  $cp_T^{S_{2412}}$  (59) is greater than  $cp_T^{S_{1102}}$  (27), their ratio is rounded to 2. Hence, the trading strategy then longs one unit of  $S_{2412}$  and shorts two units of  $S_{1102}$ . Thus, their investment capital is nearly equal.

**STEP 4.3:** The cost of this trading pair  $tpC(1102, 2412)$  is then recorded. The cost of longing  $S_{2412}$  is 59,084 ( $=59 \times 1000 \times 1 \times (1 + 0.001425)$ ), the cost of shorting  $S_{1102}$  is 54,239 ( $=27 \times 1000 \times 2 \times (1 + 0.004425)$ ), and  $tpC(S_{1102}, S_{2412})$  is 113,323 ( $=59,084 + 54,239$ ).

**STEP 4.4:** If the entry condition of  $tp(s_i, s_j)$  is not met, then it is removed from  $TPset$ .

**STEP 4.5:** Steps 4.1 to 4.4 are repeated to determine the entry condition of every pair  $tp(s_i, s_j)$  in  $TPset$  until all pairs have been checked.

**STEP 5:** The upper and lower exit bands are then used to determine whether  $tp(S_{1102}, S_{2412})$  has met the conditions to exit the market.

**STEP 5.1:** Take trading pair  $tp(S_{1102}, S_{2412})$  as an example. It is checked to determine whether it meets the following exit conditions on trading Day 12. Since  $oDay_1$  of  $C_1$  is 1, according to the stock prices on Day 11 ( $=12 - 1$ ) of both  $S_{1102}$  and  $S_{2412}$  of  $tp(S_{1102}, S_{2412})$ , the two conditions are

$$\text{Condition 1 : } (cp_{T-oDay_1}^{S_{1102}} = 27) > (LS_{1102}(T) = 24.79) > (cp_T^{S_{1102}} = 24), \text{ and}$$

$$\text{Condition 2 : } (cp_{T-oDay_1}^{S_{2412}} = 59) < (US_{2412}(T) = 62.96) < (cp_T^{S_{2412}} = 68).$$

when the above exit conditions are met, the trading strategy then longs  $S_{1102}$  and shorts  $S_{2412}$ , as shown in Step 5.2: Longing stock  $S_{1102}$  and shorting  $S_{2412}$  yields 48,068 ( $=24 \times 1000 \times 2 \times (1 + 0.001425)$ ) and 68,301 ( $=68 \times 1000 \times 1 \times (1 + 0.004425)$ ). The trading result of trading pair  $tpP(S_{1102}, S_{2412})$ , which is 15,388 ( $=(54,239 - 48,068) + (68,301 - 59,084)$ ), is then recorded.

**STEP 5.3:** The profit of the trading pair,  $profit_{(1102,2412)}$ , is then recorded as

$$profit_{(1102,2412)} = \frac{tpP_{(S_{1102}, S_{2412})}}{tpC_{(S_{1102}, S_{2412})}} = \frac{15,388}{113,323} = 0.1357.$$

**STEP 5.4:** Pair  $tp(S_{1102}, S_{2412})$  is removed from  $TPset$ .

**STEP 5.5:** The number of transactions is set to  $totalSell = totalSell + 1$ .

**STEP 5.6:** Steps 5.1 to 5.5 are repeated to determine the exit condition of the next pair  $tp(s_i, s_j)$  in  $TPset$  until all the pairs have been checked.

**STEP 6:** The trading stop conditions are checked. If stop condition ( $T + 1 < dTotal$ ) is not met, then  $T$  is set to  $T + 1$ , Step 2 is executed, and the entry and exit judgment is continued for the next trading day. Otherwise, Step 7 is executed.

**STEP 7:** Because the three profits of the trading pair are calculated as 13.57%, 10.15%, and 1%, the risk of the trading pair is 1%, which is the minimum value of the



three profits. The fitness value of  $C_1$  is then calculated by the total profit and risk of the trading pair as 24.72 (=24.72%/1%).

**STEP 8:** Steps 2 to 7 are repeated to calculate the fitness values of all chromosomes, yielding the results shown in Table 7.

**Table 7.** Fitness values of chromosomes.

$Cq$	Fitness Value
$C_1$	24.72
$C_2$	10.15
$C_3$	0.5
$C_4$	6.2
$C_5$	23.01

**STEP 9:** If the stop condition is met, Step 14. Otherwise, Step 10 is executed to generate next population.

**STEP 10:** Tournament selection is used to generate the next population.

**STEP 10.1:** Take the two chromosomes shown in Table 8 as an example. Because the fitness value of  $C_1$  is greater than  $C_4$ ,  $C_1$  is retained for the next population.

**Table 8.** Chromosomes for tournament selection.

$Cq$	Fitness Value
$C_1$	24.72
$C_4$	6.2

**STEP 10.2:** Step 10.1 is repeated until the number of chromosomes is equal to 5.

**STEP 11:** The MMA crossover operator is applied to generate offspring. The MMA parameter  $d$  and the crossover rate  $cRate$  are set to 0.7 and 0.8. For every two chromosomes, four new chromosomes are generated as candidate offspring. Take chromosomes  $C_1$  and  $C_2$  as an example. After crossover, the final offspring are shown in Table 9.

**Table 9.** Four new generated chromosomes.

Chromosome	$cLimit$	$BBentryWidth$	$BBoutWidth$	$mDay$	$cDay$	$oDay$
$Cmax$	−0.5	1.0	0.5	10	10	1
$Cmin$	−0.98	1.0	0.5	8	5	1
$Cnew1$	−0.644	1.0	0.5	9	7	1
$Cnew2$	−0.836	1.0	0.5	9	9	1

**STEP 12:** The one-point mutation operator is executed to generate new offspring according to the mutation rate.

**STEP 13:** After executing the crossover and mutation operators, Steps 2 to 8 are used to calculate the fitness value of the new chromosomes.

**STEP 14:** The chromosome with the highest fitness value is outputted. In this example, according to Table 7,  $C_1$ : [−0.98, 1.0, 0.5, 10, 10, 1] is selected and outputted as the parameters for trading.

#### 4. Experimental Results and Discussion

In this section, we describe experiments conducted to show the effectiveness of the proposed approach, and discuss the results. The experimental dataset consisted of companies selected from the Taiwan stock exchange (TSE). Companies with stock price series from the top 50 companies in the Taiwan stock market were selected. The dataset contained stock price series from 1 January 2009 to 31 December 2020. The stock price series are shown in Figure 5.

