Knowledge-Based Systems 66 (2014) 68–81

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

A novel approach to dynamic portfolio trading system using multitree genetic programming

Somayeh Mousavi, Akbar Esfahanipour *, Mohammad Hossein Fazel Zarandi

Department of Industrial Engineering and Management Systems, Amirkabir University of Technology, Tehran, Iran

ARTICLE INFO

Article history:
Received 4 November 2013
Received in revised form 9 April 2014
Accepted 10 April 2014
Available online 2 May 2014

Keywords:
Dynamic portfolio trading system
Multitree genetic programming
Technical indices
Trading rules
Tehran stock exchange
Toronto stock exchange

ABSTRACT

Dynamic portfolio trading system is used to allocate one's capital to a number of securities through time in a way to maximize the portfolio return and to minimize the portfolio risk. Genetic programming (GP) as an artificial intelligence technique has been used successfully in the financial field, especially for the forecasting tasks in the financial markets. In this paper, GP is used to develop a dynamic portfolio trading system to capture dynamics of stock market prices through time. The proposed approach takes an integrated view on multiple stocks when the GP evolves and generates a rule base for dynamic portfolio trading based on the technical indices. In the present research, a multitree GP forest has been developed to extend the GP structure to extract multiple trading rules from historical data. Furthermore, the consequent part of each trading rule includes a function rather than a constant value. Besides, the transaction cost of trading which plays an important role in the profitability of a dynamic portfolio trading system is taken into account. This model was used to develop dynamic portfolio trading systems. The profitability of the model was examined for both the emerging and the mature markets. The numerical results show that the proposed model significantly outperforms other traditional models of dynamic and static portfolio selection in terms of the portfolio return and risk adjusted return.

1. Introduction

Dynamic portfolio optimization consists of portfolio selection problem in which we find the optimum way of investing a particular amount of money in a given set of securities through time. In this problem, we have to answer three questions, which assets are proposed to buy/sell, how much and when. This definition of portfolio selection problem regards to active portfolio management approach. An alternative approach is passive portfolio management, in which the investor establishes a well diversified portfolio and maintains it for a specific period of time. The portfolio selection problem with the passive strategy is also referred as static portfolio selection.

The foundation of static portfolio selection was laid by Markowitz in 1952 [1]. He proposed a mean–variance optimization model to design an optimum portfolio based on the idea of minimizing the risk and maximizing the expected returns. After the Markowitz's mean–variance model, many other models use its fundamental assumptions [2]. In all these models, today known as the classic models, the portfolio expected return is given by the linear combination of the expected returns of the participating stocks in the portfolio. However, the literature reports many portfolio risk measures which are often based on the moments of returns of the participating stocks. These models can be viewed as an extension of Markowitz model with one, two or more objectives including some constraints such as minimum transaction lots, cardinality and boundary constraints. In order to solve the portfolio selection problem, many studies utilized artificial intelligence (AI) techniques such as the artificial neural networks [3,4] as well as the metaheuristics like simulated annealing, Tabu search and genetic algorithms [5,6].

Alternatively, the active strategy tries to find undervalued or overvalued stocks dynamically in order to achieve a significant profit. One approach is to use the technical analysis to predict stock price trends. The earlier studies have utilized the linear methods of time series analysis to forecast the stock price trends [7]. However, the linear models are established based on some assumptions such as linearity and normal distribution of stock prices. Since the stock market is a highly nonlinear dynamic system, the optimum portfolio is rather difficult to construct through the mathematical way only. Therefore, AI techniques look more promising since they have the ability to deal with the complex nonlinear problems and

* Corresponding author. Address: No 24, Hafez Ave, P.O. Box: 15875-4413, Tehran, Iran. Tel.: +98 (21) 6454 5369; fax: +98 (21) 6695 4569.

E-mail address: esfahaa@aut.ac.ir (A. Esfahanipour).

http://dx.doi.org/10.1016/j.knosys.2014.04.018
0950-7051/

they are self-adaptive for dynamically changing problems. Among these techniques, artificial neural networks, genetic algorithms and genetic programming are the most applied in technical financial forecasting [8].

As a main approach in the AI field, artificial neural network (ANN) has been widely used because of its ability to forecast financial instruments [9–13]. A vast literature about ANN applications in financial forecasting and trading system development has been presented in [14]. In this study, ANN was employed to propose a prediction-based portfolio optimization model that can capture short-term investment opportunities. Despite the wide spread use of the artificial neural networks in the financial domain, there are significant problems that must be addressed. ANNs are data driven models and the underlying rules in the data are not always apparent, which leads to so called black box models. Therefore, the investors cannot benefit from the knowledge discovery in the analytic process.

Genetic algorithm (GA) is another AI technique which has been applied successfully to financial problems. As one of the most popular heuristic optimization techniques, GA was originally developed by Holland in 1975 [15]. This search technique has been widely used because it is simple and has no restrictive assumptions about the solution space. Oh et al. used genetic algorithms to support portfolio optimization for index fund management [16]. In their later work, they proposed a new portfolio selection algorithm based on the portfolio beta using GA [17]. Lin and Liu extended Markowitz model with minimum transaction lots and used GA as their solver [18]. Moreover, GA is utilized to solve the classical portfolio optimization of Markowitz with minimum transaction lots, cardinality constraints and sector capitalization [19]. Although GA has been successfully applied to solving portfolio optimization problems as well as ANN [3,4], they showed a good performance for static portfolio selection with passive strategy. In the context of trading system development, GA was used to exploit technical trading rules in the US exchange market [20]. These trading rules led to positive excess returns in comparison with buy and hold strategy. In [21], a genetic algorithm was used to find technical trading rules for Standard and Poor's Composite Stock Index. In [22], a GA model was proposed to build an associative classifier that can discover trading rules from many numerical technical indicators. In this study, the associative classification rules were extracted to express relations between numerical data, for the first time. In this structure, the left side of the associative classification rules contains a set of trading signals, advised by the technical indicators, and the right side indicates buying or selling signals. In another research [23], a hybrid model integrating GA and support vector machines (SVM) was proposed to explore stock market trends. In this model, GA was applied to selecting the appropriate combination of input features and parameters of SVM and then least squares SVM was evolved to predict stock market movement direction in terms of historical data series. Integration of GA and ANNs was also applied successfully to predict stock markets [24,25]. In [24], the self organizing map neural network was used to cluster the data set and the GA was used to extract a fuzzy rule base from historical data. In [25], GA was used as a global search method to evolve neural networks initial weights and then the Levenberg–Marquardt back propagation algorithm was used as a local search method to tune the obtained weights.

However, genetic programming (GP) seems to be more appropriate in extracting knowledge from data, because of its structure. GP is viewed as the extension of GA, developed by Koza in 1992 [26]. The main difference between these two approaches is in the representation scheme used. GA uses string representations which prepare solutions for the problem, whereas GP represents individuals as executable programs which can solve the problem. So far GP has been applied successfully to a wide range of financial fields such as portfolio insurance [27], bankruptcy prediction [28], nonlinear time series forecasting [29] and stock trading systems [30–35]. In [27], a dynamic proportion portfolio insurance strategy was extended to generate a time adaptive risk multiplier according to the market conditions. In this strategy, GP is used to evolve an equation tree for the risk multiplier based on the risk variables. Also, GP with decision tree structure is utilized to classify bankrupt and non-bankrupt firms based on the firm's financial ratios [28]. Moreover, the ability of GP in forecasting the nonlinear time series is confirmed. In [29] a hybrid forecasting model was proposed to forecast nonlinear time series by combining autoregressive integrated moving average (ARIMA) with GP. In this approach, the ARIMA was utilized to model the linear component of time series and the GP was utilized to model the nonlinear component of time series. This study focused on the GP's ability to derive a mathematical equation from small data sets comparing with ANNs and SVM. In GP approach for trading system development, trading rules are extracted from historical data in the form of IF–THEN rules. In [30–33], the trading rules were represented as decision trees with binary or trinary outputs. The mentioned studies show that GP could generate profitable binary technical trading rules with buy and sell signals in risk-unadjusted basis [31] as well as trinary trading rules with buy, hold and sell signals when risk and transaction cost are considered [33]. Chen et al. [34,35] have used genetic network programming (GNP) to develop stock trading systems. GNP is an extension of GP in terms of individuals with graph structures. Recently, Mabu et al. [36] have extended a GNP with rule accumulation (GNP-RA) algorithm for decision making in stock trading. In GNP-RA, a large number of rules are extracted throughout the generations instead of one rule. They showed the rule-based stock trading model outperforms the conventional individual-based stock trading model.

All of the mentioned studies have developed dynamic trading systems for tracking individual securities or market indices. However, it is proved that the risk of investment can be reduced by diversification, i.e. investing in a portfolio of securities rather than one security. Hence, the most of investors tend to use portfolio selection and management tools. In the context of portfolio selection, the above-mentioned studies tried to select an optimum portfolio using passive portfolio management approach. They did not utilize technical analysis indices and technical trading rules to develop a dynamic portfolio trading system.

Ghandar and his colleagues extended a fuzzy rule based system for dynamic portfolio trading using evolutionary algorithms [37]. Their portfolio trading system uses some technical indicators of the stocks as the inputs, to rank the stocks according to their fitness function as a "buy" recommendation. Notwithstanding the good performance of their system which is accepted by both the financial industry and academia, there are three ways to improve their model from our point of view. First, Ghandar and his colleagues have provided an identical rule base for all stocks, whereas each stock may fluctuate in a different pattern. Therefore, the performance of the system can be improved by recognizing a specific pattern which can be represented by a unique set of rules for each stock. Second, the cash earned by selling the worst-ranked stocks is distributed evenly over the best-ranked stocks, whereas this distribution is not optimal in the context of portfolio optimization. In other words, the changes in the stocks weights are not determined by their system and the system uses the same weights to include the best-ranked stocks in the case of portfolio rebalancing. Third, their system uses well-known technical indicators as input variables, for example 20 days moving average (MA) with lag of 20 days. However, the optimal parameters of technical indicators are case specific and should be determined accordingly. For example, the selected lag can influence the profitability of the MA rule [38].

