Actionable Stock Portfolio Mining by Using Genetic Algorithms*

Chun-Hao Chen** and Ching-Yu Hsieh
Department of Computer Science and Information Engineering,
Tamkang University, Taipei, Taiwan
chchen@mail.tku.edu.tw, 600410293@s00.tku.edu.tw

Abstract

Financial markets have many financial instruments and derivatives, including stocks, futures, and options. Investors thus have many choices when creating a portfolio. For stock portfolio selection, many approaches that focus on optimizing the weights of assets using evolutionary algorithms have been proposed. Since investors may have various requests, an approach that takes these requests into consideration is needed. Based on the domain-driven data mining concept, this paper proposes a domain-driven stock portfolio optimization approach that can satisfy an investor's requests for mining an actionable stock portfolio. A set of stocks are first encoded into a chromosome. Two real numbers that represent whether to buy a stock and the number of purchased units, respectively, are utilized to represent each stock. In the fitness evaluation, each chromosome is evaluated in terms of the investorôs objective and subjective interestingness. Objective interestingness includes return on investment and value at risk. Subjective interestingness contains a portfolio penalty and an investment capital penalty, which reflect the satisfactions of the investorôs requests. Experiments on real datasets are conducted to show the effectiveness of the proposed approach.

Keywords: data mining, domain-driven data mining, genetic algorithms, minimum transaction lots, stock portfolio optimization

---

*This is a modified and expanded version of the paper ñThe YTM-based stock portfolio mining approach by genetic algorithm,ò *The 2013 IEEE International Conference on Granular Computing*.

**Corresponding author.

1. INTRODUCTION

Financial markets have many financial instruments and derivatives, including stocks [20, 29, 33], futures [34], and options [18]. Investors thus have many choices when creating a portfolio. The goal of portfolio selection is to minimize the value at risk (VaR) and maximize the return on investment (ROI). Since many factors affect the profit of the portfolio, a sophisticated approach for deriving a portfolio that considers these factors is needed. The well-known approach for acquiring a stock portfolio is the mean-variance (M-V) model proposed by Markowitz [26]. Investors can use the M-V model to find a portfolio with the maximum ROI or minimum VaR.

Based on the M-V model, many portfolio optimization approaches have been proposed, including genetic algorithm (GA)-based approaches [8, 9, 19] and multi-objective genetic algorithm (MOGA)-based approaches [2, 23]. Chen et al. proposed a combination genetic algorithm (CGA) for solving the portfolio optimization problem [8]. Chang et al. proposed a heuristic approach for portfolio optimization that uses GA with three risk measures (semi-variance, mean absolute deviation, and variance with skewness) [9]. Hoklie et al. proposed an optimization approach for finding a portfolio by evaluating its profit and risk [19]. Bevilacqua et al. proposed a MOGA-based approach for portfolio optimization based on the M-V model, where the objective functions are ROI and VaR [2]. Lin introduced the PONSGA model, which takes different risk measures (M-V, SV, VS, MAD, and LPM) into consideration to maximize the return and minimize the risk of the portfolio [23]. In addition, stocks may have minimum transaction lots, and portfolio optimization with minimum transaction lots has been proved to be an NP-complete problem [27]. Hence, some approaches have been proposed for deriving portfolios with minimum transaction lots [1, 13, 14, 21, 22, 24, 28, 31].

However, investors may have various requests. An approach that takes these

requests into consideration is needed. For example, assume that there are two investors that have 1 million (investor A) and 0.5 million (investor B) dollars, respectively. Then, the requests of investor A may be as follows: ñI want to buy 10 to 15 stocks that can maximize profit and minimize risk. The number of purchased units of each stock should be 5 to 10. Total capital for buying these stocks should not exceed 1 million dollars.ò Based on this description, investor A has four requests: (1) maximum number of stocks is 15, (2) total capital for buying stocks is 1 million dollars, (3) maximum number of purchased units of each stock is 10, and (4) the stock portfolio maximizes the profit and minimizes the risk. The requests of investor B may be as follows: ñI want to by 5 to 10 steady stocks that can maximize the return on investment. The maximum number of purchased units of each stock should be 3 to 5. Total capital for buying these stocks should not exceed 0.5 million dollars.ò Based on this description, investor B has five requests: (1) maximum number of stocks is 10, (2) total capital for buying stocks is 0.5 million dollars, (3) the stock portfolio maximizes the profit, (4) the purchased stocks should be steady stocks, and (5) maximum number of purchased units of each stock is 5. These requests can be divided into two categories, namely objective interestingness and subjective interestingness. The former includes maximizing the profit and/or minimizing the risk of the portfolio, and the latter includes the maximum number of stocks, total capital for buying stocks, , and the maximum number of purchased units of each stock. This study develops an approach that can take an investorôs objective and subjective requests into consideration for mining a stock portfolio.

Recently, Cao et al. [4, 5, 6, 7, 10, 11] proposed the domain-driven data mining (D^3M) concept, and combined it with industry knowledge to mine actual and useful information. The goal of D^3M is to derive actionable knowledge by considering technique and business interestingness. Many approaches have been proposed based

on D^3M for solving various problems [32, 35]. In this study, based on the D^3M concept, a domain-driven-based stock portfolio optimization (DDSPO) framework is proposed. Then, based on this framework, the DDSPO approach that considers the investorôs objective and subjective interestingness for mining an actionable stock portfolio (ASP) is developed. A set of portfolios is first generated according to the cash dividend yields of stocks. Two real numbers are used to represent a stock in the portfolio, one that indicates whether to buy the stock and the other for its purchased units. Then, each chromosome is evaluated in terms of objective criteria (ROI and VaR) and subjective criteria (suitability of the portfolio). The suitability criterion of a chromosome consists of a portfolio penalty (PP) and an investment capital penalty (ICP), which is used to reflect the satisfactions of the usersôrequests (e.g., maximum investment capital and the maximum number of purchased stocks). After evolution, the derived actionable portfolio that satisfies the investorôs objective and subjective criteria is outputted. Experiments on real datasets were conducted to show the effectiveness of the proposed approach.

This paper makes three main contributions. (1) Based on D^3M, the DDSPO framework is proposed (see Section 4). (2) Based on this framework, the DDSPO algorithm is proposed for mining an ASP; this is the first work that utilizes D^3M for solving the stock portfolio optimization problem (see Section 6). (3) The DDSPO algorithm considers objective and subjective criteria simultaneously for deriving the ASP effectively (see Section 7).

The rest of this paper is organized as follows. Preliminaries are given in Section 2. Stock portfolio optimization approaches are reviewed in Section 3. The proposed stock portfolio optimization framework is described in Section 4. The components of the proposed approach are described in Section 5. The proposed DDSPO algorithm and an example are presented in Section 6. Experimental evaluations on real datasets

are shown in Section 7. Conclusions and ideas for future work are given in Section 8.

2. PRELIMINARIES

In this section, preliminaries are given. The concept of GAs is stated in Section 2.1. Modern portfolio theory is described in Section 2.2. The D^3M concept is introduced in Section 2.3.

2.1 Genetic algorithms

GAs [15, 17] have become increasingly important for solving difficult problems since they provide feasible solutions in a limited amount of time. They were first proposed by Holland in 1975 and have been successfully applied to optimization, machine learning, neural networks, and fuzzy logic controllers [16, 17]. To apply GAs to a problem, the first step is to define a representation that describes the problem states. A bit string representation is commonly used. An initial population of individuals, called chromosomes, is then defined and three genetic operations (crossover, mutation, and selection) are performed to generate the next generation. Each chromosome in the population is evaluated in terms of a fitness function to determine its goodness. This procedure is repeated until a user-specified termination criterion is satisfied.

2.2 Modern portfolio theory

Investors want to maximize return and minimize risk. In 1952, Markowitz proposed the M-V model, which uses an efficient frontier to describe the risk in a portfolio, to help investors decide asset combinations [26]. In the M-V model, the mean and variance values represent the return and risk of a portfolio, respectively. For the M-V model, the return distribution in a specific time is used to decide the portfolio.