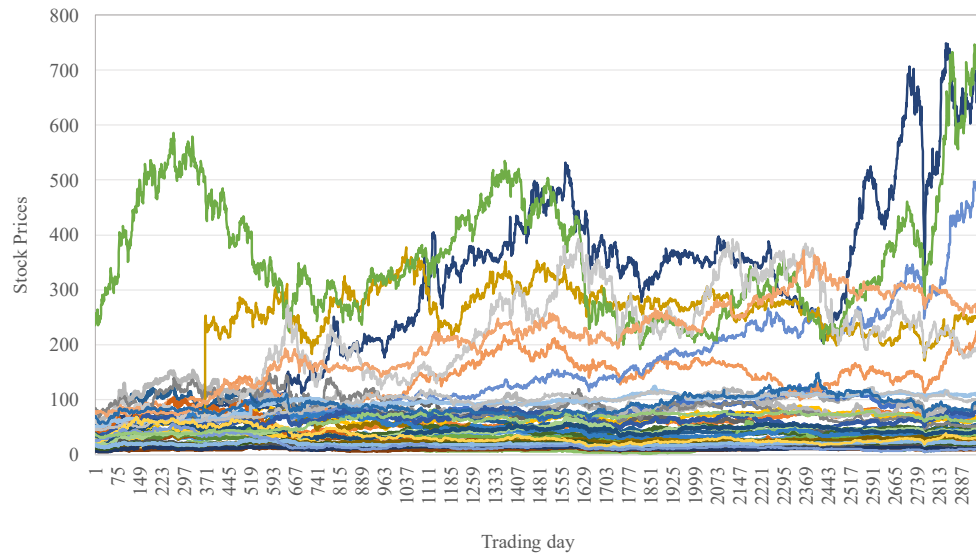


Figure 5. Stock price series of dataset.

In Figure 5, most stock prices fall between 0 and 100, with some between 100 and 400; only three exceed 400. In addition to the stock price series, the correlation coefficient distribution between companies with *cDay* set at 20 is shown in Figure 6.

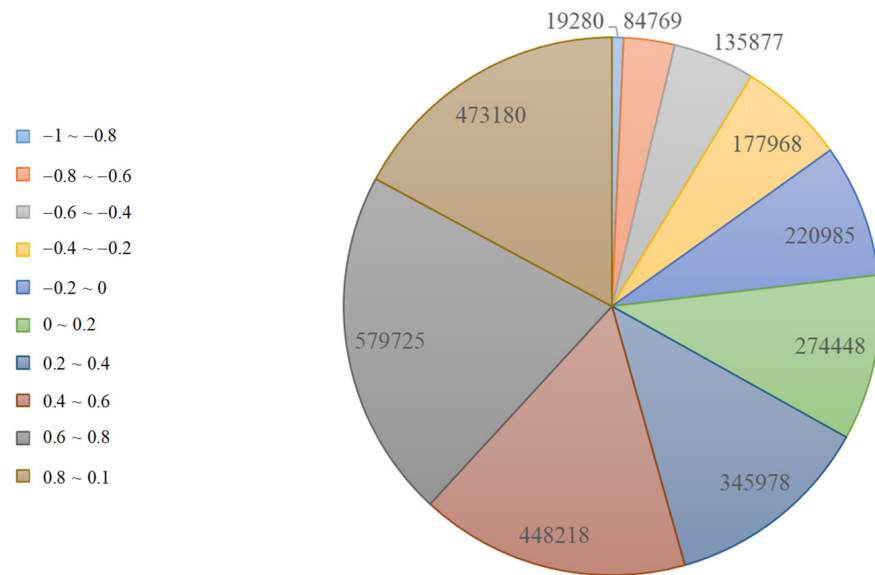


Figure 6. Distribution of correlation coefficients between companies.

In Figure 6, the ratio of the numbers of stock pairs with positive and negative correlation coefficients is 3.32 (=2,121,549/638,879), which means that the number of stock pairs with negative correlation coefficients is smaller than that with positive ones. Note that the correlation coefficient distribution may be affected by *cDay* and the period of the dataset.

To show the effectiveness of the proposed approach, we conducted three experiments concerning the following: (1) The impact of the three new parameters to the pairs trading strategy; (2) the impact of the proposed approach under different stock trends; (3) a comparison of AGBCPT and GBCPT.

#### 4.1. Impact of Three New Parameters on Pairs Trading Strategy

To observe their impacts on the pairs trading strategy, we adjusted the three new parameters: the correlation coefficient calculation days (*cDay*), the moving average calculation days (*mDay*), and the forward observation days (*oDay*). In the experiments, we adjusted one parameter at a time, while using the default values for the others. The default values for *cDay*, *mDay*, and *oDay* were 10, 10 and 1, respectively, and the values of *cLimit*, *BBentryWidth* and *BBoutWidth* were set to  $-0.73$ , 2.3 and 1.5.

The experiments made on the four datasets are shown below: (1) experiments on the four-year dataset, from 2009 to 2012, shown in Figure 7; (2) experiments on the three-year dataset, from 2010 to 2012, shown in Figure 8; (3) experiments on the two-year dataset, from 2011 to 2012, shown in Figure 9; and (4) experiments on the one-year (2012) dataset, shown in Figure 10.

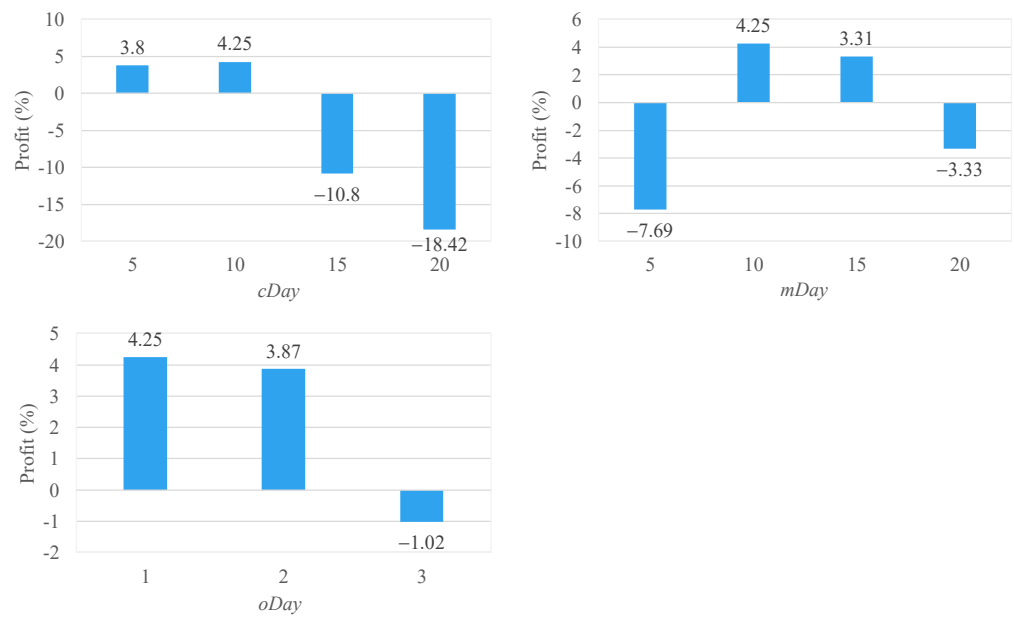


Figure 7. Profits from different parameter settings on four-year dataset (2009–2012).