Chen et al. in 2010 have extended a learning model to build a stock portfolio that changes with time, based on time adapting GNP [39]. The GNP model integrates the information from technical indices and candle stick chart to make decisions about portfolio composition. Their model is applied on the Japanese stock market and according to comparison results, it outperforms other traditional models. They have extended a different model for decision making about each stock trading. Despite the attractiveness of their model, the portfolio weights are not updated dynamically, which is a vital feature of dynamic portfolio selection. At the beginning, the initial budget is assigned to different brands according to their profitability in previous periods. Then the stocks are sold or bought according to technical rules signals. Consequently, the correlations between stocks are not considered during the period. Moreover, transaction cost and risk are not considered. A dynamic portfolio management model has been proposed based on technical analysis indicators in [40] which sends buy or sell signals according to the weighted sum of technical rules scores. The importance degree of each technical rule is learned using genetic algorithm, aiming at maximization of the return on investment. Nonetheless, the study of [40] has not rectified the defects of the two related studies of [37,39]. Recently the dynamic portfolio selection problem has been studied from a signal processing perspective [41]. This study has introduced a threshold rebalanced portfolio (TRP) based algorithm for portfolio selection among independent and identically distributed discrete time markets with proportional transaction costs aiming at the maximization of the expected cumulative wealth. The algorithm is extended and evaluated over two log-normally distributed stocks as well as "Ford—MEI Corporation" stock pair. According to the experiments, the TRP based algorithm improves the achieved wealth over portfolio selection algorithms from the literature.

This paper aims to develop a dynamic portfolio trading system based on the recognized pattern in historical data of stock market without the above mentioned flaws and assumptions. The proposed system tries to forecast the future behavior of a given set of securities and to construct an optimum portfolio using these securities. This system can adapt to the changes of stock prices by updating the portfolio weights through time. Considering the ability of GP in pattern recognition and forecasting the stock market, this technique is used to develop a dynamic portfolio trading system. For our purpose, a GP with a specific structure named multitree GP forest is utilized to evolve multiple decision trees for multiple stocks. This structure helps us to generate a rule base for portfolio trading rather than a single trading rule. The profitability of the rule based trading model over the single rule model has been confirmed recently [36]. Moreover, the rule base should propose the preferred portfolio weights. Hence, the rules outputs should be a real value rather than a binary value. Additionally, many trades take place in dynamic portfolio trading, thereby transaction cost would affect the profitability of the trading system and it is considered in the proposed model. Risk measure is another controversial subject in portfolio selection. Through the past decade, there are many researches who try to develop good risk measures [42–44]. However in this study, the risk adjusted return is used in order to turn a multi-objective problem to a single objective one.

The novelty of this study can be stated from three aspects. The first one is related to the GP structure. To the best of our knowledge, this is the first study designing a multitree GP in the form of IF–THEN rules with mathematical functions as the consequent part of each rule. Previous studies developed single decision trees or multitree GP in the form of discriminant functions [45–47]. The second aspect regards to the portfolio selection using multitree GP as a data mining tool. The GP technique with multitree structure is not developed for evolving a portfolio trading rule base so far. Current emphasis is usually placed on developing single decision trees for trading the individual stocks or indices [30–33]. However, the multitree GP was used to design classifiers for multiclass problems [45,46] and clustering [47]. The third aspect is related to the dynamic portfolio trading context. According to the literature, there is not a learning model for dynamic portfolio trading, which determines the preferred portfolio weights dynamically. Moreover our proposed model considers the imposed risk and the transaction cost, which strongly influence the performance of this investment strategy, because of active trading.

The reminder parts of this paper have been organized as follows. In the next section the GP algorithm is introduced. In the third section a multitree GP model is extended for dynamic portfolio optimization. Then, the extended multitree GP model is implemented on the both emerging and mature markets and the computational and comparison results are reported. The paper closes with our conclusion.

2. Genetic programming

Genetic programming was introduced in [26] for the first time as an extension of Genetic Algorithm (GA). The main difference between GA and GP is in their representation schemes. In GA, the individual population members are fixed length character strings that encode possible solutions to the problem, whereas in GP they are programs that provide the candidate solutions to the problem.

GP as an artificial intelligence technique has recently been used successfully to extract the knowledge in the form of IF–THEN rules and has been utilized in various fields particularly in finance and technical analysis [48,49]. In this approach, the programs are usually expressed as parse trees. A parse tree is a structure that grasps the interpretation of a computer program. In the parse tree the root is the start symbol, functions are written down as nodes and their arguments as leaves [50]. The function and terminal sets are determined according to the problem domain. The set of functions and the set of terminals must satisfy two properties of closure and sufficiency properties. The closure property demands that the function set should be well defined and closed for any input arguments that it may encounter. The sufficiency property requires that the set of functions and the set of terminals should be sufficient to express a solution for the problem [26]. In technical analysis, the function set usually includes the Boolean, relational and arithmetic operators and the domain-specific functions. The terminal set usually consists of the variables and constants. For example, a typical technical trading rule "buy when the stock's price becomes higher than the single moving average of the stock's price for the last 20 days" would be represented by a tree as shown in Fig. 1. This tree can be written in infix notation as: "IF price > Average (price, 20) THEN buy".

The basic steps in a GP system are summarized as follows [51]:

(1) Randomly create an initial population of the programs from the available primitives.

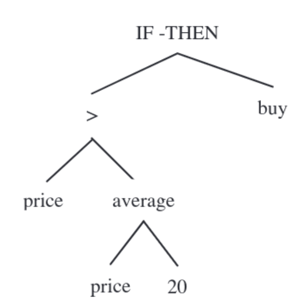

Fig. 1. A tree structure for one simple technical trading rule.

(2) Repeat
 (2.1) Execute each program and calculate its fitness.
 (2.2) Select one or two program(s) from the population with a fitness proportionate probability to participate in genetic operations.
 (2.3) Create new individual program(s) by applying genetic operators with the specified probabilities.
(3) Until an acceptable solution is found or some other stopping condition is met (e.g., a maximum number of generations is reached).
(4) Return the best-so-far individual.

3. The proposed multitree GP for portfolio trading

According to the literature, GP has been used to generate trading rules from historical data. In this approach, binary trading rules are extracted for foreign currencies [52,53], market indices [30,32] or individual stocks [31,54–56]. These trading rules develop a trading system which determines when is better to buy or sell a specific index or stock. Binary trading rules have been extended to more realistic ones which are called trinary rules using three signals of buy, sell and no trade [33].

The main aim of this study is to develop a trading system capable of proposing the best portfolio composition in every period of time. The developed trading system will be able to determine which assets should be bought or sold and how much, simultaneously. Therefore, the trading system must contain multiple rules, i.e. a rule base, at least one rule for each one of the stocks. Each rule determines the preferable amount of its underlying stock in the portfolio. Since the traditional GP structures are not able to handle this specific problem, here the consequent parts of the rules are designed as a crisp function of the weights of the stocks in the portfolio composition. Using this structure, we are able to consider transaction cost more properly.

In this section, a multitree GP is developed for dynamic portfolio selection. Compared with a single tree, the combination of multitrees is more effective for the description of a specific problem. A single decision tree can be a good descriptor for a specific part of the data set. Hence, multitree structure is required to describe the whole data set which captures relationships of different parts of the data set, simultaneously [57]. In the portfolio trading context, the multitree GP that generates a single rule for each stock and considers correlations between stocks is more logical than generating a common rule for all stocks. There are two approaches for multitree GP evolution: (1) the Pitsburgh approach in which a chromosome is encoded as a set of rules, (2) the Michigan approach in which a chromosome is encoded only for one rule. This study uses the Pitsburgh approach as the most widely used approach. In this approach, each GP individual represents a whole rule set [58]. Moreover, the output of every decision tree should be a function rather than a binary or trinary value. This property differentiates the structure of rules from traditional ones. The structure of GP chromosomes, terminal and function set, fitness measure and other preliminary setups are explained in the following sections.

3.1. Terminal and function set

Each decision tree in GP is composed of some functions and terminals i.e. internal nodes and leaves. In our extended approach of GP, the root node is IF–THEN function with two subtrees. The left subtree represents the antecedent part of the rule and the right one represents the consequent part. These two subtrees evolve with different structures and have different function and terminal sets.

For generating the antecedent part of the rules, the terminal and function sets are adapted from [31,33] as follows:

Terminal set:

Variables: the close price and transaction volume of the stocks.
Real constants: chosen in the interval of [0,100].
Boolean constants: True, False.

Function set:

Boolean operators: and, or, not.
Relational operators: <, >.
Arithmetic operators: +, −, ÷, ×.

The most used functions in technical analysis, as follows:

Norm (r_1,r_2): the absolute value of the difference between two real numbers.
Avg (s,n): average of the price or volume over the past n days.
Max (s,n): the maximum value of the price or volume over the past n days.
Min (s,n): the minimum value of the price or volume over the past n days.
Lag (s,n): the price or volume is lagged by n days.
Volatility (p,n): Standard deviation in the daily returns over the past n days.
RSI (p,n): relative strength index measures the speed and change of price movements and is defined as (1):

$$RSI(p,n) = 100 - \frac{100}{1 + RS(p,n)} \tag{1}$$

where $RS(p,n)$ is the average gains over the past n days divided by the average losses over the past n days.
ROC (p,n): rate of change measures the percent change in the price from one period to the next one and is defined as (2):

$$ROC(p,n) = \left(\frac{p}{Lag(p,n)} - 1\right) * 100 \tag{2}$$

On the other hand, in order to generate the consequent part of the rules, the set of functions is made of classical arithmetic operators and the set of terminals correspond to numerical constants and the variables of w. They are described as follows.

Terminal set:

Variables: $W(t)$, a vector of the weights of the stocks in the portfolio composition at time t.
Real constants: chosen in the interval of [0,50].

Function set:

Arithmetic operators: +, −, ÷, ×, Norm

Since the antecedent and the consequent parts of the rules have different structures, these two parts must be handled in different ways. For the antecedent part, function set variation in both Boolean and real functions violate the closure property. Consequently, the right and the left parts of trees are created separately and some restrictions are put on the structure of the right subtrees. For the right subtrees, Boolean operators can be located in the upper part; real functions and arithmetic operators can be placed in lower part of subtrees except leaves, and relational operators can be existed between the two operators. Moreover, the terminal of the subtrees corresponding to the antecedent part can be in different types, i.e. price, volume and real number. The initialization, crossover and mutation phases should take care of the type of the leaves for the last two levels of the subtrees. In this case, the arguments of relational operators are comparable and more rational rules are evolved.