The expected return of an asset is utilized to estimate its risk. When the risk (return) is fixed, the goal of the M-V model is to maximize (minimize) the return (risk). Thus, when the return is fixed, the M-V model becomes:

$$\text{Min } \sigma_{\mathrm{P}}^2 = \sum_{i=1}^{n} \sum_{j=1}^{n} w_i w_j \sigma_{ij}, \qquad \text{.....(1)}$$

$$s.t. \sum_{i=1}^{n} w_i = 1, \qquad \text{.....(2)}$$

$$0 \le w_i \le 1,\ i = 1, 2, ..., n, \qquad \text{.....(3)}$$

$$\sum_{i=1}^{n} w_i E(R_i) \ge R_P, \qquad \text{.....(4)}$$

where σ_{P}^2 is the risk value (variance value) of the portfolio, n is the number of assets in the portfolio, w_i and w_j are the weights of assets i and j, respectively, σ_{ij} is the covariance value for assets i and j, $E(R_i)$ is the expected return of the portfolio, and R_p is the desirable level of the expected return of the investor.

Based on the M-V model, many evolutionary algorithms have been proposed for deriving a portfolio [2, 8, 9, 19, 23], where some of them are GA-based approaches [8, 9, 19] and others are MOGA-based approaches [2, 23]. However, when investors have various requests, the M-V model cannot easily be utilized for deriving a portfolio that satisfies all requests of investors. Thus, the aim of this paper is to design an approach that can satisfy an investorôs requests for mining a stock portfolio.

2.3 Domain-driven data mining concept

In order to deal with the issues surrounding real-world data mining, Cao et al. [4, 5, 6, 7, 10, 11] proposed the D^3M concept. They combined this concept with industry knowledge to mine useful real information, named actionable knowledge discovery (AKD). The word ñactionableò means that the derived knowledge patterns not only provide important grounds for business decision-makers for taking appropriate actions, but also deliver expected outcomes to business. Under this goal, four frameworks for the logical concept of D^3M are described in [6]. They are post-analysis-based AKD (PA-AKD), unified-interestingness-based AKD (UI-AKD), combined-mining-based

AKD (CM-AKD), and multisource combined-mining-based AKD (MSCM-AKD).

Since the proposed approach is based on the UI-AKD framework, the details of this framework are stated. According to [11], the UI-AKD framework can be formalized as follows:

$$UI\text{-}AKD: DB \xrightarrow{e,\ i(),\ m,\ \Omega_d,\ \Omega_m} \tilde{P}, \tilde{R}$$

where $i()$ can be defined as follows:

$$i() \rightarrow \eta \check{t}_i() + \varpi \check{b}_i() ,$$

where weights η and ϖ reflect the interestingness balance between data analysts and domain experts in terms of the business problem, data, environment, and deliverable expectation. $i()$ consists of t_i and b_i, where t_i is technical interestingness and b_i is business interestingness. In some cases, both weights and aggregation can be fuzzy. In other cases, the aggregation may happen in a step-by-step manner. For each step, weights may be differentiated. In practice, the combination of technical interestingness with business expectations may be implemented using various methods. An ideal situation is to generate a single formula $i()$ that integrates t_i and b_i, and then to filter patterns accordingly. The environment (e), domain knowledge (Ω_d), meta knowledge (Ω_m), technical interestingness, and business interestingness are then merged using method m into a final pattern set ($\tilde{P}$). The patterns are converted into business rules ($\tilde{R}$) as the final deliverables that reflect business preferences and needs. Based on the UI-AKD framework, the DDSPO framework is proposed. Then, based on the proposed framework, the DDSPO algorithm is proposed for mining an ASP. Details are presented in Sections 4 and 5, respectively.

3. STOCK PORTFOLIO OPTIMIZATION APPROACHES

In this section, related stock portfolio optimization approaches are reviewed. Hybrid approaches used for portfolio optimization are described in Section 3.1. Since

minimum transaction lots should be considered in some financial markets, approaches for deriving portfolios with minimum transaction lots are reviewed in Section 3.2.

3.1 Hybrid stock portfolio optimization approaches

Many evolutionary algorithms for deriving a portfolio based on the M-V model have been proposed, including those that use GAs [8, 9, 19] and MOGAs [2, 23]. Many hybrid approaches that combine various mining techniques have also been proposed [3, 12, 13, 14, 25]. Hachloufi et al. proposed an approach based on classification and a GA for obtaining an optimal stock portfolio [12]. Assets are first classified into classes with the expected returns and risk. Then, the MinVaRMaxVaL algorithm is used for selecting the optimal assets portfolio. Gupta et al. presented an integrated approach for portfolio selection [13]. Assets are first classified into three pre-defined classes (liquid, high-yield, and less-risky assets) using support vector machines. Then, a real-code GA is utilized to derive portfolios from the three classes according to usersô preferences. Gupta et al. also proposed a three-stage framework that takes financial and ethical criteria into consideration for portfolio selection. Fu et al. proposed a GA-based approach for determining the optimal parameter settings of various technical indicators and weights of assets [14]. Bermúdez et al. proposed a MOGA-based approach that uses a fuzzy ranking strategy for selecting efficient portfolios [3]. Li et al. proposed a multi-objective portfolio selection model with fuzzy random returns with three criteria (risk, expected return, and liquidity). In order to deal with the difficulty of evaluating a large set of efficient solutions and find the best solution, a compromise-approach-based algorithm has also been designed [25]. From these approaches, we can know that various types of hybrid algorithm have been proposed for deriving portfolios. However, the userôs requests, e.g., maximum number of purchased stocks in the portfolio and maximum investment capital, are not

taken into consideration for mining the stock portfolio. The present study, based on D^3M, develops an approach that considers the user's requests for deriving an ASP.

3.2 Portfolio optimization with minimum transaction lots

Generally speaking, portfolio selection approaches are used for deriving weights of assets. However, assets may have minimum transaction lots. For instance, the minimum purchased shares (minimum transaction lots) of a stock on the Taiwan Stock Exchange (TSE) is 1000 shares. Some approaches for deriving portfolios have been proposed with this consideration [1, 21, 22, 24, 27, 28, 31]. Mansini et al. showed that portfolio selection with minimum transaction lots is an NP-complete problem, and proposed three heuristic approaches for solving the problem [27]. Kellerer et al. proposed an approach for selecting portfolios with fixed costs and minimum transaction lots using a mixed-integer linear programming model [21]. Konno et al. proposed a branch-and-bound algorithm for deriving a portfolio under concave transaction costs and minimum transaction unit constraints [22]. Mansini et al. proposed an exact approach for portfolio selection with transaction costs and rounds [28]. Based on the M-V model, Lin et al. presented three possible models for the portfolio selection problem with minimum transaction lots by GA [24]. Soleimani et al. took minimum transaction lots, cardinality constraints, and capitalization into consideration for portfolio selection using GA [31]. Babaei et al. proposed a simulated annealing meta heuristic approach for solving an extended M-V model [1].

4. PROPOSED DOMAIN-DRIVEN STOCK PORTFOLIO OPTIMIZATION FRAMEWORK

In this section, the proposed DDSPO framework is stated. The framework is shown in Figure 1.