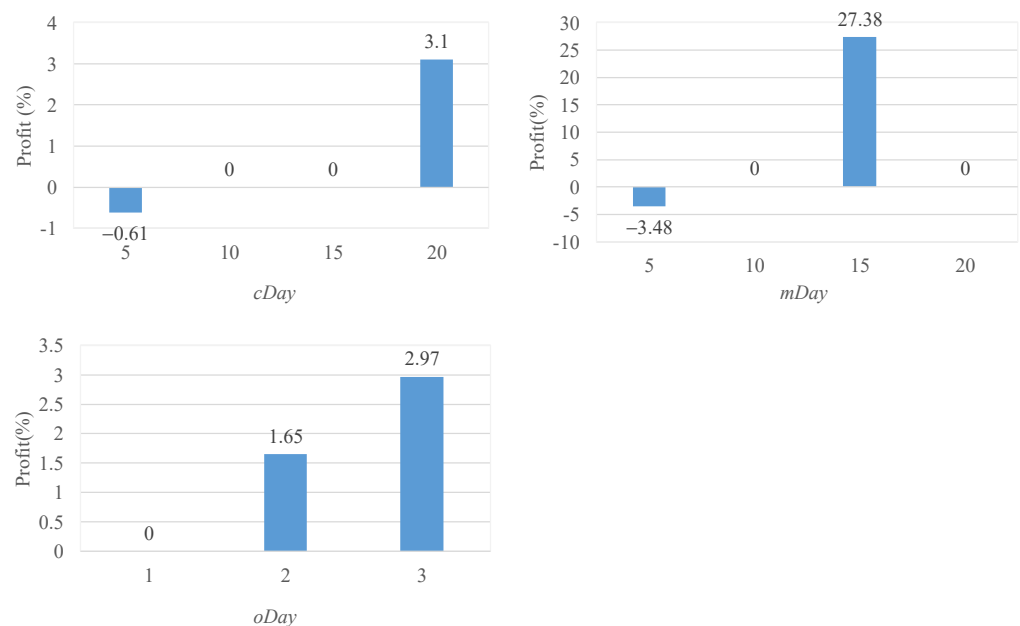


Figure 8. Profits from different parameter settings on three-year dataset (2010–2012).

Figure 7 shows that when  $cDay$  is set to 5 and 10, the profits of the pairs trading strategy yield a positive profit, with the best profit at 4.25%. When  $cDay$  is set to 15 and 20, the pairs trading strategy yields a negative profit, with the worst profit at  $-18.42\%$  when  $cDay$  is set to 20. For the parameter  $mDay$ , the pairs trading strategy yields positive profits when  $mDay$  is set to 10 and 15. When  $mDay$  is set to 5 and 20, it yields negative profits, with the worst profit at  $-7.69\%$  when  $mDay$  is set to 5. The experimental results for parameter  $oDay$  show that the best profit is 4.25% when  $oDay$  is set to 1, and the profit becomes worse when  $oDay$  increases. When  $oDay$  is set to 3, the profit is negative. Hence, for long training periods, the suggested parameter setting is 10, 10 and 1 for  $cDay$ ,  $mDay$ , and  $oDay$ .

In Figure 8, when  $cDay$  is set to 5, 10 and 15, the pairs trading strategy yields no profits, or negative profits, on the three-year dataset. When  $cDay$  is set to 20, the profit is best at 3.1%. For parameter  $mDay$ , the pairs trading strategy yields the best profit at 27.38% when  $mDay$  is set to 15. For parameter  $oDay$ , the profit increases with increases in  $oDay$ ; the best profit at 2.97% when  $oDay$  is set to 3. As a result, for the three-year training period, the suggested parameter setting is 20, 15 and 3 for  $cDay$ ,  $mDay$ , and  $oDay$ .

From Figure 9, we see that in the two-year dataset, the worst profit is  $-5.62\%$  when  $cDay$  is set to 5. For parameter  $mDay$ , the best profit is 13.32% when  $mDay$  is set to 13.32%, and the profit is 0, 1.25, and 3.61 when  $mDay$  is set to 10, 15, and 20, respectively. For  $oDay$ , the profit increases while the set of  $oDay$  becomes larger; the best profit is 4.83% when  $oDay$  is set to 3. Hence, for the two-year training period, the suggested parameter setting is 5 and 3 for  $mDay$  and  $oDay$ . For  $cDay$ , however, additional experiments are needed to determine a suitable setting.

The results of Figure 10 are as follows. For parameter  $cDay$ , the best profit is 6.97% when it is set to 10. When  $cDay$  is set to 5, 15, 20, the profits are around zero. For parameter  $mDay$ , the positive profits are 1.82%, 6.97%, and 3.41% when  $mDay$  is set to 5, 10, and 15, respectively. In addition, the best profit is 6.97% when  $oDay$  is set to 1, and the worst is  $-0.93\%$  when  $oDay$  is set to 3. Thus, for short training periods, the suggested parameter setting is 10, 10, and 1 for  $cDay$ ,  $mDay$ , and  $oDay$ .

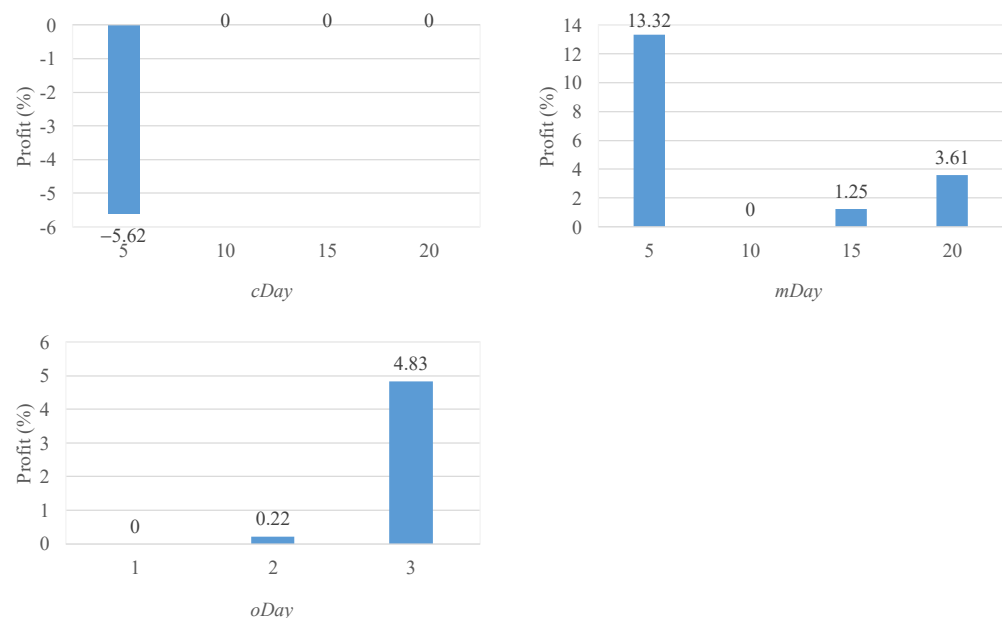


Figure 9. Profits of different parameter settings on two-year dataset (2011–2012).