3.2. Representation

In our design, each individual includes a rule base, i.e. one rule for each stock. If n stocks are going to be investigated to form a portfolio, each chromosome will have n trees. Trees will be denoted by their corresponding stock name T_i, $i = 1, 2, 3, \ldots, n$.

For each stock, we have a decision rule in which the output of the rule is in the form of a mathematical function. In our specific design, each rule is coded as one tree and the root node is always a conditional operator, i.e. IF–THEN node. The left part of the tree is the antecedent of the rule, which sends true or false according to the historical data, and the right part of the tree is the consequent of the rule, which is a function of $w_i(t)$, as the weight of the ith stock in the portfolio at time t. The consequent part of the rule is calculated according to the value of the $w_i(t)$, and sends back the preferred weight of the stock at time $t+1$, $w_i(t+1)$. In this way, the portfolio weights are updated after running a rule base and normalizing the vector of $W(t+1)$. This structure allows us to include transaction cost of trading which makes our system more realistic. Transaction cost is imposed when the weight of the stock is changed in portfolio. With this structure, the relations between multiple stocks are considered too. The ith rule corresponding to the ith stock predicts the future of the stock based on the technical indices and proposes the new $w_i(t+1)$ based on $w_i(t)$. If the rule predicts an upward (downward) trend for the stock, the proposed $w_i(t+1)$ is larger (smaller) than $w_i(t)$. After running the whole rule base, the $W(t+1)$ vector is normalized. In this way, the weights of the stocks in portfolio composition affect each other. If one stock is highly recommended to buy, its weight will increase and the other weights will slake. Furthermore, the entire rule base is coded in one GP individual and the stocks rules are evaluated altogether. The rules interact with each other during evolutionary process, because all of them are evaluated by one fitness function.

A typical chromosome representing a rule base is illustrated in Fig. 2. This chromosome includes n trees, each tree belongs to one stock. The second tree corresponding to the second stock is highlighted for example. The tree is a decision rule which determines the second stock's weight in the portfolio as follows:

- IF $RSI(p,\ 10) < 2 \times 18$ AND $Lag(V,6) > |90{,}000–880{,}000|$ THEN $w_2(t+1) = (w_2(t) + 3) \times (w_2(t)/0.5)$

This rule states that when the RSI of the second stock price over the past 10 days is less than 36 and the transaction volume of the second stock on 6 days ago is greater than 790,000, the weight of the second stock should be updated as $w_2(t+1) = (w_2(t) + 3) \times (w_2(t)/0.5)$, where $w_2(t)$ is the weight of the second stock in the portfolio at time t. Otherwise, the second stock's weight should not be updated.

3.3. Initialization

Trees are initialized randomly using function and terminal set. The function and terminal set used here are described in Section 3.1. Different approaches could be used to generate random initial population, including full, grow and ramped half-and-half methods [51]. In the full method, trees are created with the same shape and size. In this method, all branches of the trees should have the maximum predefined depth. In the grow method, the trees are allowed to have more varied shapes and sizes. However, the depth of each rule should not exceed from the maximum predefined depth.

The ramped half-and-half method is a combination of the two above-mentioned methods. In this approach, the half of the initial population is produced using the grow method and the other half of the initial population is produced using the full method. As shown in Fig. 2, we used the ramped half-and-half method to initialize the population as a widely used method [49]. Because of the particular structure of our rules, trees should be created in a recursive manner from the roots to the leaves.

3.4. Fitness evaluation

Fitness evaluation is one of the most time consuming steps in GP training, especially in the case of large training data sets. For this reason, instead of training the GP with all training samples (N) in each generation, it is done in a step-wise manner in which the number of training samples increases linearly in steps. According to the literature, the step-wise learning can reduce the computational time of the learning process significantly [59], and more surprisingly it can improve the performance of the trained individuals [60].

In the step-wise learning, the number of the steps, s_1, is determined based on the maximum number of generations. Then the training dataset is divided into s_1 sub-datasets where each subset

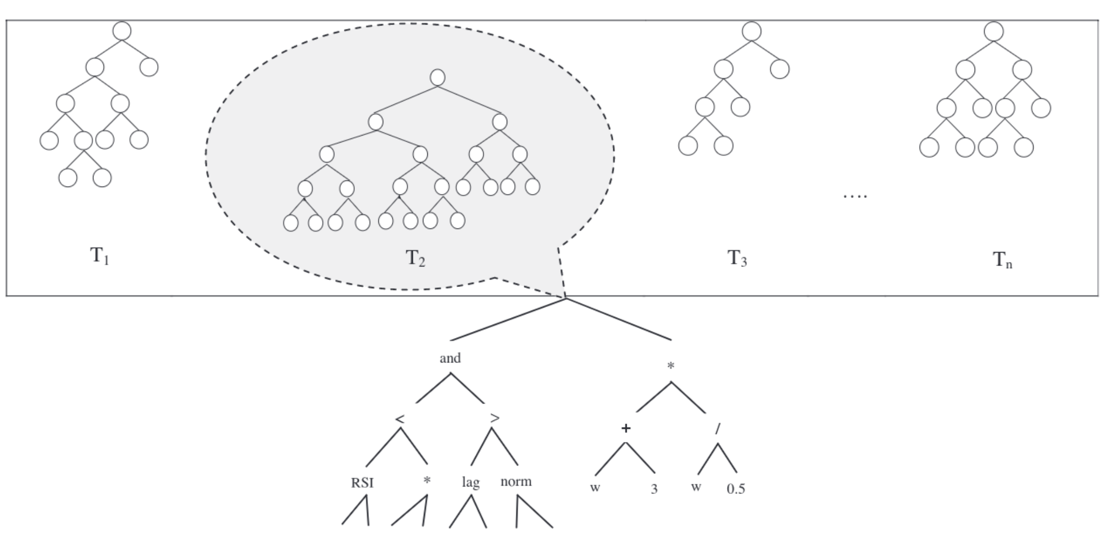

Fig. 2. A typical chromosome representing a rule base.

includes N/s_1 training samples and N is the total number of training samples. The step-wise learning is accomplished by the predetermined M_s generations of GP, where $M_s \leqslant M$ and M is the maximum number of generations. This learning method works as follows. In the first step, the learning procedure is done using the training samples of the first sub-dataset. This step is terminated after the step size of M_s/s_1 GP generations. For the second step, the second sub-dataset is fed into the available training samples. And in the ith step the GP individuals are trained with the sub-datasets of $1, 2, 3, \ldots, i$. This process is repeated for M_s GP generations. Then GP evolutionary process is continued with all N training samples till Mth generation. A more detailed explanation of this technique is presented in [45,59,60].

In this study, the risk adjusted measure is chosen to evaluate the fitness of individuals. Risk adjusted measures are generally calculated by the ratio of the excess return of the investment over its risk, where excess return is the acquired return of the investment over risk free rate. There are a various number of performance measures because of their different definitions of risk. Among a large number of risk adjusted measures, the conditional Sharpe ratio is selected to apply in the proposed GP model. Conditional Sharpe ratio was suggested by Agarwal and Naik in 2004 [61]. This measure replaces standard deviation with conditional value at risk (CVaR) in the denominator of the Sharpe ratio. Conditional Sharpe ratio is the expected excess return of the investment per units of CVaR. For portfolio selection problem, the conditional Sharpe ratio can be calculated as (3):

$$\text{Conditional Sharpe Ratio} = \frac{E(r - r_f)}{CVaR_{1-\alpha}} \tag{3}$$

where r is the return of the portfolio; r_f is the risk free rate and $CVaR_{1-\alpha}$ is the conditional value at risk of the portfolio return over the given time horizon with $100(1-\alpha)\%$ confidence level.

CVaR is a coherent risk measure which can include the subjective risk aversion of the investor and also takes into account the shape of the loss distribution in the tail of the distribution [62,63]. It is the expected loss under the condition that the loss is greater than the value at risk (VaR). VaR is interpreted as the maximum expected loss in $100(1-\alpha)\%$ of the cases, whereas CVaR can be interpreted as the average expected loss in the worst $100\alpha\%$ of the cases. There are three approaches to calculate CVaR as the model building approach, the historical simulation and the Monte Carlo simulation. In our proposed model, the historical simulation has been chosen because of the availability of the historical data. Also, this approach makes no assumptions about the probability distributions of the stock returns.

The fitness of an individual is evaluated as the cumulative risk adjusted measure of the portfolio trading system advised by the multitree GP individual. Cumulative risk adjusted measure is the sum of the risk adjusted measures of all transactions in the period and is calculated as (4).

$$\text{Fitness Function} = \sum_{i=1}^{m} \frac{R_i - r_f}{(CVaR_{1-\alpha})_i} \tag{4}$$

where m is the number of portfolio trades in the investment period advised by the trading system and $(CVaR_{1-\alpha})_i$ is the imposed risk from the ith portfolio trading and R_i is the return of the ith portfolio trading which is calculated as (5).

$$R_i = \frac{P_i - P_{i-1} - TC_i}{P_{i-1}} \tag{5}$$

where P_i is the market value of the portfolio at the end of the ith trading period and TC_i is the imposed transaction cost for the ith portfolio transaction.

3.5. Selection

Once a GP generation is completely evaluated, some individuals should be selected to create a new population. There are two common selection methods in evolutionary algorithms including roulette wheel selection and tournament selection [51]. In the present model, the roulette wheel method has been chosen as the most widely used selection method. In roulette wheel, each individual is selected with a probability that is proportionate to its fitness. The selection probability of the ith individual is calculated as (6).

$$\Pr_i = \frac{f_i}{\sum_{j=1}^{PS} f_j} \tag{6}$$

where f_i is the fitness of individual i and PS is the population size of each GP run.

3.6. Genetic operators

In each generation, new population should be created using genetic operators. Here, crossover, mutation and reproduction have been used as genetic operators.

Bloat, uncontrolled growth of program sizes during GP runs, is an inevitable event in GP evolution. Crossover has a strong preference for creating offspring with a very non-uniform size. To prevent bloating, the most common method is to impose either a size limit or a depth limit or even both for the generated programs [51]. In our model, maximum depth of offspring is considered to prevent bloating. Therefore, offspring cannot exceed a specified depth. Genetic operator should be implemented again if offspring depth exceeds the predefined depth [64].