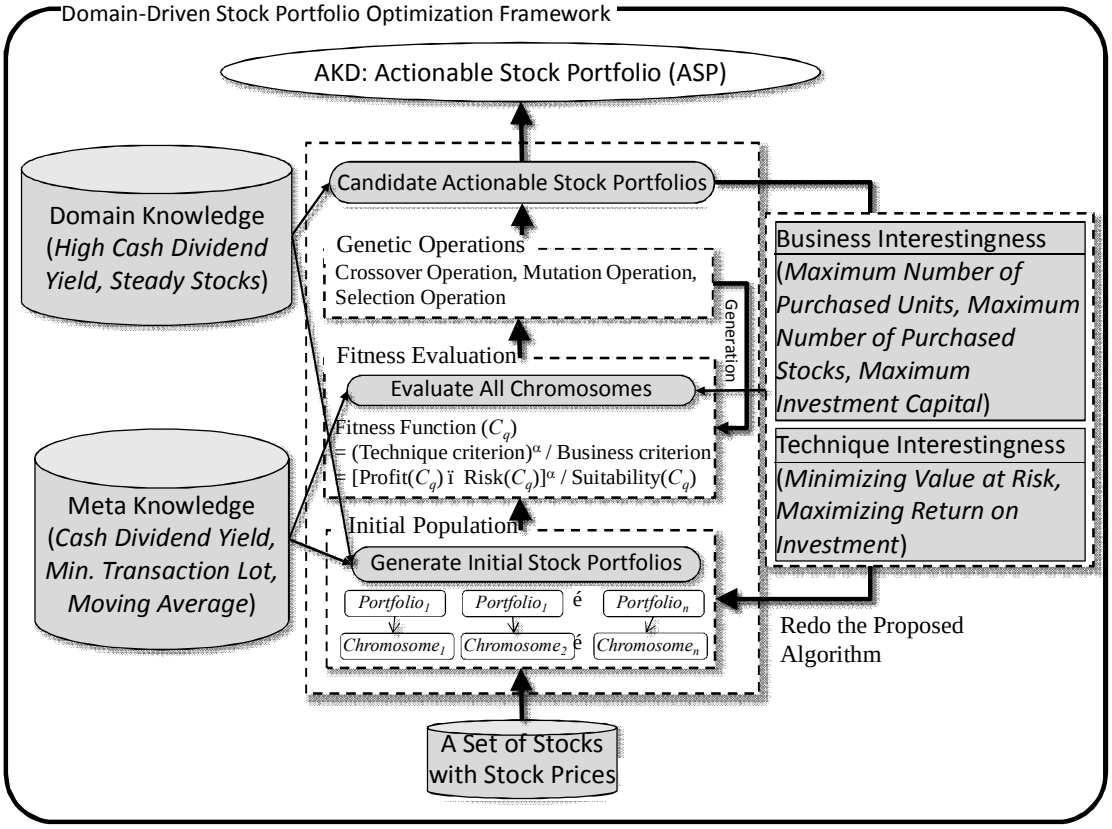

Figure 1. DDSPO framework.

The proposed framework consists of three main parts, namely the generation of the initial population, fitness evaluation, and genetic operations. To generate the initial stock portfolio, the cash dividend yield is used as the meta knowledge to generate initial chromosomes. The higher the cash dividend yield of a stock is, the larger is the probability of the stock being selected in the portfolio. The business and technique interestingness of investors are considered to design the fitness function. The fitness value of each chromosome is calculated using the profit, risk (technique interestingness), and suitability (business interestingness) factors of the portfolio, and parameter α is used to adjust the importance between technique and business interestingness. The genetic operations are then executed to generate new portfolios. The candidate ASPs are checked using domain knowledge after the optimization process. The proposed algorithm is executed until a satisfactory ASP is derived. Since the framework is based on the UI-AKD framework, the proposed DDSPO-AKD

framework for mining ASPs can be formalized as follows:

$$DDSPO\text{-}AKD: A\ Set\ of\ Stocks\ with\ Stock\ Prices \xrightarrow{e,\ i(),\ m,\ \Omega_d,\ \Omega_m} ASP$$

where e is the stock portfolio optimization environment, $i()$ consists of the business interestingness and technique interestingness, specifically the maximum number of purchased units, maximum number of purchased stocks, and maximum investment capital for business (subjective) interestingness, and maximizing return on investment and minimizing value at risk for technical (objective) interestingness. m represents the proposed DDSPO algorithm (for details, please see Section 6). Ω_m is meta knowledge, which contains the cash dividend yield, minimum transaction lot, and moving average. Ω_d is domain knowledge. The high cash dividend yield and steady stocks are utilized to evaluate the actionable ability of the derived stock portfolio. In other words, the framework combines the financial environment, business and technical interestingness, and meta and domain knowledge for deriving an ASP using the proposed algorithm m (DDSPO algorithm). The UI-AKD framework is a general D^3M framework proposed by Cao et al. for mining actionable knowledge (patterns) [5, 11]. Based on the UI-AKD framework, this study proposes the DDSPO-AKD framework for mining ASPs.

5. COMPONENTS OF PROPOSED APPROACH

This section describes the components of the proposed approach, namely chromosome representation, initial population, fitness and selection, and genetic operations.

5.1 Chromosome representation

The encoding of chromosomes for GAs is important. Several possible encoding approaches have been proposed [15, 16, 17]. When purchasing stocks, the minimum

transaction lots may exist. In Taiwan Stock Exchange, the minimum transaction lot is 1 unit (= 1000 shares), and is considered in the encoding scheme. Hence, each company (stock) com_i in the portfolio is represented by two genes, namely b_i and u_i, where b_i denotes a threshold. If the value of b_i is larger than or equal to 0.5, company com_i is selected in the portfolio. u_i is the number of purchased units of com_i, where one unit means 1000 shares. Thus, if there are n companies in a chromosome, the length of the chromosome is $2n$. A chromosome C_q that contains n companies is shown in Figure 2.

b_1	u_1	b_2	u_2	...	b_i	u_i	...	b_{n-1}	u_{n-1}	b_n	u_n

Figure 2. Representation of chromosome C_q.

Since the length of each chromosome is $2n$, one-point crossover and one-point mutation operations could be executed directly without checking how many companies are in a chromosome. Of course, only selected companies are encoded in chromosome is also workable if appropriate genetic operations and chromosome checking mechanisms are defined. An example is given below to demonstrate the process of encoding a set of chromosomes.

***Example 1*:** Assume that there are four companies, com_1, com_2, com_3, and com_4, according to Figure 2, two possible chromosomes are shown as follows:

$$b_1, u_1, \quad b_2, u_2, b_3, u_3, \quad b_4, u_4$$
$$C_1: (0.58, 8, 0.69,5, 0.13, 3, 0.18, 6),$$
$$C_2: (0.64, 2, 0.38, 9, 0.71, 5, 0.19, 8).$$

Take C_1 as an example. Since the b_1 and b_2 values of com_1 and com_2 are larger than 0.5, they are selected in the portfolio. Their numbers of purchased units are 8 and 5, which means that 8000 and 5000 shares of com_1 and com_2 are purchased, respectively. In the same way, C_2 represents 2000 and 5000 shares of com_1 and com_3 being purchased, respectively.

5.2 Initial population

A genetic algorithm requires a population of feasible solutions to be initialized and updated during the evolution process. As mentioned above, each individual in the population is a set of companies and their numbers of purchased units. How to generate population such that chromosomes could achieve good profits is an important problem. In this paper, the cash dividend yield is utilized to generate initial population. The cash dividend yield is calculated as dividend per share divided by current stock price. In the following, two companies, named Chunghwa Telecom (CHT) and Nan Ya Plastics Corporation (NPC), are used to explain the benefits of using cash dividend yield to generate initial population. By using cash dividend per share and current stock price, the cash dividend yields of two companies are shown in Table 1.

Table 1. Cash dividend yields of two companies

	2011	2012	2013	2014
Cash dividends of CHT (per share)	5.46	5.35	4.53	4.86
Stock price of CHT	100.00	94.5	93.10	94.00
Cash dividend yield of CHT	5.46%	5.66%	4.86%	5.71%
Cash dividends of NPC (per share)	2.10	0.30	1.90	2.30
Stock price of NPC	60.10	56.00	68.9	65.5
Cash dividend yield of NPC	3.49%	0.53%	2.75%	3.51%

From Table 1, since the cash dividends of CHT are NT$ 5.46, 5.35, 4.53 and 4.86, the cash dividend yields of CHT in 2011, 2012, 2013 and 2014 are calculated as 5.46%, 5.66%, 4.86% and 5.71%, respectively. In the same way, the cash dividend yields of NPC are 3.49%, 0.53%, 2.75% and 3.51%. Comparing CHT with NPC in terms of cash dividend yields, CHT is better than NPC because cash dividend yield of CHT is stable. In other words, investors who buy CHT could get stable revenues and low value at risk. Thus, assume there are n companies, the initial population is generated according to the cash dividend yield (y_i) of each company. Consider the

cash dividend yields shown in Table 2.

Table 2. Cash dividend yields (y_i) of companies

com_1	com_2	…	com_i	…	com_{n-1}	com_n
y_1	y_2	…	y_i	…	y_{n-1}	y_n

Using Table 2, the probabilities of companies to be selected into chromosomes are calculated. The results are shown in Table 3.

Table 3. Probabilities of all companies

com_1	com_2	…	com_i	…	com_{n-1}	com_n
$\frac{y_1}{\sum_{h=1}^{n} y_h}$	$\frac{y_2}{\sum_{h=1}^{n} y_h}$	…	$\frac{y_i}{\sum_{h=1}^{n} y_h}$	…	$\frac{y_{j-1}}{\sum_{h=1}^{n} y_h}$	$\frac{y_j}{\sum_{h=1}^{n} y_h}$

The cumulative probability value of each company is then calculated using Table 3. For example, the cumulative probability of com_2 is ($\frac{y_1}{\sum_{h=1}^{n} y_h} + \frac{y_2}{\sum_{h=1}^{n} y_h}$). Then, the cumulative probabilities of all companies are used to form the initial population. As a result, if a stock has a high cash dividend yield, it has a high probability of being selected to form the chromosome. By utilizing cash dividend yields of stocks to generate the initial population, the proposed approach can be improved when comparing to that generating initial population without cash dividend yields (see Section 7).