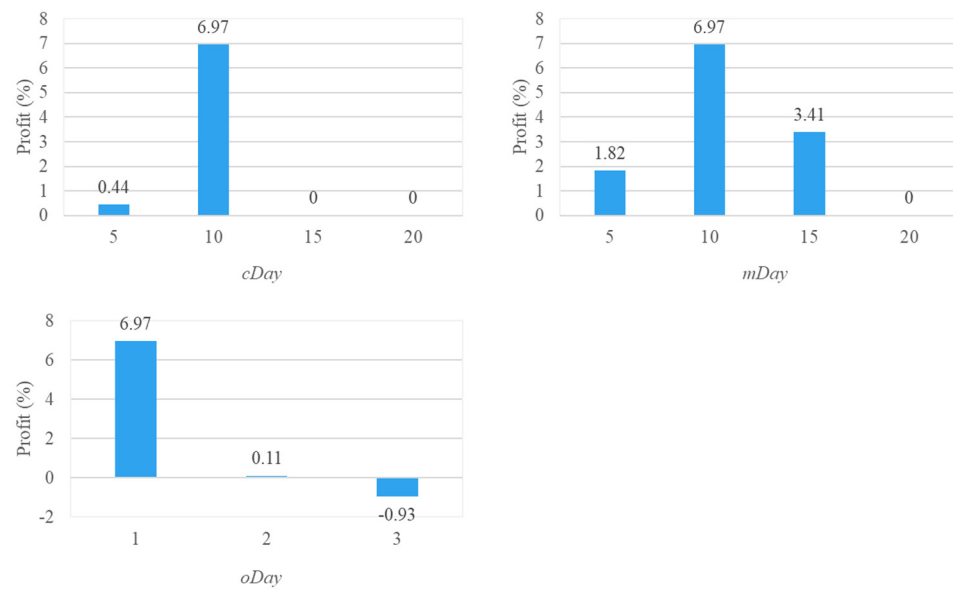


Figure 10. Profit of different parameter settings on one-year dataset (2012).

Notably, the results show that various parameter settings influence the profit of the pairs trading model. That is, determining suitable parameters for the pairs trading strategy is a difficult task and constitutes an optimization problem. We thus use AGBCPT to determine parameters that yield better performance for the pairs trading strategy.

#### 4.2. Impact of AGBCPT under Different Stock Trends

To show the effectiveness of the proposed approach on different trends, we conducted experiments using different stock trends as the testing datasets, including upward-trend, correction-trend, and downward-trend datasets. The buy-and-hold method (BAH) was used in comparison with AGBCPT. BAH is executed as the following steps. It buys all of the stocks on the first trading day and sells them all on the last trading day, after which the profit of the transactions is calculated. As shown in Figure 11, the datasets used in this experiment were selected correspond to Taiwan stock market trends.



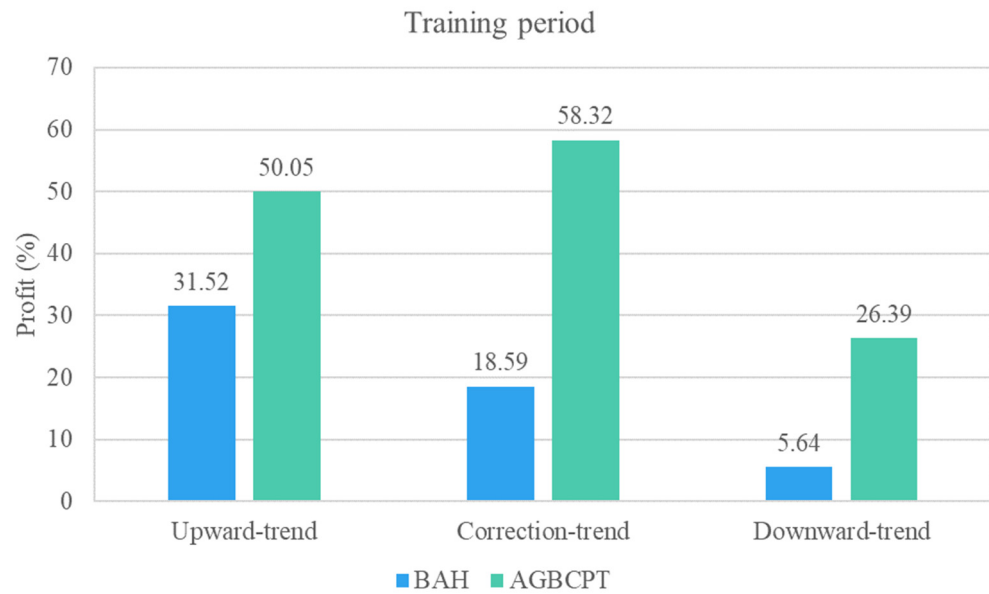
Figure 11. Taiwan stock market trends.

The trends were chosen as the following testing intervals: (1) 2020 was selected as the upward-trend dataset, (2) 2012 is selected was the correction-trend dataset, and (3) 2015 was selected as the downward-trend dataset. According to the trend periods, three training and testing periods are shown in Table 10.