3.6.1. Crossover

Crossover plays a critical role in GP evolutionary process. This operation is applied to selected individuals of C^1 and C^2 as parents with a predetermined crossover rate. We have modified the crossover operation for our specific application. The operation is similar to the modified crossover operation which was presented in [45]. In our approach, one tree is selected randomly from the first parent (T_s^1), and again a node is randomly selected from T_s^1 as the first crossover point. However another crossover point is selected on the second parent from a subset of allowed nodes of T_s^2; because of the specific structure of the decision trees in this application. This subset includes terminals or functions which create meaningful offsprings. Offsprings are created by swapping the subtrees rooted at the crossover points of the sth trees. In addition, the trees T_i^1, T_i^2 are swapped for all $i = s+1, s+2, \ldots, n$. In this way, subtrees are swapped between rules of the same stock. The reason is that a subtree of a good decision rule for stock i, may not be useful for another stock j, $i \neq j$. Also, swapping the subsequent trees leads to the rule bases with different combination of the evolved rules. The new combination of the evolved rules can propose better portfolio trading systems. Fig. 3 shows an example of crossover operator in the case of multitree individuals.

In this example, one node of T_2^1, shown by the dotted circle, is randomly selected as one crossover point. Then, one node of T_2^2 is randomly selected from the subset of the allowed nodes as another crossover point. The subtrees lying crossover points are swapped and also $T_3^1, T_4^1, \ldots, T_n^1$ are swapped with $T_3^2, T_4^2, \ldots, T_n^2$ respectively. This study implements this GP crossover which is similar to the one-point crossover of GA. However, two-point crossover or others can be applied too.

3.6.2. Mutation

Mutation is another genetic operator which is implemented on some individuals with a predetermined mutation rate. In our

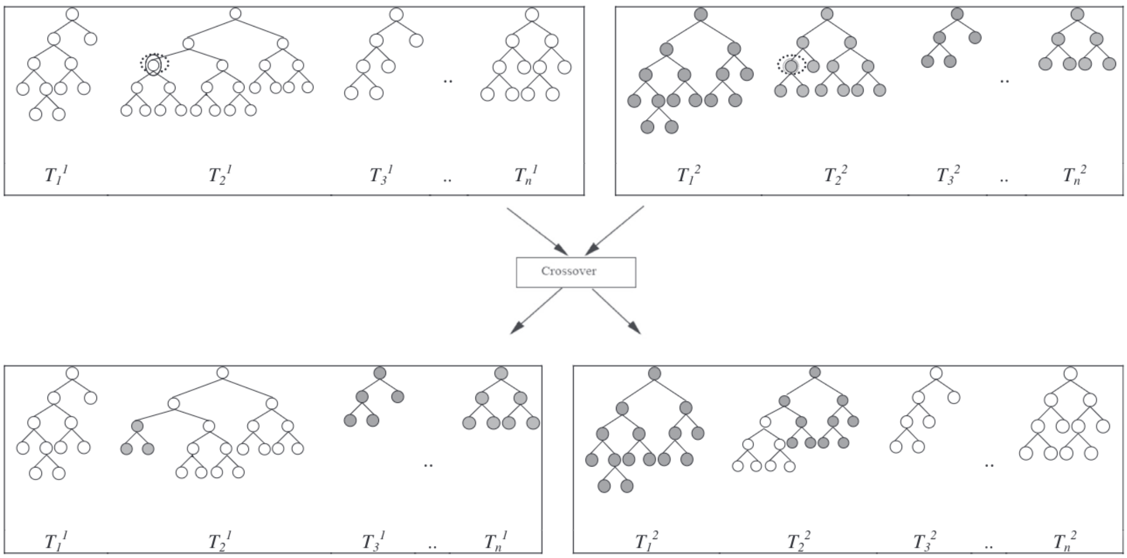

Fig. 3. A typical representation of crossover operator.

design, a tree is randomly chosen from the selected individual and the mutation is applied like traditional mutation in GP with tree structure. Several mutation strategies have been developed as follows [49]:

- Prune mutation: a non-leaf node is randomly selected and replaced by a leaf node. The leaf node should be selected from the subset of allowed terminal set, because of the specific structure of rules in this design.
- Grow mutation: a node, whether leaf or non-leaf one, is randomly selected and replaced by a randomly generated subtree. The root of subtree should be one of the allowed functions to generate a meaningful rule.
- Node mutation: the content of a selected node is mutated by another value from a subset of allowed function or terminal set, according to mutation point.

In our proposed model, all of the three mutation strategies have been implemented randomly on the selected trees.

3.6.3. Reproduction

Reproduction is implemented as another genetic operator to prevent losing the best trained individuals. This operator simply involves the selection of some individuals based on the fitness and insertion of a copy of them into the next generation without any modification. In our design, the best found individuals are also copied to the new generation with a predetermined elite size.

4. Experimental results

In this section, two sets of experiments are designed to evaluate the performance of the proposed multitree GP model for dynamic portfolio trading in different markets, emerging and mature markets. In the experiments, the proposed model is applied to fifteen selected companies listed on each market. Our proposed model can be implemented to develop a trading system for larger portfolios. However, in the present study, only 15 stocks are considered because of the time limitations. According to our experiments, the running time of the algorithm increases almost linearly with the size of the portfolio. Nonetheless, the fitness evaluation module of the algorithm is extremely time-consuming. The parallel implementation of the algorithm for fitness evaluation can be used to speed-up the process for larger portfolio trading. In this section, the main objective is to examine the profitability of our proposed portfolio trading system in different markets. In the following, first, the data are described. Then, the parameter settings for multitree GP algorithm are presented in Section 4.2. Finally, numerical results obtained from the proposed portfolio trading systems implementation as well as the comparison results are reported in Section 4.3.

4.1. Data

To test the proposed multitree GP model for dynamic portfolio trading, Iranian and Canadian stock exchange markets were chosen as the emerging and mature markets. Among 723 Iranian companies listed on the Tehran Stock Exchange, 15 companies were selected to examine the proposed model in an emerging market. These companies were selected from 30 large companies selected by TEFIX30 [65] and the top 50 shares on the Tehran Stock Exchange [66]. Also, 15 Canadian companies listed on the Toronto stock exchange were selected. All these companies were selected based on their market value and liquidity. Moreover, the companies were chosen from different activity sectors, which were active before 2007. The selected companies from Tehran and Toronto stock exchanges and their activity sectors have been presented in Tables 1 and 2 respectively. The historical data used here, are the stock price and the transaction volume for each working day between March 20, 2007 and September 22, 2011. Transaction costs for each market were calculated in a different manner according to the markets regulations. The GP based trading rules were learned on the training data in a step-wise manner. The details of this process are described in Section 3.4. The trained rules were then evaluated on the previously unseen data associated with a testing period. An initial period of two years data was put aside to start the process, because some of the primitive functions used by rules such as avg as well as CVaR may require data from the last 300 days in the worst case.

There are three approaches to select data windows for training which are initial window, extending window, and sliding window [37]. In this study, the sliding window approach was used to select the data windows. In sliding window, the training process is done based on the data of the most recent time window. The best trained rule base is then tested on the following time window. In

Table 1
Selected Iranian companies listed on Tehran Stock Exchange.

Activity sector	Company	Symbol
Motor vehicles and auto parts	Saipa Co.	SIPA1
Multidisciplinary industrial companies	GhadirInv	GDIR1
Financial services	Parsian Bank	BPAR1
Pharmaceuticals	JaberHayan P.	DJBR1
Refined petroleum products & nuclear fuel	Behran Oil	NBEH1
Metal ores mining	Chadormalu	CHML1
Basic metals	I. N. C. Ind.	MSMI1
Food and beverages other than sugar	Behshahr Ind.	SBEH1
Chemicals & by-products	Khark Petr.	PKHA1
Cement, Lime & Plaster	F. &Kh. Cement	SFKZ1
Electrical machinery and apparatus	irantransfo	TRNS1
Construction	Sadra	SDRA1
Engineering and Technical Services	Mapna	MAPN1
Mass, real estate	Shahed Investment	SAHD1
Investment	Iran National Investment	NIKI1

Table 2
Selected Canadian companies listed on Toronto Stock Exchange.

Activity Sector/Industry	Company	Symbol
Basic Materials/Gold	Barrick Gold Corporation	ABX
Utilities/Diversified Utilities	Algonquin Power & Utilities Corp.	AQN
Industrial Goods/Aerospace/Defense Products & Services	CAE Inc.	CAE
Technology/Printed Circuit Boards	Celestica Inc.	CLS
Services/Railroads	Canadian National Railway Company	CNR
Basic Materials/Oil & Gas Pipelines	Enbridge Inc.	ENB
Technology/Information Technology Services	CGI Group, Inc.	GIB-A
Financial/REIT – Diversified	H&R REIT	HR-UN
Industrial Goods/Waste Management	Newalta Corporation	NAL
Financial/Money Center Banks	Royal Bank of Canada	RY
Consumer Goods/Food – Major Diversified	SunOpta Inc.	SOY
Basic Materials/Major Integrated Oil & Gas	SunWise 2001 Fidelity Intl Port 75/75	SU
Technology/Wireless Communications	Trans Canadian Fixed Pay GIF-TIP 75/100	T
Healthcare/Drugs – Generic	Valeant Pharmaceuticals International, Inc.	VRX
Consumer Goods/Auto Parts	Westport Innovations Inc.	WPT

this study, four time windows were selected with the window size and window shifts of six months. The chosen windows as the training and testing periods are listed in Table 3.

4.2. Parameter settings

The best parameter settings for the multitree GP algorithm were determined through preliminary experiments. Although the larger population sizes led to the better results, we set population size to 100 because of the time complexity. Crossover and mutation rates were selected according to preliminary experiments. The number of generations was set to 70, as no significant improvement was observed after generation 50. Different values were also tried for other parameters. Based on these experiments, the parameter values shown in Table 4 were finally selected. For risk evaluation, the confidence level was set 95%, which is a confidence level selected by regulators [67].

Table 3
The training and testing periods of the experiments.