5.3 Fitness and selection

In order to develop a good set of chromosomes from an initial population, the genetic algorithm may select parent chromosomes for mating in a probabilistic way according to their fitness values. Thus, designing an effective fitness function for evaluating chromosomes is important. According to the D^3M concept, the fitness

function should consider both business and technique interestingness in order to satisfy an investorôs requests. In this paper, the profit and risk of portfolio are the technique interestingness, and the other requests are represented as business interestingness. Hence, the fitness function is defined as:

$$F(C_q) = \frac{[Technique(C_q)]^{\alpha}}{Business(C_q)} = \frac{[ROI(C_q) + Risk(C_q)]^{\alpha}}{suitability(C_q)}, \quad \text{.....(5)}$$

where parameter α is used to adjust the importance between technique and business interestingness, and $ROI(C_q)$ is the profit of portfolio in chromosome C_q, defined as:

$$ROI(C_q) = \sum_{i=1}^{j} [(s_s^{(i)} - s_b^{(i)}) * u_i + Div^{(i)} * u_i] / Cap_q, \quad \text{.....(6)}$$

where u_i is the number of purchased units of company com_i, and $s_s^{(i)}$ and $s_b^{(i)}$ are the selling price and buying price of company com_i, respectively. Cap_q and $Div^{(i)}$ are the investment capital of chromosome C_q and cash dividend of company com_i, respectively. Since cash dividend means money paid to investors (stockholders), it is a kind of profit. In general, the more the cash dividend, the better the company is. So, cash dividend is considered to evaluate the profit of a chromosome.

$Risk(C_q)$ is used to evaluate the risk of the portfolio in chromosome C_q. Here, the historical simulation (HS) method is utilized to evaluate the risk of the portfolio [30]. The HS method consists of four steps, namely collecting data, generating scenarios, simulating portfolio return, and reordering results. In the collecting data stage, the observation period t (e.g., a day, a week, a month) is defined and the stock prices are collected. Then, the portfolio return (percentage price change) of each stock is calculated using $(price_t - price_{t-1}) / price_{t-1}$. The derived returns of stocks are sorted in increasing order. According to the predefined confidence level, the n-th VaR is calculated. In order to obtain more reliable results, the average VaR of the first n values is used in this paper.

$suitability(C_q)$ consists of two parts. The first part is the investment capital

penalty, $ICP(C_q)$, defined as:

$$ICP(C_q) = \begin{cases} \dfrac{max\ Inves}{Cap_q}, if\ max\ Inves \le Cap_q; \\ \dfrac{Cap_q}{max\ Inves}, if\ Cap_q \le max\ Inves, \end{cases} \quad \text{.....(7)}$$

where Cap_q is the investment capital of chromosome C_q and *maxInves* is the predefined maximum investment. The second part is the portfolio penalty, $PP(C_q)$, which is defined as:

$$PP(C_q) = \begin{cases} \dfrac{numCom_q}{numCom}, if\ numCom \le numCom_q; \\ \dfrac{numCom}{numCom_q}, if\ numCom_q \le numCom, \end{cases} \quad \text{.....(8)}$$

where $numCom_q$ is the number of purchased stocks of the portfolio in chromosome C_q and *numCom* is the predefined maximum number of purchased stocks. Thus, $suitability(C_q)$ is defined as:

$$suitability(C_q) = ICP\ (C_q) + PP\ (C_q) \quad \text{.....(9)}$$

$suitability(C_q)$ is utilized to evaluate whether the portfolio in the chromosome meets the investorôs subjective criteria. The criteria in this paper are the predefined maximum investment capital and the predefined maximum number of purchased stocks in the portfolio. $ICP(C_q)$ is used to measure the satisfaction degree of investment capital in chromosome C_q to the predefined maximum investment capital. As a result, if investment capital of a chromosome is close to investorôs maximum investment, then that chromosome will be kept in next generation with high probability. $PP(C_q)$ is used to measure the satisfaction degree of the number of purchased stocks in the portfolio in chromosome C_q to the predefined maximum number of purchased stocks. In other words, if a chromosome has near the same number of stocks to investorôs desired number of stocks, then the chromosome will have high probability to survive in next generation. An example is given below to

illustrate the fitness evaluation of a chromosome.

***Example 2*:** Continuing *Example 1*, assume that the predefined *maxInves* is one million and that the maximum number of purchased stocks in the portfolio is set at 3. The selling and buying prices are 62 and 67.7 for com_1, and 38.05 and 28.5 for com_2, respectively. The cash dividends of companies com_1 and com_2 are 0.0302 and 0.016, respectively. The observation period used in the historical simulation is a week. Assume that the confidence level is set at 99%, and the percentages price change are -0.06841 and -0.05692 for com_1, and -0.06991 and 0.00877 for com_2. Thus, the average VaR values of com_1 and com_2 are -0.06266 and -0.03057, respectively. Since the chromosome C_1 is (0.58, 8, 0.69, 5, 0.13, 3, 0.18, 6), the investment capital is 684.1 (= 67.7*8 + 28.5*5). The fitness value of C_1 is calculated as follows:

$ROI(C_1)$=([(62-67.7)*8+0.0302*8]+[(38.05-28.5)*5+0.016*5])/684.1= 0.003613
$Risk(C_1)$ = -0.06266*8 + -0.03057*5 = -0.65413
$suitability(C_1)$ = ICP (C_1) + PP (C_1) = 684.1/1000 + 2/3 = 0.6841 + 0.67=1.3541
F(C_q) = (0.003613 + -0.65413) / 1.3541 = -0.4804

5.4 Genetic operations

Two genetic operations, one-point crossover and one-point mutation, are used in the proposed approach. The one-point crossover operator generates two new offspring from their parents with a random crossover location *d*. More sophisticated crossover operations could be utilized, e.g., the max-min-arithmetical crossover operation. In the mutation operation, a gene is randomly selected for mutation. When the odd gene in a chromosome is picked, its value is changed from [0.5, 1] to [0, 0.5] or [0, 0.5] to [0.5, 1] since the odd gene means whether to buy a certain stock. When the even gene in a chromosome is picked, a new random value in the range [1, *maxUnit*] is generated since an even gene means purchased number of units of a stock.

6. PROPOSED DDSPO ALGORITHM

In this section, the proposed DDSPO algorithm is described. An example is then

given to illustrate the proposed approach.

6.1 Proposed domain-driven stock portfolio optimization algorithm

The proposed approach first utilizes cash dividend yields of stocks to form the initial population. Each chromosome is then evaluated using the designed fitness function, which takes subjective and objective interestingness into consideration. Genetic operations are then used to generate new offspring, and the evolution process is repeated to derive an ASP. The definition of ASP is given as follows:

Definition 1 (Actionable Stock Portfolio, ASP). Given a set of stocks $S = \{s_1, s_2, ..., s_n\}$, an actionable stock portfolio is a sub set of S that could satisfy investor's objective and subjective requests and each stock in ASP has its purchased units.

INPUT: A set of companies (stocks) with stock prices $comName = \{com_i \mid 1 \le i \le n\}$, the predefined maximum number of purchased stocks in portfolio *numCom*, the predefined maximum investment capital *maxInves*, the predefined maximum number of purchased units of a stock *maxUnit,* where a unit means 1000 shares, cash dividend yields of stocks $Y = \{y_i \mid 1 \le i \le n\}$, the predefined confidence level λ, parameter α, population size P, crossover rate P_c, mutation rate P_m, and number of generations G.

OUTPUT: ASP.

STEP 1: Calculate the selection probability of each company using cash dividend yields Y:

$$\mathrm{P}(Y_i) = \frac{y_i}{\sum_{h=1}^{n} y_h},$$

where y_i is the cash dividend yield of company com_i.

STEP 2: Calculate cumulative probability using the selection probability of each company:

$$C(y_i) = \sum_{j=1}^{i} P(y_j).$$

STEP 3: Generate the initial population using the following sub-steps:

Sub-step 3.1: Randomly generate *numCom* values in the range [0, 1] and collect them in a set $S = \{r_k \mid 1 \le k \le numCom\}$.