**Table 10.** Training and testing periods of three trends.

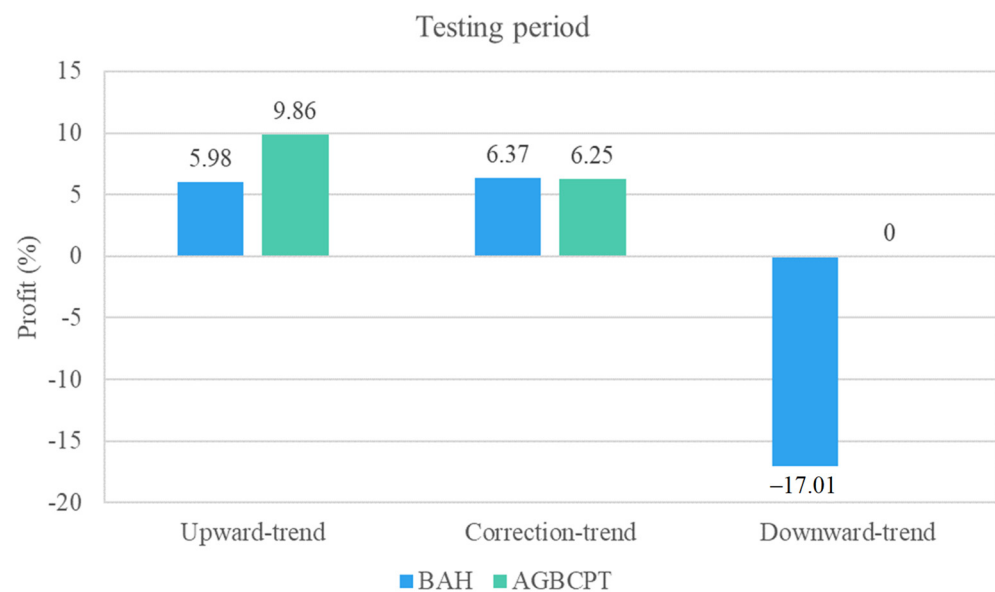
Market Trend	Training Period	Testing Period
Upward trend	2016–2019	2020
Correction trend	2009–2011	2012
Downward trend	2011–2014	2015

The training results of the AGBCPT and BAH methods are shown in Figure 12.



**Figure 12.** AGBCPT and BAH training results.

From Figure 12, in the three training periods, we observe that both methods yield positive profits. The profits of the three trends with AGBCPT are 50.05%, 58.32%, and 26.39%, which are all better than that with BAH. Based on the trained results, their profits on the testing datasets are shown in Figure 13.



**Figure 13.** AGBCPT and BAH testing results.



From Figure 13, we observe that in the testing phase, the results of the upward-trend dataset show that the 9.86% profit of AGBCPT is better than that of BAH (5.98%). In the correction-trend dataset, the results show similar profits for both AGBCPT and BAH: the profit of AGBCPT is 6.25%, and that of BAH is 6.37%. For the downward-trend dataset, AGBCPT yields no profit in the testing period (0%). However, compared with the BAH, AGBCPT is better than BAH because the profit of BAH is  $-17.01\%$ . This shows that AGBCPT reduces risk on a downward-trend dataset.

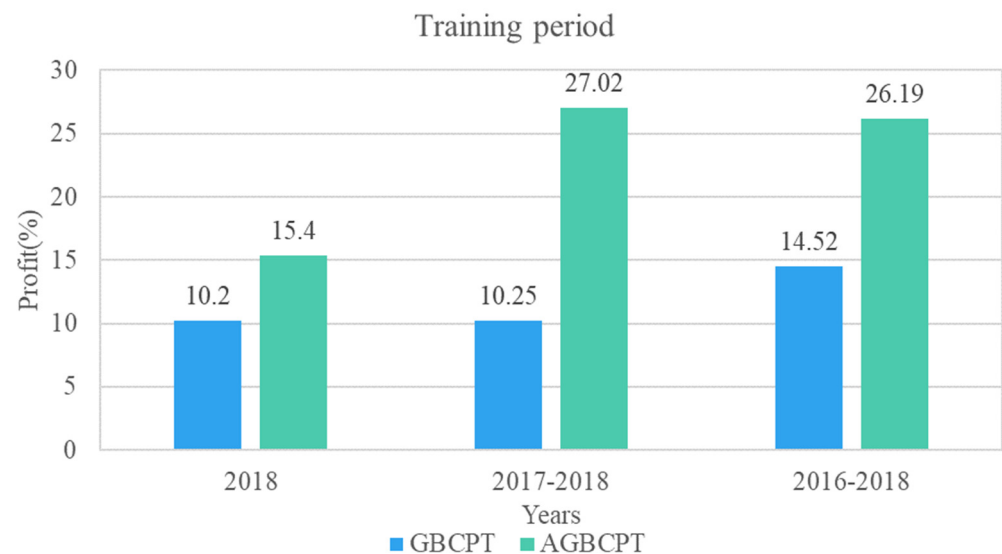
#### 4.3. Comparison of AGBCPT and GBCPT

In this section, we compare the proposed AGBCPT method with the previous GBCPT method [28]. Table 11 shows the datasets used in the experiments. They are: (1) the three-year training period (2016–2018), (2) the two-year training period (2017–2018), and (3) the one-year training period (2018). The testing period of these training periods is 2019.

**Table 11.** Training and testing periods for comparison.

Training Period	Testing Period
2016–2018	2019
2017–2018	2019
2018	2019

The profits of AGBCPT and GBCPT in the training phase are shown in Figure 14.



**Figure 14.** AGBCPT and GBCPT profits in training phase.

In Figure 14, the AGBCPT and GBCPT profits are positive in the training phase, with AGBCPT achieving higher profits than GBCPT. Based on the trained results, the profits on the testing datasets are shown in Figure 15.

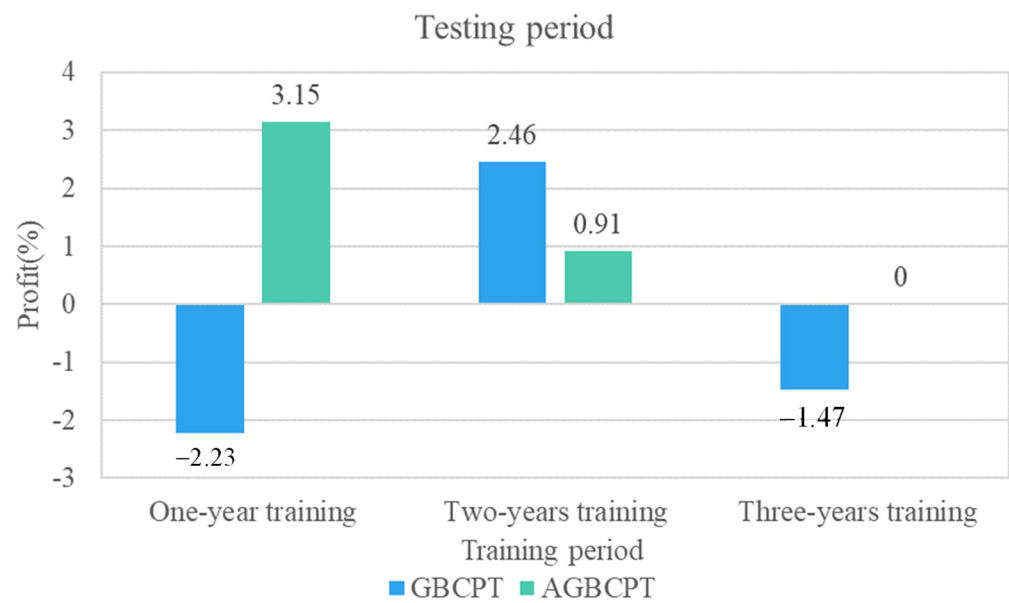


Figure 15. AGBCPT and GBCPT profits in testing phase.