Training period	Testing period
March, 20, 2009–September, 21, 2009	September, 22, 2009–March, 19, 2010
September, 22, 2009–March, 19, 2010	March, 20, 2010–September, 21, 2010
March, 20, 2010–September, 21, 2010	September, 22, 2010–March, 19, 2011
September, 22, 2010–March, 19, 2011	March, 20, 2011–September, 21, 2011

4.3. Results

Multitree GP model was used to develop a portfolio trading system composed of the fifteen above-mentioned Iranian companies as well as Canadian companies along the four mentioned time windows. The rule base was generated over the training periods and the best trained rule base was then evaluated over the following testing period. In this design, the best rule base with greatest cumulative conditional Sharpe ratio was used for portfolio trading immediately after the training period. For the first testing period, the initial portfolio composition was adopted from the portfolio composition of the best trained rule base at the end of the first training period. For the next three testing periods, the initial portfolio composition was set same as the final portfolio composition of the previous testing period. During the testing period, the portfolio was updated every day based on the output function of each stock's decision rule. The portfolio weights were changed, even if one rule fires. Short sale was not permitted in the developed systems and the weights can only get positive values, i.e. $w_i \geqslant 0$. The weights were normalized after updating to satisfy a primary condition in portfolio selection problem, i.e. $\sum_{i=1}^{n} w_i = 1$. The profitability of our proposed model was investigated in terms of the cumulative conditional Sharpe ratio (CSharpe) and rate of return (ROR) of the portfolio in the testing periods. For comparison purposes, in addition to the proposed multitree GP model for portfolio trading (it is labeled here as MGP), three other approaches were applied on the selected companies including:

- A static portfolio selection model (which is labeled here as GA) developed with the same objective function and constraints of our proposed model. Genetic algorithm was utilized to solve the static portfolio selection problem. The GA model used only one factor of historical stock prices. The transaction volume and technical indices were not included in the model. This approach is selected to compare our proposed model with a profitable well-known portfolio selection model. According to the preliminary experiments, the population size, number of generations, crossover rate and mutation rate were set at 500, 100, 0.7, and 0.3, respectively.

Table 4
Parameter settings.

Population size	100
Number of generations	70
Initialization method	Ramped half and half
Max initial tree depth for IF-part	5
Max initial tree depth for THEN-part	3
Selection method	Roulette wheel
Crossover rate	0.7
Mutation rate	0.3
Elite Size	10
Max following tree depth for IF-part	7
Max following tree depth for THEN-part	3
CVaR confidence level	95%
Number of steps for step-wise learning	10
Total number of generations for step-wise learning	20

- An individual stock trading system with GP (which is labeled here as GP) adapted from our previous work [33]. In our previous work, a GP model was proposed to extract three signals trading rules from historical data for individual stock trading. The GP model uses historical data of stock price, transaction volume and technical indices in the learning process. For a detailed introduction to the GP model, the evolutionary process and the chromosome representation scheme used, the interested reader may refer to [33]. This approach is selected to compare the proposed portfolio trading system with the most similar individual stock trading system and to illustrate the diversification effect.
- The simple buy and hold strategy (which is labeled here as BH), which is usually considered as a benchmark strategy in stock market trading.

Consequently, the proposed model as the portfolio selection model with active strategy was compared with a portfolio selection model with passive strategy, an individual stock trading model with active strategy and an individual stock trading with passive strategy. Unlike the GA and BH approaches, the GP models are able to extract easily interpretable knowledge from historical data in form of IF–THEN rules. Therefore, GP models present additional knowledge about the problem in comparison with the other two approaches. Although the GP models are more beneficial regarding knowledge extraction, the performance of these approaches was compared in terms of the cumulative conditional Sharpe ratio using Eq. (4) and rate of return using Eq. (5). The performance of the portfolio selection models were compared with an equally weighted portfolio of individual stock trading models. The results of the above mentioned models implemented on Tehran and Toronto stock exchanges are investigated in the following subsections.

4.3. Neumerical experiments

4.3.1. Experiment 1: Tehran Stock Exchange

In this experiment, the performance of our proposed model was investigated in Tehran Stock Exchange as an emerging market. The experimental results in terms of conditional Sharpe ratio and rate of return are shown in Tables 5 and 6, respectively. The reported values are the average performance measures of 30 independent runs. Several findings are observed in Tables 5 and 6.

First, according to the results of the experiments, the proposed model outperforms the other three approaches. The MGP model reached the CSharpe ratios of 5.44, 28.02, 3.45, 2.83 during the four test periods of six months, whereas the other approaches achieved much less CSharpe ratios. The difference between CSharpe ratios is more apparent in the second period, when the market is in up trend. The CSharpe for BH is 0.250, for the GP and GA is 1.114 and 0.857 respectively, while for the MGP the CSharpe ratio reached 28.024, which is considerably higher than the other approaches. Also, according to Table 6, the MGP model could earn remarkable profits with rates of 17.6%, 78.5%, 21.2% and 11.4%. Furthermore, the MGP model yielded the highest rate of returns in all the investigated periods. Again, the difference between the RORs of the approaches is more prominent in the second period, when the market is in up trend.

Second, the GA model is defeated by MGP on both the risk unadjusted and risk adjusted basis. This indicates that the return on investment grows by dynamic trading and simultaneously the risk is reduced. On the other hand, the GA outperforms the BH based on the both evaluated measures. This shows that the risk is diminished by diversification whereas the return is boosted.

Third, the superiority of GP over GA is questionable. Although GA was outperformed by GP on risk adjusted basis in all the testing periods, but it could earn much more ROR in the second and the third periods. Comparing these results, we can say the dynamic trading effect in risk reduction is more than the diversification effect. However, the effect of transaction volume and technical indices should not be disregarded. The superiority of the GP in the most cases can be because of the inclusion of these factors.

Fourth, comparing the results of the GP and BH strategies, we find the CSharpe ratio and ROR can be improved by the GP dynamic trading model, especially when the market is in down trend. Fig. 4 shows the RORs of the GP model for individual stock trading versus the buy and hold strategy. According to this figure, the GP model is capable in generating profitable three signals trading rules. The GP shows very good performance where the buy and hold strategy generates small positive returns or negative returns (i.e., when the market falls or is relatively stable). However, the GP could not outperform the BH for some of the profitable stocks of TSE. These observations confirm the results of our previous study [33].

According to Tables 5 and 6, the proposed MGP model is capable of extracting the profitable portfolio trading rule bases from historical data. However, the statistical tests were carried out to examine whether the proposed MGP model significantly outperforms the other three models. Table 7 shows the results of student's t-tests for the investigated Iranian stocks among the four models. The statistical tests were conducted with respect to the average results of the four testing periods in terms of CSharpe ratio and ROR. According to the t-tests, the differences between the CSharpe ratios of the

Table 5
Performance of the four models on Tehran Stock Exchange in terms of conditional Sharpe ratio.

Test period	September 2009–March 2010				March 2010–September 2010				September 2010–March 2011				March 2011–September 2011			
Symbol	MGP[a]	GA	GP	BH	MGP[a]	GA	GP	BH	MGP[a]	GA	GP	BH	MGP[a]	GA	GP	BH
SIPA1			0.947	0.430			1.878	0.768			0.827	0.066			0.148	−0.544
GDIR1			2.291	0.277			0.544	0.103			1.069	0.753			−0.130	−0.573
BPAR1			0.706	0.191			0.592	0.058			0.474	0.092			0.203	0.015
DJBR1			0.000	−0.427			0.969	0.225			0.448	−0.426			0.583	−0.236
NBEH1			0.768	0.250			2.940	1.199			0.792	−0.669			0.298	−0.388
CHML1			0.214	−0.137			0.037	−0.966			0.439	0.390			0.059	−0.224
MSMI1			0.270	−0.068			0.319	−0.086			1.217	0.564			0.009	−1.262
SBEH1			1.324	1.263			1.117	0.269			0.996	−0.244			1.783	0.460
PKHA1			0.575	0.220			0.337	−0.449			1.067	0.651			0.707	−0.027
SFKZ1			0.000	−0.711			0.138	−0.162			0.084	−0.191			−0.085	−0.199
TRNS1			1.099	0.388			0.192	0.155			0.135	0.448			0.036	−1.396
SDRA1			−0.167	−0.530			1.620	0.739			0.217	−0.451			1.090	0.132
MAPN1			0.269	0.079			3.697	0.856			0.000	−0.197			0.798	−0.674
SAHD1			0.000	−0.197			0.379	0.231			0.000	−0.259			−0.570	−0.715
NIKI1			0.333	−0.082			1.956	0.805			0.232	0.220			0.298	0.084
Portfolio	6.042	0.118	0.575	0.063	28.024	0.857	1.114	0.250	3.457	0.509	0.533	0.050	2.831	−0.364	0.348	−0.370

[a] MGP: Our proposed Multitree GP model.

Table 6
Performance of the four models on Tehran Stock Exchange in terms of rate of return.

Test Period	September 2009–March 2010				March 2010–September 2010				September 2010–March 2011				March 2011–September 2011			
Symbol	MGP[a]	GA	GP	BH	MGP[a]	GA	GP	BH	MGP[a]	GA	GP	BH	MGP[a]	GA	GP	BH
SIPA1			0.374	0.330			0.406	0.524			0.294	0.097			0.040	−0.237
GDIR1			0.249	0.243			0.200	0.144			0.128	0.590			0.012	−0.253
BPAR1			0.151	0.194			0.128	0.118			0.174	0.116			0.061	0.058
DJBR1			0.000	−0.159			0.352	0.214			0.083	−0.255			0.089	−0.075
NBEH1			0.249	0.228			0.305	0.770			0.124	−0.429			0.008	−0.155
CHML1			0.040	0.007			0.004	−0.467			0.254	0.329			0.016	−0.068
MSMI1			0.103	0.046			0.099	0.036			0.240	0.455			0.001	−0.617
SBEH1			0.445	0.806			0.259	0.239			0.158	−0.125			0.294	0.293
PKHA1			0.241	0.211			0.082	−0.171			0.213	0.517			0.065	0.036
SFKZ1			0.000	−0.320			0.076	−0.007			0.050	−0.087			−0.045	−0.055
TRNS1			0.152	0.307			0.140	0.174			0.149	0.371			0.008	−0.688
SDRA1			0.072	−0.217			0.610	0.507			0.047	−0.273			0.330	0.120
MAPN1			0.143	0.130			0.672	0.574			0.000	−0.091			0.340	−0.306
SAHD1			0.000	−0.027			0.105	0.217			0.000	−0.135			−0.019	−0.328
NIKI1			0.085	0.038			0.363	0.545			0.132	0.208			0.124	0.095
Portfolio	0.226	0.129	0.154	0.121	0.785	0.453	0.253	0.228	0.282	0.261	0.136	0.086	0.114	−0.067	0.088	−0.145

[a] MGP: Our proposed Multitree GP model.