Sub-step 3.2: For each element in S, if the random value r_k is between $C(y_{j-1})$ and $C(y_j)$, then company com_j is put into the candidate portfolio. Note that a company com_j can only appear in chromosome once. The duplicate companies are removed directly.

Sub-step 3.3: Generate a chromosome according to the candidate portfolio. For each selected company com_i, set its b_i in the chromosome to larger than 0.5. Otherwise, set the non-selected company less than 0.5. Randomly generate the corresponding number of purchased units of each company in the range [0, *maxUnit*].

Sub-step 3.4: If P chromosomes are generated, go to the next step. Otherwise, go to Sub-step 3.1.

Step 4: Calculate the fitness value of each chromosome using the following sub-steps:

Sub-step 4.1: Calculate the profit of each chromosome C_q using Eq. (6).

Sub-step 4.2: Calculate the risk of each chromosome using the HS method with the predefined confidence level λ.

Sub-step 4.3: Calculate the suitability of each chromosome using Eq. (9).

Sub-step 4.4: Set the fitness value of each chromosome C_q using Eq. (5).

Step 5: Execute the selection operation on the population to form the next population. Here, elitist or roulette wheel selection strategies can be used. In this paper, the elitist selection strategy is utilized for evaluating the results of the proposed approach.

Step 6: Execute the one-point crossover operation on the population.

Step 7: Execute the one-point mutation operation on the population.

Step 8: If the stop criterion is satisfied, go to the next step. Otherwise, go to step 4.

Step 9: Output the chromosome with the best fitness value.

6.2. An Example

In this section, an example is given to illustrate the proposed algorithm for finding an ASP. Assume that there are 15 companies. Their related data for 2007 are shown in Table 4.

Table 4. Companies used in example

com_i	*Buying price (2007/1/2)*	*Selling price (2007/12/31)*	*Average of three-year cash dividend yield (y_i)*	*Cash dividend (2007)*
com_1	62	67.7	3.02%	5.6
com_2	97.4	90.3	6.77%	5.31
com_3	86	55.2	2.57%	2.74
com_4	91.1	54.4	4.43%	4.93
com_5	43.5	34.85	1.85%	7.62
com_6	32.5	36.95	1.22%	2.38
com_7	202	237.5	2.60%	7.36
com_8	38.05	28.5	1.60%	9.75
com_9	35.5	29.1	2.28%	7.36
com_{10}	12.8	16	0.43%	3.45
com_{11}	23.95	25.2	0.8%	7.37
com_{12}	23.05	27.35	1.08%	8.43
com_{13}	58.3	50.5	2%	3.96
com_{14}	34.85	28.65	0.87%	4.23
com_{15}	63.5	68.6	2.95%	1.98

Step 1: The selection probability of each company is calculated according to its y_i. Take company com_2 as an example. Since the sum of the cash dividend yields of all companies is 34.47, the selection probability of com_2 is 0.1964 (= 6.77 / 34.47). The results of all companies are shown in Table 5.

Table 5. Selection probabilities of all companies

com_i	$P(y_i)$	com_i	$P(y_i)$	com_i	$P(y_i)$
com_1	0.0876	com_6	0.0353	com_{11}	0.0232

com_2	0.1964	com_7	0.0754	com_{12}	0.0313
com_3	0.0745	com_8	0.0464	com_{13}	0.058
com_4	0.1285	com_9	0.0661	com_{14}	0.0252
com_5	0.0536	com_{10}	0.0124	com_{15}	0.0855

Step 2: The cumulative probability of each company is then calculated. The results are used to generate the initial population. They are shown in Table 6.

Table 6. Cumulative probabilities of all companies

com_i	$C(y_i)$	com_i	$C(y_i)$	com_i	$C(y_i)$
com_1	0.088	com_6	0.576	com_{11}	0.799
com_2	0.284	com_7	0.651	com_{12}	0.831
com_3	0.359	com_8	0.698	com_{13}	0.889
com_4	0.487	com_9	0.764	com_{14}	0.914
com_5	0.541	com_{10}	0.776	com_{15}	1.000

Step 3: The initial population is generated using the following sub-steps.

Sub-step 3.1: Since *numCom* is set at 10, ten random numbers in the range [0, 1] are generated and collected in a set S = {0.08, 0.2, 0.3, 0.57, 0.6, 0.68, 0.77, 0.78, 0.85, 0.9}.

Sub-step 3.2: According to the elements in S, the following ten companies are selected to form a candidate portfolio: com_1, com_2, com_3, com_6, com_7, com_8, com_{10}, com_{11}, com_{13}, and com_{14}.

Sub-steps 3.3 and 3.4: The number of purchased units of each selected company is generated in the range [0, *maxUnit*]. After Sub-step 3.4**,** assume that the initial population is generated as follows:

C_1: (0.724, 8, 0.599, 10, 0.541, 8, 0.357, 5, 0.115, 9, 0.911, 7, 0.861, 7, 0.726, 3, 0.415, 5, 0.690, 4, 0.879, 2, 0.267, 6, 0.808, 6, 0.514, 7, 0.282, 5);
C_2: (0.980, 10, 0.357, 8, 0.674, 1, 0.600, 7, 0.949, 8, 0.184, 4, 0.961, 6, 0.547, 8, 0.007, 3, 0.962, 5, 0.236, 2, 0.588, 10, 0.978, 4, 0.293, 5, 0.728, 2);
C_3: (0.260, 6, 0.855, 4, 0.336, 6, 0.516, 2, 0.725, 5, 0.803, 9, 0.927, 6, 0.560, 4, 0.614, 4, 0.795, 6, 0.638, 6, 0.336, 5, 0.292, 9, 0.798, 2, 0.035, 8);
C_4: (0.517, 8, 0.180, 4, 0.327, 8, 0.926, 2, 0.977, 10, 0.395, 5, 0.853, 9, 0.795, 9, 0.977, 5, 0.562, 5, 0.017, 9, 0.603, 5, 0.849, 2, 0.353, 6, 0.798, 7);
C_5: (0.510, 2, 0.928, 7, 0.492, 5, 0.529, 5, 0.073, 3, 0.638, 4, 0.862, 6, 0.905, 6, 0.668, 7, 0.190, 2, 0.514, 5, 0.384, 2, 0.770, 3, 0.282, 7, 0.524, 10)
C_6: (0.885, 9, 0.071, 1, 0.620, 3, 0.795, 3, 0.689,1, 0.489, 7, 0.536, 6, 0.044, 8, 0.937, 9, 0.374, 5, 0.934, 9, 0.789, 8, 0.591, 3, 0.791, 6, 0.482, 6)
C_7: (0.744, 8, 0.523,10, 0.477, 10, 0.611, 9, 0.753, 6, 0.283, 9, 0.136, 6, 0.790, 2, 0.989, 5, 0.031, 8, 0.719, 10, 0.715, 6, 0.406, 10, 0.963, 5, 0.830, 7)
C_8: (0.793, 5, 0.084, 6, 0.996, 10, 0.405, 8, 0.847, 5, 0.648, 9, 0.875, 4, 0.474, 1, 0.517, 5, 0.720, 3, 0.786, 8, 0.619, 1, 0.425, 4, 0.681, 1, 0.317, 5)

C_9: (0.289, 7, 0.639, 5, 0.010, 3, 0.990, 10, 0.820, 9, 0.083, 3, 0.696, 10, 0.984, 10, 0.164, 5, 0.854, 2, 0.717, 6, 0.778, 6, 0.005, 10, 0.519, 8, 0.620, 7)
C_{10}: (0.549, 1, 0.486, 3, 0.811, 2, 0.745, 8, 0.355, 4, 0.032, 3, 0.289, 5, 0.525, 9, 0.056, 5, 0.834, 7, 0.909, 3, 0.738, 9, 0.165, 6, 0.952, 2, 0.551, 2)

Step 4: The fitness value of each chromosome is calculated using the following sub-steps.

Sub-step 4.1: The profit of each chromosome is calculated. Take C_1 as an example. Using Eq. (6), $ROI(C_1)$ is calculated as 0.082 (= 277.11 / 3378.75). The profit values of all chromosomes are shown in Table 7.