Figure 15 shows that the AGBCPT and GBCPT profits using the three trained models are 3.15%, 0.91%, and 0%, and -2.33%, 2.46%, and -1.47%, respectively. In addition, the GBCPT profits are negative on the one-year and three-year datasets. However, the AGBCPT profits are positive. Thus, we conclude that the fitness function in the AGBCPT method, which accounts for risk, reduces the possibility of negative profits in the testing period. Next, to show the profitability of AGBCPT, experiments were conducted on the one-year training dataset (2015) and the one-year (2016), two-year (2016–2017), and three-year (2016–2018) testing datasets; the results on the training period are shown in Figure 16.

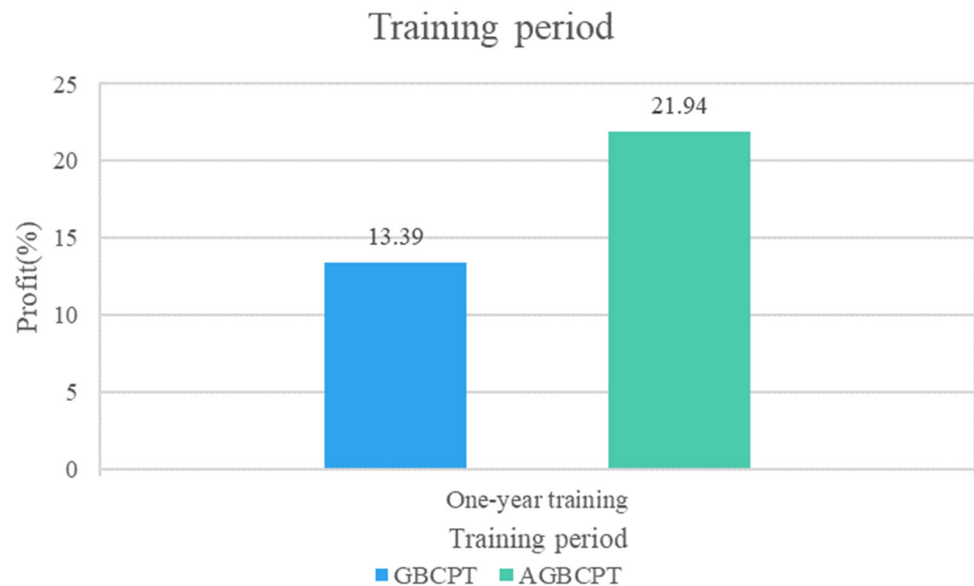
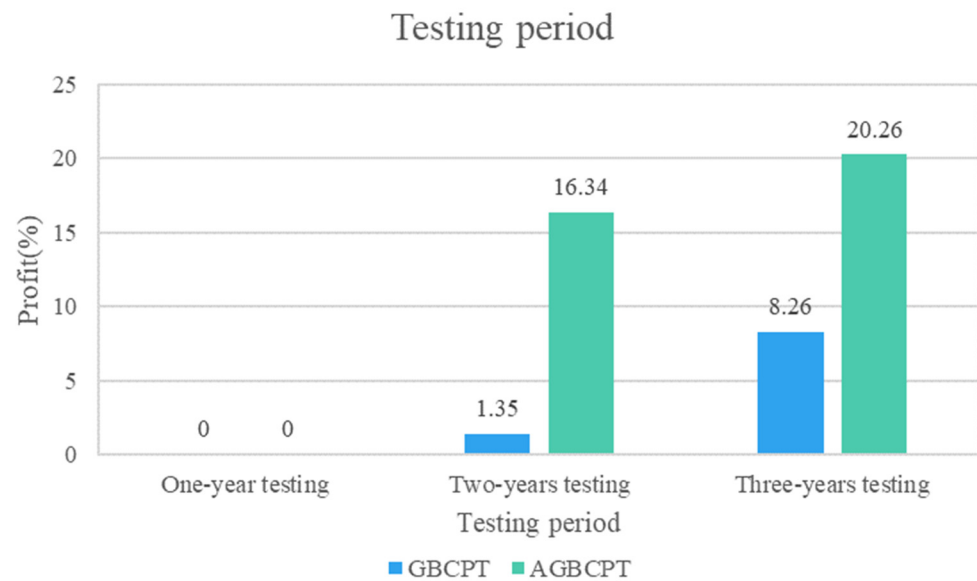


Figure 16. AGBCPT and GBCPT profits in training period.

In Figure 16, the 21.9% profit of AGBCPT is better than the 13.39% of CBCPT. Figure 17 compares the two approaches on the testing periods in terms of profit.



**Figure 17.** AGBCPT and GBCPT profits in different testing periods.

Figure 17 shows that in the one-year testing period, neither method yields a profit. In the two-year testing period, the 16.34% profit of AGBCPT is better than the 1.35% of GBCPT. In the three-year testing period, the 20.26% profit of AGBCPT is better than the 8.26% of GBCPT. From the experimental results, we conclude that AGBCPT is effective and profitable for middle-long-term trading.

#### 4.4. Discussion

In this section, we discuss how to improve the efficiency of the proposed AGBCPT method, how to make the derived trading strategy more profitable and stable, and the applications of AGBCPT.

For the first issue, the main differences between AGBCPT, the proposed approach, and GBCPT, the compared approach, are the encoding schema and the fitness function. In AGBCPT, six parameters are encoded into a chromosome, and the return and risk are jointly considered to evaluate the fitness value of a chromosome. In GBCPT, three parameters are used, and every chromosome is evaluated only by return. Hence, execution times for AGBCPT are slightly longer than those for GBCPT. Taking AGBCPT as an example, the execution time is 43,741 s with a population size of 60 and 40 stocks, which is time-consuming. The efficiency of AGBCPT can be improved via soft computing techniques or hardware devices. For example, chromosomes could be clustered into groups. For every group, the fitness value of a selected representative chromosome could be calculated and used as the fitness value of the other chromosomes in the same group. By thus using  $k$ -means clustering, only  $k$  chromosomes are selected to calculate fitness values, resulting in reduced time costs. Alternatively, the graphics processing unit (GPU) could be utilized to speed up data calculations.

For the second issue, in the proposed approach, the correlation coefficient of stocks and Bollinger Bands are used to identify trading pairs and signals. However, other factors could be considered to increase profitability and reduce risk. For instance, company fundamentals, e.g., earnings per share, or the P/E ratio, could be used as a filter to avoid high risk stocks. In addition, industrial information could be considered by using the correlation coefficient of industries to identify relationships between industries, yielding more profitable and stable trading pairs.

As to the applications of AGBCPT, along with the popularity of program trading in recent years, AGBCPT could be enclosed as a module for providing programmers to design trading procedure which can generate trading signals automatically for trading. Besides, for securities company, from customer relationship management point of view, AGBCPT

can be embed in their trading system as a function for providing more information to users, which may increase customer loyalty.