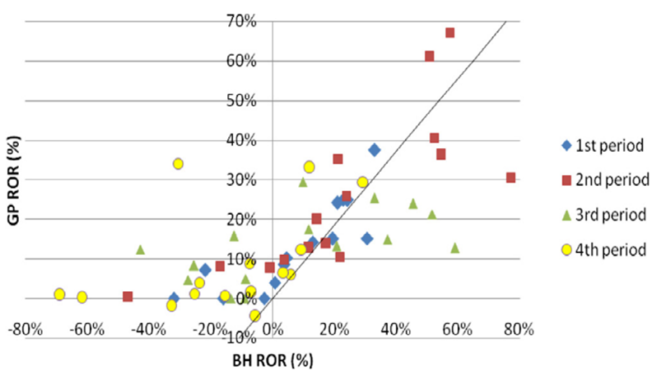

Fig. 4. The GP ROR versus BH ROR for the investigated Iranian companies.

Table 7
Student t-test for pair wise comparison of performance on Tehran Stock Exchange.

Model	GA	GP	BH
MGP	0.0000 [0.1578]	0.0000 [0.1940]	0.0000 [0.2793]
GA		0.1629 [0.0363]	0.0000 [0.1215]
GP			0.01029 [0.0853]

Notes: The table reports the p-values of tests for the pair wise dominance of models' RORs. The difference between the six-months ROR averages of the respective models are reported in brackets.

four models are significant at 1% statistical significance level. Table 7 reports the p-values for the differences between RORs of the four models. The value in bracket is the difference between the averages of the six-month RORs of the respective models through the overall testing period of two years. According to Table 7, the proposed MGP model outperformed the other three models at 1% significant level. The GA and GP models performed better than the BH at 1% and 5% significant level, respectively. Although the GP model significantly outperformed the GA in terms of risk adjusted measure, it gained a lower rate of return which is not significant.

Fig. 5 presents a schematic description of the proposed MGP model's performance during the overall testing period of two years. In this figure, the ROR curve of one of the best trained portfolio trading systems by MGP is compared with the Tehran Stock Exchange main index (TEPIX) and the buy and hold strategy. The proposed model could beat the market index remarkably. The MGP reached ROR of 203.5% at the end of the two years, while the TEPIX was grown by 132.8%. Even so, the ROR of the proposed portfolio by the MGP model was always much more than an equally weighted portfolio with BH strategy (the ROR of 203.5% versus 14.33%). On the other hand, the proposed MGP model outperformed the BH strategy in terms of the maximum drawdown. Maximum drawdown is one of the widely used risk measures by investors. It measures the largest single drop from peak to bottom in the value of a portfolio before a new peak is achieved. The maximum drawdown of the selected portfolio trading system was 21.10% during the two years testing period, which is considerably less than the maximum drawdown of the BH approach of 36.01%. However, the maximum drawdown of the TEPIX was 9.47%. The maximum drawdown can be considered as an objective function of the proposed model to be reduced. In this way, the conditional drawdown at risk (CDaR) would help. CDaR is proposed by Chekhlov and colleagues as a risk measure which combines the drawdown concept with CVaR approach [68]. It is the expected drawdown that exceed a certain threshold defined at $100(1-\alpha)\%$ confidence level.

4.3.2. Experiment 2: Toronto Stock Exchange

This experiment evaluates the efficiency of the proposed MGP model on the Toronto Stock Exchange as a mature market. The experimental results in terms of conditional Sharpe ratio and rate of return are shown in Tables 8 and 9, respectively. Comparing the results of the experiments, the following findings are obtained in the case of Toronto Stock Exchange market.

First, our proposed MGP model shows a good performance for dynamic portfolio trading in Toronto Stock Exchange. The MGP generated rule bases for portfolio trading reached average CSharpe ratios of 3.074, 2.787, 7.135 and 3.053 through the four testing periods. These ratios are much higher than the achieved ratios by the other models. The MGP model outperformed the other models in terms of ROR, too. The proposed model for dynamic portfolio trading yielded RORs of 30.9%, 14.8%, 29.3% and 7.4% on average, which are promising RORs for six months investment in Toronto Stock Exchange. The minimum ROR of 7.4% belongs to the fourth period in which the market is in down trend.

Second, while the GA model for static portfolio selection was outperformed by the proposed dynamic portfolio trading system,

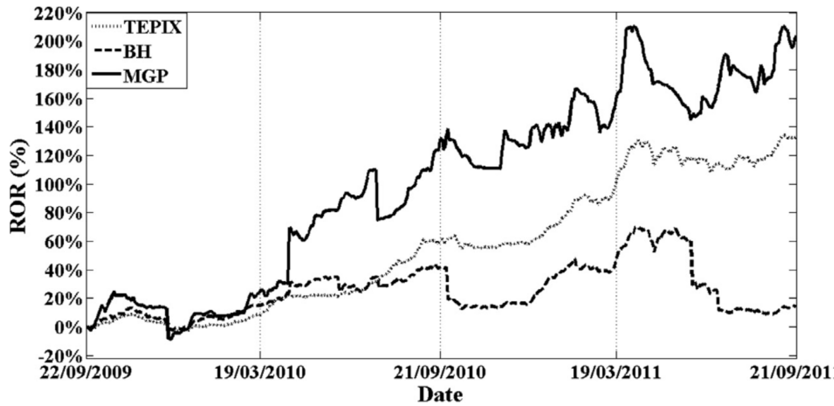

Fig. 5. ROR of the proposed MGP model on Tehran stock exchange versus TEPIX and BH strategy during the overall testing period.

Table 8
Performance of the four models on Toronto Stock Exchange in terms of conditional Sharpe ratio.

Test period	September 2009–March 2010				March 2010–September 2010				September 2010–March 2011				March 2011–September 2011			
Symbol	MGP[a]	GA	GP	BH	MGP[a]	GA	GP	BH	MGP[a]	GA	GP	BH	MGP[a]	GA	GP	BH
ABX			0.163	−0.158			0.919	0.004			1.109	−0.393			1.942	−0.249
AQN			1.169	0.255			0.147	−0.090			0.932	0.307			2.354	0.315
CAE			0.126	0.079			0.400	0.310			2.085	0.470			−2.855	−0.710
CLS			0.350	0.070			−0.284	−0.324			0.976	0.413			2.093	−0.535
CNR			0.369	0.166			0.733	0.179			0.577	0.274			1.449	−0.369
ENB			0.631	0.365			2.252	0.169			2.200	0.528			1.696	−2.374
GIB-A			0.317	0.308			1.150	−0.075			2.180	0.946			1.328	−0.333
HR-UN			0.463	0.136			1.951	0.331			−0.216	0.108			1.093	0.065
NAL			0.757	0.197			1.357	0.108			0.901	−0.070			3.799	0.871
RY			0.306	0.031			−1.263	−0.247			0.361	0.342			−1.565	−0.709
SOY			1.202	−0.042			0.898	0.465			1.303	0.089			−0.826	−0.295
SU			−0.008	−0.156			0.758	0.058			1.729	0.776			−0.671	−0.897
T			0.754	0.122			1.069	0.427			0.208	0.315			1.176	0.206
VRX			1.174	0.082			8.308	1.260			5.516	1.615			−0.135	−0.223
WPT			1.273	0.197			0.400	0.108			0.803	−0.070			0.913	0.871
Portfolio	3.074	0.151	0.603	0.110	2.787	0.193	1.253	0.179	7.135	1.064	1.378	0.377	3.053	−0.121	0.786	−0.291

[a] MGP: Our proposed Multitree GP model.

Table 9
Performance of the four models on Toronto Stock Exchange in terms of rate of return.

Test period	September 2009–March 2010				March 2010–September 2010				September 2010–March 2011				March 2011–September 2011			
Symbol	MGP[a]	GA	GP	BH	MGP[a]	GA	GP	BH	MGP[a]	GA	GP	BH	MGP[a]	GA	GP	BH
ABX			0.042	−0.019			0.092	−0.184			0.054	−0.026			0.072	−0.100
AQN			0.288	0.310			0.050	−0.043			0.128	0.120			0.160	0.100
CAE			0.092	0.073			0.164	0.159			0.177	0.165			−0.084	−0.200
CLS			0.157	0.082			−0.085	−0.234			0.146	0.220			0.111	−0.230
CNR			0.140	0.122			0.096	0.088			0.077	0.089			0.021	−0.096
ENB			0.132	0.190			0.135	0.060			0.128	0.114			0.100	−0.466
GIB-A			0.130	0.214			0.023	−0.024			0.323	0.319			0.054	−0.098
HR-UN			0.176	0.156			0.232	0.199			−0.005	0.044			0.075	0.026
NAL			0.313	0.294			0.223	0.116			0.096	−0.051			0.412	0.694
RY			0.072	0.031			−0.099	−0.105			0.134	0.124			−0.058	−0.224
SOY			0.129	−0.058			0.299	0.493			0.158	0.068			−0.110	−0.192
SU			0.002	−0.206			0.152	0.052			0.209	0.332			−0.142	−0.370
T			0.104	0.078			0.113	0.177			0.087	0.095			0.090	0.057
VRX			0.131	0.074			0.410	0.662			0.386	0.484			−0.069	−0.073
WPT			0.449	0.294			0.148	0.116			0.115	−0.051			0.168	0.694
Portfolio	0.309	0.110	0.157	0.109	0.148	0.113	0.130	0.102	0.293	0.279	0.148	0.136	0.074	−0.027	0.053	−0.032

[a] MGP: Our proposed Multitree GP model.

it outperformed the simple BH strategy for all the investigated periods. The portfolio composition advised by GA led to more profit and less risk than an equally weighted portfolio of stocks.

Third, the GP model for individual stock trading reached higher performance measures in comparison with the GA portfolio selection model. This finding is observed on both risk adjusted

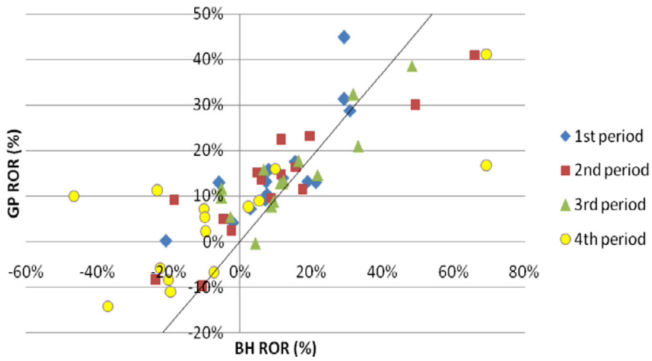

Fig. 6. The GP ROR versus BH ROR for the investigated Canadian companies.