Table 7. Profit values of all chromosomes

C_q	$ROI(C_q)$	C_q	$ROI(C_q)$
C_1	0.082	C_6	0.321
C_2	0.232	C_7	0.200
C_3	0.149	C_8	0.149
C_4	0.248	C_9	0.293
C_5	0.186	C_{10}	0.058

Sub-step 4.2: The risk of each portfolio is then calculated. Since the confidence level is set at 99%, the first and second VaR values of company com_1 in C_1 are -0.082 and -0.073, respectively. The number of purchased units of com_1 is 8. The risk value of com_1 is -0.619 (= ((-0.082 + (-0.073)) / 2 × 8). The risk values of the other companies can be similarly calculated. Thus, $Risk(C_1)$ is set at -4.684, which is the summation of risks of all stocks in the portfolio. The risk values of all chromosomes are shown in Table 8.

Table 8. Risk values of all chromosomes

C_q	$Risk(C_q)$	C_q	$Risk(C_q)$
C_1	-4.684	C_6	-5.266
C_2	-5.340	C_7	-6.135
C_3	-3.963	C_8	-4.323
C_4	-5.557	C_9	-6.370
C_5	-5.143	C_{10}	-3.918

Sub-step 4.3: The suitability criterion of a chromosome consists of *ICP* and *PP*. Take C_1 as an example. Assume that *maxInves* and *numCom* are set at 1000 and 10, $ICP(C_1)$ and $PP(C_1)$ are calculated as 3.379 (= 3378.75 / 1000) and 1.5 (= 15 / 10),

respectively. *suitability* (C_1) is 4.879 (= 3.379 + 1.5).

Sub-step 4.4: The fitness value of chromosome C_1 is set at $[(ROI(C_1) + Risk(C_1))]^{\alpha}$ / *suitability*(C_1), which is -0.943 (= (0.082 + (-4.684)) / 4.879) assuming that α is set at 1. The fitness values of all chromosomes after Step 4 are shown in Table 9.

Table 9. Fitness values of all chromosomes

C_q	$f(C_q)$	C_q	$f(C_q)$
C_1	-0.943	C_6	-1.369
C_2	-1.009	C_7	-1.299
C_3	-0.823	C_8	-0.999
C_4	-1.059	C_9	-1.126
C_5	-1.273	C_{10}	-0.798

Steps 5 to 7: The genetic operations are executed on the population. For crossover operation, take C_3 and C_{10} as an example, let a cut point is between com_{13} and com_{14}, the two new chromosomes C_3' and C_{10}' are generated as follows:

C_3': (0.260, 6, 0.855, 4, 0.336, 6, 0.516, 2, 0.725, 5, 0.803, 9, 0.927, 6, 0.560, 4, 0.614, 4, 0.795, 6, 0.638, 6, 0.336, 5, 0.292, 9, **0.952, 2, 0.551, 2**);
C_{10}': (0.549, 1, 0.486, 3, 0.811, 2, 0.745, 8, 0.355, 4, 0.032, 3, 0.289, 5, 0.525, 9, 0.056, 5, 0.834, 7, 0.909, 3, 0.738, 9, 0.165, 6, **0.798, 2, 0.035, 8**).

For mutation operation, take C_{10}' as an example. Let the b_i of com_4 is mutated to 0.025 and the u_i of com_8 is mutated to 10, the new chromosome C_{10}'' is shown below.

C_{10}'': (0.549, 1, 0.486, 3, 0.811, 2, **0.025**, 8, 0.355, 4, 0.032, 3, 0.289, 5, 0.525, **10**, 0.056, 5, 0.834, 7, 0.909, 3, 0.738, 9, 0.165, 6, 0.798, 2, 0.035, 8);

After genetic operations, three new chromosomes, C_3', C_{10}', C_{10}'', are generated. Then, after first generation, totally thirteen chromosomes could be competed with each other to survive in next generation.

Steps 8 to 9: If the stop condition is not satisfied, Steps 4 to 8 are repeated. Otherwise, the chromosome (ASP) with the best fitness value is outputted. From Table 4, the selling prices are smaller than the buying prices of stocks, and thus it is reasonable that the fitness values of chromosomes (Table 9) in the initial population are negative. The fitness values of chromosomes will be improved after generations.

7. EXPERIMENTAL RESULTS

In this section, experiments are conducted to show the performance of the DDSPO approach. The experiments were implemented in Java on a personal computer with an Intel Core i5 3.19-GHz CPU and 4.0 GB of RAM. The initial population size P was set at 80, the crossover rate p_c was set at 0.8, the mutation rate p_m was set at 0.06, and the number of generations G was set at 10000. The maximum number of purchased stocks in the portfolio *numCom* was set at 15, the predefined maximum investment capital *maxInves* was set at 2 million, the maximum number of purchased units of a stock *maxUnit* was set at 10, parameter α was set at 1, and the confidence level was set at 99%. The experimental dataset is described in Section 7.1. Then, the effectiveness of the designed fitness function is shown in Section 7.2. An analysis of the derived ASP is given in Section 7.3.

7.1 Data descriptions

The experimental dataset was collected from the TSE from 2007/01/01 to 2011/12/31. The stocks used in the experiments were selected from an exchange traded fund, namely the Taiwan Top 50 Traded Fund (TF50). According to expertsô evaluations, stocks in the TF50 may slightly change over time. The 45 stocks always included in the TF50 were used for experiments. The dataset contains the stock prices of 45 stocks from 2007/01/01 to 2011/12/31 and the cash dividend yields of all stocks. In the experiments, the data from 2007/01/01 to 2007/12/31 were used as the training dataset, and the remaining data were used as the testing dataset.

7.2 Effectiveness of designed fitness function

The experiments were first conducted to show the fitness convergence results of the proposed approach. The average fitness values after 10000 generations are shown in

Figure 3.

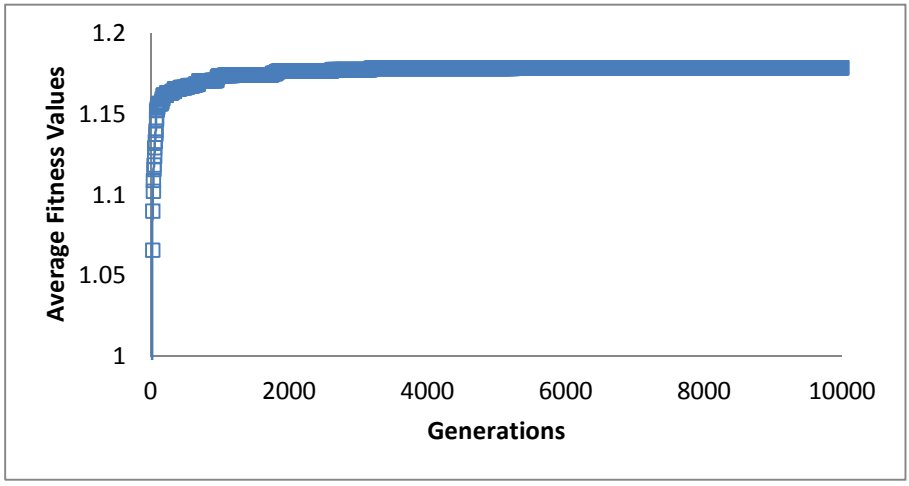

Figure 3. Fitness convergence results of proposed approach.

In Figure 3, the average fitness values of the proposed approach increase with increasing number of generations, and converge to a certain value around 3500 generations. (Note that different datasets may need different numbers of generations to reach convergence.) Thus, the results show that the designed fitness function can produce stable results. Since the average fitness results are positive, the designed fitness function can be used to derive the ASP with the maximum ROI while satisfying the usersô requests (e.g., the maximum number of purchased stocks, the maximum investment capital, and the maximum number of purchased units). More details are given in the following experiments.

Experiments were then conducted to show the merits of the designed fitness function. Three types of fitness function were used in the proposed approach. They are F1: the designed fitness function (= (*roi*+*risk*) / *suitability*), F2: *roi*+*risk* as fitness function, and F3: *suitability* as fitness function. Comparison results of the three fitness functions in terms of the average *roi*+*risk* value are shown in Figure 4.