## 5. Conclusions and Future Work

Trading strategies are commonly used approaches for finding buying or selling signals for trading. One type of trading strategy is the pairs trading strategy. In the past, parameters in pairs trading strategies are usually set through experience, which is typically time-consuming. In this paper, negative correlation coefficient trading pairs, genetic algorithms, and Bollinger Bands are considered in AGBCPT, the proposed advanced genetic Bollinger Bands and correlation-coefficient based pairs trading algorithm, to determine the appropriate parameters for the long-short pairs trading strategy. To verify the effectiveness of AGBCPT, experiments were conducted on real datasets, showing that the parameters considered in pairs trading do affect the profitability of the pairs trading strategy; AGBCPT profit is superior to that of BAH and GBCPT for three stock market trends on various training and testing periods; and the fitness function used in AGBCPT also outperforms that of the previous approach in terms of reducing the trading risk of the trained model. Besides, AGBCPT can also be used as a module or function for securities company for providing more information to users to increase customer loyalty. In the future, we will enhance the proposed approach in the following directions: (1) by enhancing the pairs trading optimization algorithm by adding more stocks to the dataset to identify more profitable potential pairs; (2) by using other algorithms in pairs trading strategies to determine better parameter settings for more complex financial problems; (3) by utilizing statistical tests to verify whether AGBCPT is significantly better than existing approaches, or comparing with other pairs trading algorithms to identify the merits of AGBCPT; and (4) by considering industry relations among stocks to classify stocks as groups, generating more profitable trading pairs.

**Author Contributions:** Conceptualization, C.-H.C. and T.-P.H.; methodology, C.-H.C.; formal analysis, T.-P.H. and C.-H.C.; data curation, W.-H.L.; writing—original draft preparation, C.-H.C., W.-H.L. and S.-T.H.; writing—review and editing, C.-H.C. and T.-P.H.; supervision, T.-P.H.; funding acquisition, T.-P.H. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported by the Ministry of Science and Technology of the Republic of China under Grant MOST 109-2221-E-390-015-MY3.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Chang, H.-H.; Dai, T.-S.; Wang, K.-L.; Chu, C.-H.; Wang, J.-Z. Improving pair trading performances with structural change detections and revised trading strategies. In Proceedings of the 2020 International Conference on Pervasive Artificial Intelligence (ICPAI), Taipei, Taiwan, 3–5 December 2020; pp. 105–109.
2. Ding, W.; Mazouz, K.; Wang, Q. Volatility timing, sentiment, and the short-term profitability of VIX-based cross-sectional trading strategies. *J. Empir. Financ.* **2021**, *63*, 42–56. [[CrossRef](#)]
3. Prasetijo, A.; Saputro, T.A.; Windasari, I.P.; Windarto, Y.E. Buy/sell signal detection in stock trading with Bollinger Bands and parabolic SAR: With web application for proofing trading strategy. In Proceedings of the 2017 4th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE), Semarang, Indonesia, 18–19 October 2017; pp. 41–44.
4. Wu, X.; Chen, H.; Wang, J.; Troiano, L.; Loia, V.; Fujita, H. Adaptive stock trading strategies with deep reinforcement learning methods. *Inf. Sci.* **2020**, *538*, 142–158.
5. Wu, M.-E.; Syu, J.-H.; Lin, J.C.-W.; Ho, J.-M. Effective fuzzy system for qualifying the characteristics of stocks by random trading. *IEEE Trans. Fuzzy Syst.* **2022**. [[CrossRef](#)]
6. Wu, J.M.-T.; Sun, L.; Srivastava, G.; Lin, J.C.-W. A long short-term memory network stock price prediction with leading indicators. *Big Data* **2021**, *9*, 343–357. [[PubMed](#)]