Table 10
Student t-test for pair wise comparison of performance on Toronto Stock Exchange.

Model	GA	GP	BH
MGP	0.0000 [0.0873]	0.0000 [0.0840]	0.0000 [0.1273]
GA		0.5692 [−0.0033]	0.0000 [0.0400]
GP			0.0003 [0.0433]

Notes: The table reports the p-values of tests for the pair wise dominance of models' RORs. The difference between the six-months ROR averages of the respective models are reported in brackets.

and unadjusted basis, except for the return in the third period. Its reason can be the good performance of the investigated stocks with a little fluctuation in the third period. The GP trading rules could not detect good opportunities to send profitable buy and sell signals. On the other hand, the GA could advise a portfolio composition with a noticeable ROR of 27.9%.

Fourth, the GPs capability in extracting profitable trading rules is confirmed once more. According to Tables 8 and 9, an equally weighted portfolio of GP trading rules got into higher RORs and lower risks than an equally weighted portfolio with BH strategy. Fig. 6 compares the profit gained by three signals GP trading rules versus the buy and hold approach for individual stocks. According to this figure, the GP model is able to generate profitable trading rules for the falling and stable stocks and it is not beneficial for the extremely rising stocks. This finding is consistent with the results of Potvin et al. [31].

According to the statistical t-tests, all the differences between the risk-adjusted measures of the four models are significant. The results of the t-tests comparing the RORs are not the same. The p-values for the differences between RORs of the four models are reported in Table 10. The value in bracket is the difference between the averages of the six-month RORs of the respective models. According to the t-test results, the proposed MGP model significantly outperforms the other three models. Also, the GA and GP models performed better than BH at the 1% level of significance. However, the superiority of GP over GA is statistically meaningless.

Fig. 7 shows the profit gained by the one of the best trained portfolio trading systems, the S&P 500 index and the buy and hold approach through the overall testing period of two years. It can be seen that the ROR curve of the MGP model is always above the BH and S&P 500. This means the proposed model can consistently bring more profit. At the end of the overall testing period, the portfolio trading system advised by the MGP led to the ROR of 110.1%, while the equally weighted portfolio of the fifteen stocks and the S&P 500 was grown by 24.41% and 8.87%, respectively. Therefore, the proposed model is able to extract extremely beneficial portfolio trading rule bases for trading in the Toronto stock exchange. Conversely, the maximum drawdown of the selected portfolio trading system was 19.81%, while the maximum drawdown of the S&P 500 and BH was 17.91% and 19.92%, respectively. Again, it seems the proposed model can outperform the main index in terms of the maximum drawdown by revising the fitness function.

5. Conclusion

This paper presents a novel approach based on the multitree GP forest for dynamic portfolio selection. Compared to traditional static portfolio selection models, the proposed model can easily adapt to dynamically changing stock markets, with real time adapting portfolio construction. A multitree GP model was developed to automatically extract a rule base from historical data to develop a dynamic portfolio trading system. The rule base proposes the best combination of portfolio based on the stock's history and technical indices. The proposed multitree GP model has some advantages. First, although the model generates a distinct decision rule for every stock, nevertheless it considers correlations between the stocks. Second, using the proposed trading system the investor can profit from the small fluctuations of the stock prices. This advantage is related to the dynamic portfolio trading context. However, transaction cost affects the performance of dynamic trading, because of frequent trading. Therefore, transaction cost is considered in the model for performance evaluation. Moreover, conditional Sharpe ratio is selected for fitness evaluation of trading system. Conditional Sharpe ratio is a risk-adjusted performance measure and uses CVaR for risk evaluation which is a coherent risk measure. Above all, the model presents a rule base which is easily understandable and interpretable for all investors. Hence, they can

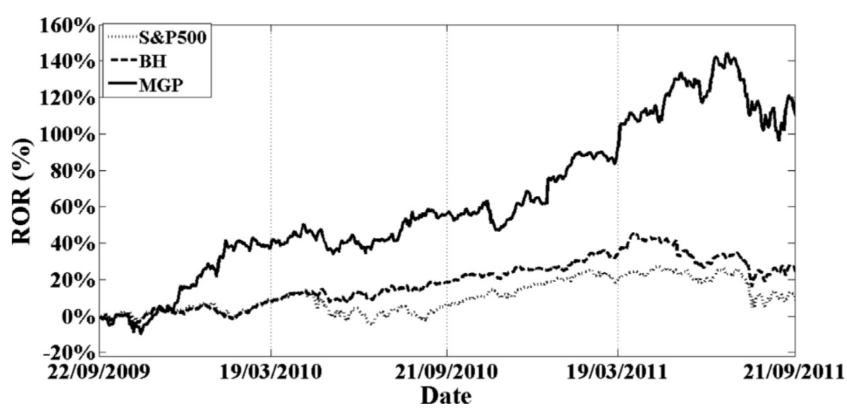

Fig. 7. ROR of the proposed MGP model on Toronto Stock Exchange versus S&P 500 and BH strategy during the overall testing period.

modify the rule base according to their financial knowledge and expectations.

Compared with the previous GP models for stock trading, the proposed multitree GP model overcomes two problems in the case of portfolio trading. First, in the proposed multitree GP model the rules for multiple stocks are evolved simultaneously and the correlations between multiple stocks are considered whereas the previous GP models generated one trading rule for each stock regardless of interdependencies of multiple stocks. Second, the multitree GP model evolves rule bases which determine the preferred weight of each stock in the portfolio composition, while the previous GP models extracted trading rules with binary/trinary signals and the whole stock is bought or sold when the respective signal is appeared.

The proposed model was implemented to examine the performance of the portfolio trading system on both the emerging and the mature markets. The Tehran Stock Exchange and the Toronto Stock Exchange were selected as two samples of emerging and mature markets, respectively. The performance of the model was examined through four time windows with different training and testing periods. The performance of the model was then compared with a static portfolio selection model with GA, an individual stock trading system developed by GP and the buy and hold strategy. Although the GP models are more beneficial regarding knowledge extraction, the performances of these approaches were also compared in terms of return and risk adjusted return. The results of both experiments show that the proposed model can benefit from the stock market fluctuations and therefore, proposes a decision making system for investors to yield outstanding rate of returns as well as high risk adjusted returns. According to the statistical tests, the proposed model significantly outperforms the traditional models of static and dynamic portfolio trading in terms of portfolio return and risk adjusted return, whether in an emerging or in a mature market. In addition, the proposed model can outstandingly beat the TEPIX and S&P 500 as the Tehran and Toronto Stock Exchange main indices. Furthermore, the results confirm once more the profitability of GP trinary trading rules compared with the buy and hold strategy for individual stock trading on Tehran and Toronto Stock Exchange. Consequently, GP is a powerful tool for the knowledge extraction in stock markets and multitree GP is capable of dealing with multiple stocks simultaneously to develop a dynamic portfolio trading system whether on emerging or mature markets. The proposed multitree GP model can dynamically have an integrated view on multiple stocks, which is important in dynamic portfolio trading.

The proposed multitree GP model can be extended to include some constraints such as minimum transaction lots and cardinality constraints with more objective functions such as maximum drawdown. Furthermore, the proposed model could be extended to cover the derivative tools. Using the proposed model as a knowledge extraction tool, it is possible to develop an expert system. In this way, the knowledge base contains the rule base obtained by the proposed multitree GP model. Expert's knowledge and viewpoints can also be added to the extracted knowledge base. Therefore, a hybrid expert system can be developed using the proposed model which benefits from the direct and indirect approaches strengths. Furthermore, the multitree GP can be used to develop a fuzzy rule based system for dynamic portfolio trading. Further studies are required to consider these issues.

References

[1] H.M. Markowitz, Portfolio selection, J. Finance 1 (1952) 77–91.

[2] E.J. Elton, M.J. Gruber, S.J. Brown, W.N. Goetzmann, Modern Portfolio Theory and Investment Analysis, seventh ed., Wiley, New York, 2007.

[3] A. Fernandez, S. Gomez, Portfolio selection using neural networks, Comput. Oper. Res. 34 (4) (2007) 1177–1191.

[4] L. Yu, S. Wang, K.K. Lai, Neural network-based mean–variance–skewness model for portfolio selection, Comput. Oper. Res. 35 (1) (2008) 34–46.

[5] J. Branke, B. Scheckenbach, M. Stein, K. Deb, H. Schmeck, Portfolio optimization with an envelope-based multi-objective evolutionary algorithm, Eur. J. Oper. Res. 199 (3) (2009) 684–693.

[6] T.J. Chang, S.C. Yang, K.J. Chang, Portfolio optimization problems in different risk measures using genetic algorithm, Expert Syst. Appl. 36 (2009) 10529–10537.

[7] G.E.P. Box, G.M. Jenkins, G.C. Reinsel, Time Series Analysis: Forecasting and Control, third ed., Prentice-Hall, Englewood Cliffs, NJ, 1994.

[8] M.A.H. Dempster, C.M. Jones, A real-time adaptive trading system using genetic programming, Quant. Finance 1 (2001) 397–413.

[9] Y. Chen, B. Yang, A. Abraham, Flexible neural trees ensemble for stock index modeling, Neurocomputing 70 (2007) 697–703.

[10] W. Shen, X. Guo, C. Wu, D. Wu, Forecasting stock indices using radial basis function neural networks optimized by artificial fish swarm algorithm, Knowl.-Based Syst. 24 (3) (2011) 378–385.

[11] J.W. Lee, J. Park, J.O.J. Lee, E. Hong, A multiagent approach to Q-Learning for daily stock trading, IEEE Trans. Syst. Man Cybern. – Part A: Syst. Humans 37 (6) (2007) C1–853.

[12] J. Moody, M. Saffell, Learning to trade via direct reinforcement, IEEE Trans. Neural Netw. 12 (4) (2001) 875–889.

[13] N. O'Connor, M.G. Madden, A neural network approach to predicting stock exchange movements using external factors, Knowl.-Based Syst. 19 (2006) 371–378.

[14] F.D. Freitas, A.F. DeSouza, A.R. deAlmeida, Prediction-based portfolio optimization model using neural networks, Neurocomputing 72 (2009) 2155–2170.

[15] J.H. Holland, Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence, University of Michigan Press, 1975.