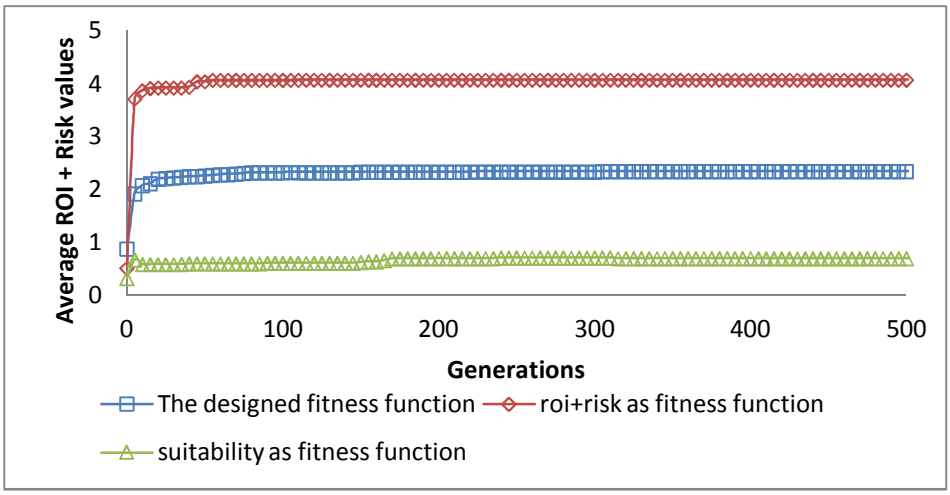

Figure 4. Comparison results of three fitness functions in terms of average *roi+risk* value.

Figure 4 shows that using F2 and F3 as the fitness functions yields the largest and smallest average *roi+risk* values, respectively. When using F2 as the fitness function, the proposed approach focuses on deriving an ASP with the highest profit (its suitability is poor, see Figure 5). When using F3 as the fitness function, the proposed approach focuses on deriving an ASP that satisfies the investorôs subjective requests. Thus, its profit is the worst. When using F1 (the designed fitness function) as the fitness function, the profit of the derived ASP is between those of F2 and F3. Although the ASP derived using F1 as the fitness function does not have the highest profit, it yields the best ASP when considering the investorôs objective and subjective requests (see Figure 6). The comparison results of the three fitness functions in terms of average suitability are shown in Figure 5.

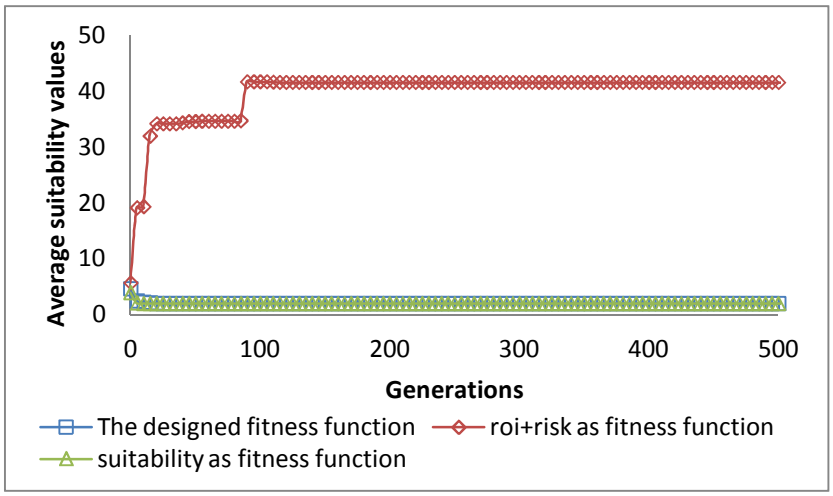

Figure 5. Comparison results of three fitness functions in terms of average suitability value.

Figure 5 shows that the average suitability values obtained using F1 and F3 are better than those obtained using F2. In other words, although using F2 as the fitness function can derived an ASP with the largest profit, it does not satisfy the user's requests. Although the ASP derived using F3 has the best average suitability values, its profit is the worst (see Figure 4). When using F1 as the fitness function, the average suitability values are similar to those for F3. Comparison results of the three fitness functions in terms of average designed fitness value are shown in Figure 6.

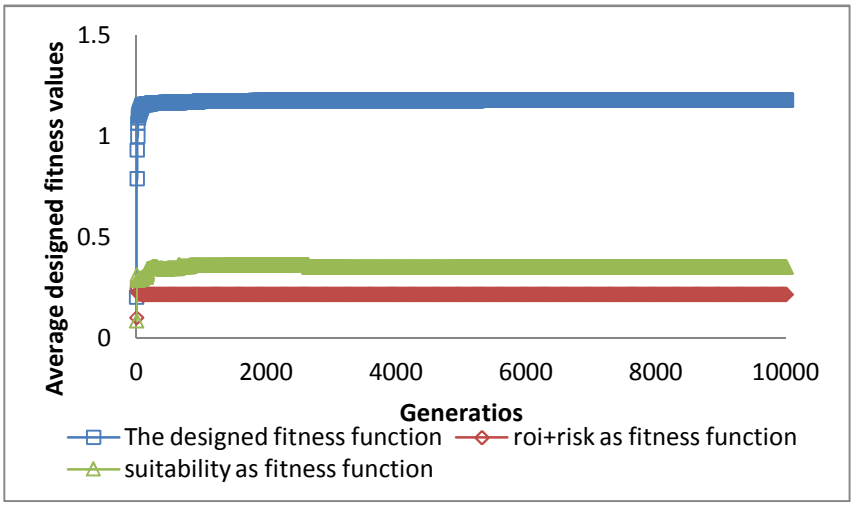

Figure 6. Comparison results of three fitness functions in terms of average designed fitness value.

Figure 6 shows that using F1 as the fitness function produces the best average designed fitness values. Thus, from Figures 4, 5, and 6, it can be concluded that using

the proposed approach with the designed fitness function (F1) yields the ASP that satisfies the user's subjective requests (e.g., maximum number of purchased stocks) as well as objective requests (e.g., maximum profit).

7.3 Analysis of derived ASP

In the previous section, the experimental results showed the effectiveness of the designed fitness function. In this section, the derived ASP is analyzed. Table 10 shows the initial portfolio and ASP derived using the proposed approach with the designed fitness functions.

Table 10. Initial portfolio and ASP derived using DDSPO approach

Initial portfolio				
2311(2)	2324(6)	2801(1)	2912(8)	2881(9)
Derived ASP				
2311(10)	2324(10)	2880(1)	2888(1)	2801(1)
1216(9)	2201(1)	1326(1)	2886(1)	1802(3)
2912(1)	2881(10)	4904(10)	2885(1)	2890(1)

From Table 10, the initial portfolio contains five companies, whose stock symbols are 2311, 2324, 2801, 2912, and 2881. The purchased units of these companies are 2, 6, 1, 8, and 9, respectively. Since the predefined maximum number of purchased stocks and the maximum number of purchased units of a stock were set at 15 and 10, respectively, the predefined maximum number of purchased stocks of the initial stock portfolio does not satisfy the investor's request. After 10000 generations, the derived ASP contains fifteen companies (stock symbols: 2311, 2324, 2880, 2888, 2801, 1216, 2201, 1326, 2886, 1802, 2912, 2881, 4904, 2885, and 2890). The number of purchased units of each stock is less than 10. The derived ASP thus satisfies the investor's subjective requests.

As to the investor's objective requests, the profit of the ASP (average results of five runs) derived using the DDSPO approach with different training and testing periods are shown in Table 11.

Table 11. Profit of derived ASP with different training and testing periods

Training period	One year (2007/01 to 2007/12)			
Testing period	2008/01 ~2008/12	2008/01 ~2009/12	2008/01 ~2010/12	2008/01 ~2011/12
Return (%)	-30.0	13.7	32.0	26.41
Training period	Two years (2007/01 to 2008/12)			
Testing period	2009/01~12	2009/01~2010/12	2009/01~2011/12	
Return (%)	46.1	62.4	55.1	

When the training period was one year (2007/01 to 2007/12), the derived ASP had returns of -30%, 13%, 32%, and 26.41% in periods of 2008/01 to 2008/12, 2008/01 to 2009/12, 2008/01 to 2010/12, and 2008/01 to 2011/12, respectively. They are all positive returns except that for 2008/01 to 2008/12, which is due to the 2008 financial crisis. When the training period was two years, the returns of the derived ASP for the periods of 2009/01 to 2009/12, 2009/01 to 2010/12, and 2009/01 to 2011/12 are 46.1%, 62.4%, and 55.1%, respectively. Thus, it can be concluded that the DDSPO approach is effective in deriving ASPs that satisfy the investorôs objective and subjective requests.