7. Zhang, W.; Wang, L.; Xie, L.; Feng, K.; Liu, X. TradeBot: Bandit learning for hyper-parameters optimization of high frequency trading strategy. *Pattern Recognit.* **2021**, *124*, 108490. [[CrossRef](#)]
8. Cocco, L.; Tonelli, R.; Marchesi, M. An agent-based artificial market model for studying the bitcoin trading. *IEEE Access* **2019**, *7*, 42908–42920. [[CrossRef](#)]
9. Ferreira, F.G.D.C.; Gandomi, A.H.; Cardoso, R.T.N. Artificial intelligence applied to stock market trading: A review. *IEEE Access* **2021**, *9*, 30898–30917.
10. Jirapongpan, R.; Phumchusri, N. Prediction of the Profitability of Pairs Trading Strategy Using Machine Learning. In Proceedings of the 2020 IEEE 7th International Conference on Industrial Engineering and Applications (ICIEA), Bangkok, Thailand, 16–21 April 2020; pp. 1025–1030.
11. Mochón, A.; Quintana, D.; Sáez, Y.; Isasi, P.; Mochón, M.A. Soft computing techniques applied to finance. *Appl. Intell.* **2007**, *29*, 111–115. [[CrossRef](#)]
12. Stadnik, B. Interest rates sensitivity arbitrage—Theory and practical assessment for financial market trading. *J. Bus. Manag. Econ. Eng.* **2021**, *19*, 12–23. [[CrossRef](#)]
13. Chen, S.; Zhou, C. Stock prediction based on genetic algorithm feature selection and long short-term memory neural network. *IEEE Access* **2021**, *9*, 9066–9072. [[CrossRef](#)]
14. Kim, K.-J.; Han, I. Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. *Expert Syst. Appl.* **2000**, *19*, 125–132. [[CrossRef](#)]
15. Patel, J.; Shah, S.; Thakkar, P.; Kotecha, K. Predicting stock market index using fusion of machine learning techniques. *Expert Syst. Appl.* **2015**, *42*, 2162–2172. [[CrossRef](#)]
16. Shen, J.; Shafiq, M.O. Short-term stock market price trend prediction using a comprehensive deep learning system. *J. Big Data* **2020**, *7*, 66. [[CrossRef](#)]
17. Chen, C.-H.; Chen, Y.-H.; Diaz, V.G.; Lin, J.C.-W. An intelligent trading mechanism based on the group trading strategy portfolio to reduce massive loss by the grouping genetic algorithm. *Electron. Commer. Res.* **2021**. [[CrossRef](#)]
18. Chen, C.-H.; Lu, C.-Y.; Hong, T.-P.; Lin, J.C.-W.; Gaeta, M. An effective approach for the diverse group stock portfolio optimization using grouping genetic algorithm. *IEEE Access* **2019**, *7*, 155871–155884. [[CrossRef](#)]
19. Lim, S.; Kim, M.-J.; Ahn, C.W. A genetic algorithm (GA) approach to the portfolio design based on market movements and asset valuations. *IEEE Access* **2020**, *8*, 140234–140249. [[CrossRef](#)]
20. Bowen, D.A.; Hutchinson, M.C. Pairs trading in the UK equity market: Risk and return. *Eur. J. Financ.* **2016**, *22*, 1363–1387. [[CrossRef](#)]
21. Elliott, R.J.; Hoek, J.V.D.; Malcolm, W.P. Pairs trading. *Quant. Financ.* **2005**, *5*, 271–276. [[CrossRef](#)]
22. Flori, A.; Regoli, D. Revealing pairs-trading opportunities with long short-term memory networks. *Eur. J. Oper. Res.* **2021**, *295*, 772–791. [[CrossRef](#)]
23. Krauss, C. Statistical arbitrage pairs trading strategies: Review and outlook. *J. Econ. Surv.* **2016**, *31*, 513–545. [[CrossRef](#)]
24. Sarmento, S.M.; Horta, N. Enhancing a pairs trading strategy with the application of machine learning. *Expert Syst. Appl.* **2020**, *158*, 113490. [[CrossRef](#)]
25. Gatev, E.G.; Goetzmann, W.N.; Rouwenhorst, K.G. Pairs trading: Performance of a relative-value arbitrage rule. *Rev. Financ. Stud.* **2006**, *19*, 797–827. [[CrossRef](#)]
26. Fil, M.; Kristoufek, L. Pairs trading in cryptocurrency markets. *IEEE Access* **2020**, *8*, 172644–172651. [[CrossRef](#)]
27. Oh, E.; Son, S.-Y. Pair matching strategies for prosumer market under guaranteed minimum trading. *IEEE Access* **2018**, *6*, 40325–40333. [[CrossRef](#)]
28. Huang, C.C. Correlation-Based Pair Trading Optimization Techniques. Master's Thesis, Department of Computer Science and Information Engineering, Tamkang University, Taipei, Taiwan, 2020.
29. Shen, L.; Shen, K.; Yi, C.; Chen, Y. An evaluation of pairs trading in commodity futures markets. In Proceedings of the 2020 IEEE International Conference on Big Data (Big Data), Atlanta, GA, USA, 10–13 December 2020; pp. 5457–5462.
30. Clegg, M.; Krauss, C. Pairs trading with partial cointegration. *Quant. Financ.* **2018**, *18*, 121–138. [[CrossRef](#)]
31. Do, B.; Faff, R. Does simple pairs trading still work? *Financ. Anal. J.* **2010**, *66*, 83–95. [[CrossRef](#)]
32. Do, B.; Faff, R. Are pairs trading profits robust to trading costs? *J. Financ. Res.* **2012**, *35*, 261–287. [[CrossRef](#)]
33. Liang, S.; Lu, S.; Lin, J.; Wang, Z. Low-latency hardware accelerator for improved engle-granger cointegration in pairs trading. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2021**, *68*, 2911–2924. [[CrossRef](#)]
34. Liu, B.; Chang, L.-B.; Geman, H. Intraday pairs trading strategies on high frequency data: The case of oil companies. *Quant. Financ.* **2017**, *17*, 87–100. [[CrossRef](#)]
35. Rad, H.; Low, R.K.Y.; Faff, R. The profitability of pairs trading strategies: Distance, cointegration and copula methods. *Quant. Financ.* **2016**, *16*, 1541–1558. [[CrossRef](#)]
36. Ramos-Requena, J.; Trinidad-Segovia, J.; Sánchez-Granero, M. Introducing Hurst exponent in pair trading. *Phys. A Stat. Mech. Its Appl.* **2017**, *488*, 39–45. [[CrossRef](#)]
37. Jacobs, H.; Weber, M. On the determinants of pairs trading profitability. *J. Financ. Mark.* **2015**, *23*, 75–97. [[CrossRef](#)]
38. Rende, J. Pairs trading with the persistence-based decomposition model. *Manag. Econ.* **2020**, *20*, 151. [[CrossRef](#)]
39. Stäbinger, J.; Endres, S. Pairs trading with a mean-reverting jump-diffusion model on high-frequency data. *Quant. Financ.* **2018**, *18*, 1735–1751. [[CrossRef](#)]

40. Fallahpour, S.; Hakimian, H.; Taheri, K.; Ramezanifar, E. Pairs trading strategy optimization using the reinforcement learning method: A cointegration approach. *Soft Comput.* **2016**, *20*, 5051–5066. [[CrossRef](#)]
41. Lintilhac, P.S.; Tourin, A. Model-based pairs trading in the bitcoin markets. *Quant. Financ.* **2016**, *17*, 703–716. [[CrossRef](#)]
42. Katoch, S.; Chauhan, S.S.; Kumar, V. A review on genetic algorithm: Past, present, and future. *Multimed. Tools Appl.* **2021**, *80*, 8091–8126. [[CrossRef](#)]
43. Whitley, D. A genetic algorithm tutorial. *Stat. Comput.* **1994**, *4*, 65–85. [[CrossRef](#)]
44. Chen, Y.J.; Lin, J.A.; Chen, Y.M.; Wu, J.H. Financial forecasting with multivariate adaptive regression splines and queen genetic algorithm-support vector regression. *IEEE Access* **2019**, *7*, 112931–112938. [[CrossRef](#)]
45. Huang, C.-F. A hybrid stock selection model using genetic algorithms and support vector regression. *Appl. Soft Comput.* **2012**, *12*, 807–818. [[CrossRef](#)]
46. Cheong, M.-S.; Wu, M.-C.; Huang, S.-H. Interpretable stock anomaly detection based on spatio-temporal relation networks with genetic algorithm. *IEEE Access* **2021**, *9*, 68302–68319. [[CrossRef](#)]
47. Sermpinis, G.; Stasinakis, C.; Zong, X. Deep Reinforcement Learning and Genetic Algorithm for a Pairs Trading Task on Commodities. 2020. Available online: <https://ssrn.com/abstract=3770061> (accessed on 16 January 2022). [[CrossRef](#)]
48. Goldkamp, J.; Dehghanimohammadabadi, M. Evolutionary multi-objective optimization for multivariate pairs trading. *Expert Syst. Appl.* **2019**, *135*, 113–128. [[CrossRef](#)]
49. Huang, C.-F.; Hsu, C.-J.; Chen, C.-C.; Chang, B.R.; Li, C.-A. An intelligent model for pairs trading using genetic algorithms. *Comput. Intell. Neurosci.* **2015**, *2015*, 939606. [[CrossRef](#)] [[PubMed](#)]
50. Windasari, I.P.; Prasetijo, A.; Pangabean, R.P. Indonesia stock exchange securities buy/sell signal detection using Bollinger Bands and Williams percent range. In Proceedings of the 2018 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), Yogyakarta, Indonesia, 21–22 November 2018; pp. 633–636.