[16] K.J. Oh, T.Y. Kim, S. Min, Using genetic algorithm to support portfolio optimization for index fund management, Expert Syst. Appl. 28 (2005) 371–379.

[17] K.J. Oh, T.Y. Kim, S.H. Min, H.Y. Lee, Portfolio algorithm based on portfolio beta using genetic algorithm, Expert Syst. Appl. 30 (2006) 527–534.

[18] C.C. Lin, T.Y. Liu, Genetic algorithms for portfolio selection problems with minimum transaction lots, Eur. J. Oper. Res. 185 (1) (2008) 393–404.

[19] H. Soleimani, H.R. Golmakani, M.H. Salimi, Markowitz-based portfolio selection with minimum transaction lots, cardinality constraints and regarding sector capitalization using genetic algorithm, Expert Syst. Appl. 36 (3) (2009) 5058–5063.

[20] R.J. Bauer, Genetic Algorithms and Investment, Wiley, New York, 1994.

[21] F. Allen, R. Karjalainen, Using genetic algorithms to find technical trading rules, J. Financ. Econ. 51 (1999) 245–271.

[22] Y.-W. Chang Chien, Y.-L. Chen, Mining associative classification rules with stock trading data – a GA-based method, Knowl.-Based Syst. 23 (2010) 605–614.

[23] L. Yu, H. Chen, S. Wang, K.K. Lai, Evolving least squares support vector machines for stock market trend mining, IEEE Trans. Evol. Comput. 13 (1) (2009) 87–102.

[24] E. Hadavandi, H. Shavandi, A. Ghanbari, Integration of genetic fuzzy systems and artificial neural networks for stock price forecasting, Knowl.-Based Syst. 23 (2010) 800–808.

[25] S. Asadi, E. Hadavandi, F. Mehmanpazir, M.M. Nakhostin, Hybridization of evolutionary Levenberg–Marquardt neural networks and data pre-processing for stock market prediction, Knowl.-Based Syst. 35 (2012) 245–258.

[26] J.R. Koza, Genetic Programming: On the Programming of Computers by Means of Natural Selection, MIT Press, Cambridge, MA, 1992.

[27] J.S. Chen, C.L. Chang, J.L. Hou, Y.T. Lin, Dynamic proportion portfolio insurance using genetic programming with principal component analysis, Expert Syst. Appl. 35 (2008) 273–278.

[28] H. Etemadi, A.A. Anvary Rostamy, H. Farajzadeh Dehkordi, A genetic programming model for bankruptcy prediction: empirical evidence from Iran, Expert Syst. Appl. 36 (2) (2009) 3199–3207.

[29] Y.-S. Lee, L.-I. Tong, Forecasting time series using a methodology based on autoregressive integrated moving average and genetic programming, Knowl.-Based Syst. 24 (2011) 66–72.

[30] C. Neely, Risk-adjusted, ex ante, optimal, technical trading rules in equity markets, Int. Rev. Econ. Finance 12 (1) (2003) 69–87.

[31] J. Potvin, P. Soriano, M. Vall'ee, Generating trading rules on the stock markets with genetic programming, Comput. Oper. Res. 31 (2004) 1033–1047.

[32] C. Fyfe, J.P. Marney, H. Tarbert, Risk adjusted returns from technical trading: a genetic programming approach, Appl. Financ. Econ. 15 (2005) 1073–1077.

[33] A. Esfahanipour, S. Mousavi, A genetic programming model to generate risk-adjusted technical trading rules in stock markets, Expert Syst. Appl. 38 (7) (2011) 8438–8445.

[34] Y. Chen, S. Mabu, K. Shimada, K. Hirasawa, Trading rules on stock markets using genetic network programming with sarsa learning, J. Adv. Comput. Intell. Intell. Inform. 12 (4) (2008) 383–392.

[35] Y. Chen, S. Mabu, K. Shimada, K. Hirasawa, Real time updating genetic network programming for adapting to the change of stock prices, IEEJ Trans. Electron. Inform. Syst. 129 (2) (2009) 344–354.

[36] S. Mabu, K. Hirasawa, M. Obayashi, T. Kuremoto, Enhanced decision making mechanism of rule-based genetic network programming for creating stock trading signals, Expert Syst. Appl. 40 (2013) 6311–6320.

[37] A. Ghandar, Z. Michalewicz, M. Schmidt, T.D. Tô, R. Zurbrugg, Computational intelligence for evolving trading rules, IEEE Trans. Evol. Comput. 13 (1) (2009) 71–86.
[38] Y. Zhu, G. Zhou, Technical analysis: an asset allocation perspective on the use of moving averages, J. Financ. Econ. 92 (2009) 519–544.
[39] Y. Chen, S. Mabu, K. Hirasawa, A model of portfolio optimization using time adapting genetic network programming, Comput. Oper. Res. 37 (2010) 1697–1707.
[40] A. Gorgulho, R. Neves, N. Horta, Applying a GA kernel on optimizing technical analysis rules for stock picking and portfolio composition, Expert Syst. Appl. 38 (2011) 14072–14085.
[41] S. Tunc, M.A. Donmez, S.S. Kozat, Optimal investment under transaction costs: a threshold rebalanced portfolio approach, IEEE Trans. Signal Process. 61 (12) (2013) 3129–3142.
[42] X. Huang, Portfolio selection with a new definition of risk, Eur. J. Oper. Res. 186 (1) (2008) 351–357.
[43] X. Huang, Risk curve and fuzzy portfolio selection, Comput. Math. Appl. 55 (6) (2008) 1102–1112.
[44] Y. Zhao, W.T. Ziemba, Calculating risk neutral probabilities and optimal portfolio policies in a dynamic investment model with downside risk control, Eur. J. Oper. Res. 185 (3) (2008) 1525–1540.
[45] D.P. Muni, N.R. Pal, J. Das, A novel approach to design classifiers using genetic programming, IEEE Trans. Evol. Comput. 8 (2) (2004) 183–196.
[46] D.P. Muni, N.R. Pal, J. Das, Genetic programming for simultaneous feature selection and classifier design, IEEE Trans. Syst. Man Cybern. – Part B: Cyberne. 36 (1) (2006) 106–117.
[47] N. Boric, P.A. Estevez, Genetic programming-based clustering using an information theoretic fitness measure, in: IEEE Congress on Evolutionary Computation, 2007, pp. 31–38.
[48] S. Chou, H. Hsu, C. Yang, F. Lai, A stock selection DSS combining AI and technical analysis, Ann. Oper. Res. 75 (1997) 335–353.
[49] A.P. Engelbrecht, L. Schoeman, S. Rouwhorst, A building block approach to genetic programming for rule discovery, in: H. Abbass, R. Sarker, C. Newton (Eds.), Data Mining: A Heuristic Approach, Idea Group Publishing, USA, 2002.
[50] A. Abraham, N. Nedjah, L. Mourelle, Evolutionary computation: from genetic algorithms to genetic programming, in: J. Kacprzyk (Ed.), Studies in Computational Intelligence (SCI), Genetic Systems Programming, vol. 13, Springer Verlag, Berlin Heidelberg, 2006, pp. 1–20.
[51] R. Poli, W.B. Langdon, N.F. McPhee, A Field Guide to Genetic Programming, 2008. <http://www.gp-field-guide.org.uk> or <http://lulu.com>.
[52] C. Neely, P. Weller, R. Dittmar, Is technical analysis in the foreign exchange market profitable? A genetic programming approach, J. Financ. Quant. Anal. 32 (1997) 405–426.
[53] C. Neely, P. Weller, Technical trading rules in the European monetary system, J. Int. Money Finance 18 (1999) 429–458.
[54] D. Mallick, V.C. Lee, Y.S. Ong, An empirical study of genetic programming generated trading rules in computerized stock trading service system, in: IEEE International Conference on Service Systems and Service Management, 2008, pp. 1–6.
[55] A. Esfahanipour, B. Karimi, S. Mousavi, The profitability of technical trading rules in Tehran stock exchange: an application of genetic programming, in: Proc. International Symposium on INnovations in Intelligent SysTems and Applications (INISTA), Turkey, 2009.
[56] A. Esfahanipour, S. Mousavi, Genetic programming application to generate technical trading rules in stock markets, Int. J. Reason. Intell. Syst. 2 (3) (2010) 244–250.
[57] C.Y. Ma, X.Z. Wang, Inductive data mining based on genetic programming: automatic generation of decision trees from data for process historical data analysis, Comput. Chem. Eng. 33 (10) (2009) 1602–1616.
[58] F.J. Berlanga, A.J. Rivera, M.J. del Jesus, F. Herrera, GP-COACH: Genetic Programming-based learning of COmpact and ACcurate fuzzy rule-based classification systems for High-dimensional problems, Inf. Sci. 180 (2010) 1183–1200.
[59] C. Fonlupt, Solving the ocean color problem using a genetic programming approach, Appl. Soft Comput. 1 (2001) 63–72.
[60] J.K. Kishore, L.M. Patnaik, V. Mani, V.K. Agrawal, Application of genetic programming for multicategory pattern classification, IEEE Trans. Evol. Comput. 4 (2000) 242–258.
[61] V. Agarwal, N.Y. Naik, Risk and portfolio decisions involving hedge funds, Rev. Financ. Stud. 17 (1) (2004) 63–98.
[62] K. Inui, M. Kijima, On the significance of expected shortfall as a coherent risk measure, J. Bank. Finance 29 (2005) 853–864.
[63] C. Acerbi, Spectral measures of risk: a coherent representation of subjective risk aversion, J. Bank. Finance 26 (2002) 1505–1518.
[64] P.W.H. Smith, Genetic programming as a data-mining tool, in: H. Abbass, R. Sarker, C. Newton (Eds.), Data Mining: A Heuristic Approach, Idea Group Publishing, USA, 2002.
[65] Tehran Stock Exchange, Articles. <http://www.irbourse.com/userfiles/file/rrm2.pdf> (accessed 11.12.11).
[66] Top 50 Shares on Tehran Stock Exchange. <http://www.50sahm.com/> (accessed 11.12.11).
[67] C. Acerbi, D. Tasche, On the coherence of expected shortfall, J. Bank. Finance 26 (2002) 1491–1507.
[68] A. Chekhlov, S. Uryasev, M. Zabarankin, Drawdown measure in portfolio Optimization, Int. J. Theor. Appl. Finance 8 (2005) 13–58.