Finally, experiments were conducted to compare the DDSPO approach with and without generating the initial population using the cash dividend yields (CDY) of companies. Comparison results in terms of average fitness value are shown in Figure 7.

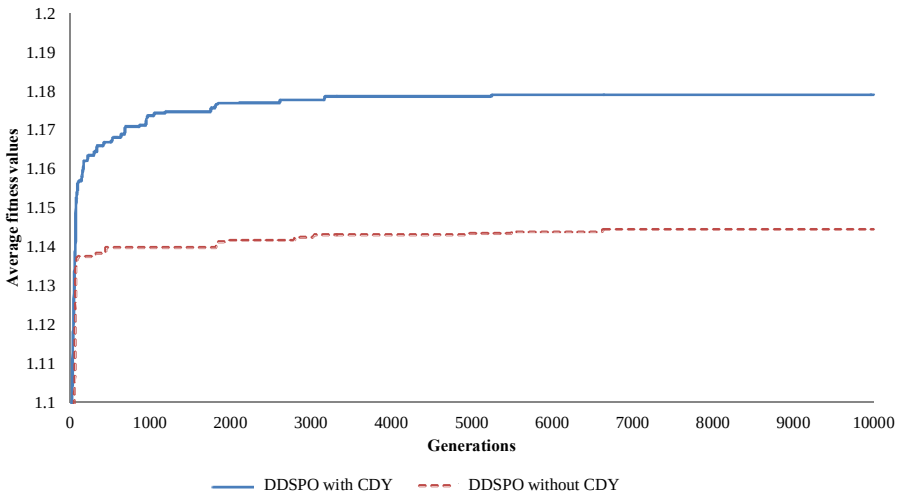

Figure 7. DDSPO approach with and without CDY of companies.

Figure 7 shows that the average fitness values of the DDSPO approach with CDY are better than those of the DDSPO approach without CDY. These results indicate that utilizing the meta knowledge (cash dividend yields of companies) improves the effectiveness of the DDSPO algorithm.

8. DISCUSSIONS

Portfolio mining is an attractive research topic for researchers since various factors which could be divided into objective interestingness and subjective interestingness should be considered to achieve acceptable results. The objective interestingness is like return on investment and value at risk. The subjective interestingness contains requests from investors like maximum capital investment and number of purchased stocks.

In MV model, given the expected return, standard deviation (risk) of each asset, and the correlation matrix between these assets that are objective interestingness, the goal of MV model is to find the efficient frontier. In other words, the efficient frontier means a set of portfolios that reaches certain expected return, or keeps the risk under a predefined threshold. As mentioned in related work, many optimization approaches have been proposed to find the portfolios. Hence, when investors exactly know the stocks they want to purchase, MV-model could be utilized to get the portfolio.

The DDSPO approach, however, takes both objective (return and risk) and subjective (requests from investors) interestingness into consideration for deriving a portfolio by genetic algorithms. The fitness value of each chromosome is evaluated by the profit and suitability of a chromosome. The profit of a chromosome is used to measure the objective interestingness. The suitability of a chromosome is designed to appraise the subjective interestingness. Thus, when investors have their own objective and subjective interestingness, the proposed approach could be suggested and used to

derive the portfolio.

9. CONCLUSION AND FUTURE WORK

Based on the D^3M concept, this study proposed the DDSPO framework that takes meta knowledge, domain knowledge, and subjective and objective interestingness into consideration for deriving ASPs. Then, based on the proposed framework, the DDSPO approach was proposed for mining ASPs. The DDSPO approach considers the subjective and objective requests of investors when designing the fitness function. Experiments on a real dataset showed that the designed fitness function can be utilized for mining ASPs. Then, experiments showed that the derived ASP satisfies the investorôs subjective requests. Finally, experiments indicated that utilizing the meta knowledge (cash dividend yields of companies) improves the effectiveness of the DDSPO algorithm.

In the future, many topics will be considered. For example, the DDSPO approach can be extended to other D^3M frameworks (e.g., CM-AKD, MSCM-AKD). In addition, since stocks may have some kinds of similarity, the proposed approach can be extended to group-based ASP optimization approaches. Since each request of users is an objective function, the proposed approach can also be extended to a MOGA-based approach by utilizing MOGA techniques (e.g., SPEA2, NSGAII) for deriving the Pareto solutions.

ACKNOWLEDGMENT

This research was supported by the National Science Council of the Republic of China under grant NSC 101-2221-E-032-057.

REFERENCES

[1] M. H. Babaei, M. Hamidi, E. Jahani and H. Pasha, "A new approach to solve an extended

portfolio selection problem," *International Conference on Industrial Engineering and Operations Management,* pp. 1954-1960, 2012.
[2] V. Bevilacqua, V. Pacelli and S. Saladino, "A novel multi objective genetic algorithm for the portfolio optimization," *Advanced Intelligent Computing*, pp. 186-193, 2012.
[3] J. Bermúdez, J. Segura and E. Vercher, "A multi-objective genetic algorithm for cardinality constrained fuzzy portfolio selection," *Fuzzy Sets and Systems*, Vol. 188, pp. 16-26, 2012.
[4] L. Cao, "Domain driven data mining (D^3M)," *IEEE International Conference on Data Mining Workshops*, pp. 74-76, 2008.
[5] L. Cao, "Domain-driven data mining: challenges and prospects," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 22, No. 6, pp.755-769, 2010.
[6] L. Cao, P. S. Yu, C. Zhang and Y. Zhao, ñDomain driven data mining,ò *Springer*, 2010.
[7] L. Cao, "Actionable knowledge discovery and delivery," *WIREs Data Mining and Knowledge Discovery*, Vol. 2, No. 2, pp. 149-163, 2012.
[8] J. S. Chen, J. L. Hou, S. M. Wu and Y. W. Chang-Chien, "Constructing investment strategy portfolios by combination genetic algorithms," *Expert Systems with Applications*, Vol. 36, pp. 3824-3828, 2009.
[9] T. J. Chang, S. C. Yang and K. J. Chang, "Portfolio optimization problems in different risk measures using genetic algorithm," *Expert Systems with Applications*, Vol. 36, pp. 10529-10537, 2009.
[10] L. Cao, H. Zhang, Y. Zhao, D. Luo and C. Zhang, "Combined mining: discovering informative knowledge in complex data," *The IEEE Transactions on Systems, Man, and Cybernetics, Part B*, Vol. 41, No. 3, pp. 699 - 712, 2011.
[11] L. Cao, Y. Zhao, H. Zhang, D. Luo, C. Zhang, and E. K. Park, "Flexible frameworks for actionable knowledge discovery," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 22, No. 9, pp. 1299-1312, 2010.
[12] Z. G. M. Elhachlouý, F. Hamza, "Stocks Portfolio Optimization Using Classification and Genetic Algorithms," *Applied Mathematical Sciences*, Vol. 6, pp. 4673-4684, 2012.
[13] P. Gupta, M. K. Mehlawat and G. Mittal, "Asset portfolio optimization using support vector machines and real-coded genetic algorithm," *Journal of Global Optimization*, Vol. 53, pp. 297-315, 2012.
[14] P. Gupta, M. K. Mehlawat and A. Saxena, "Hybrid optimization models of portfolio selection involving financial and ethical considerations," *Knowledge-Based Systems*, Vol. 37, pp. 318-337, 2012.
[15] D. E. Goldberg, "Genetic algorithms in search, optimization, and machine learning," *Addison Wesley*, 1989.
[16] J. J. Grefenstette, "Optimization of control parameters for genetic algorithms," *IEEE Transactions on System Man, and Cybernetics*, Vol. 16, pp. 122-128, 1986.
[17] J. H. Holland, "Adaptation in natural and artificial systems," *University of Michigan Press*, 1975.
[18] F. Hassanzadeh, M. Collan and M. Modarres, "A practical approach to R&D portfolio selection using the fuzzy pay-off method," *IEEE Transactions on Fuzzy Systems*, Vol. 20, pp. 615-622, 2012.
[19] L. R. Z. Hoklie, "Resolving multi objective stock portfolio optimization problem using genetic algorithm," *International Conference on Computer and Automation Engineering*, pp. 40-44, 2010.
[20] R. Kumar and S. Bhattacharya, "Cooperative search using agents for cardinality constrained portfolio selection problem," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, Vol. 42, pp. 1510 - 1518, 2012.
[21] H. Kellerer, R. Mansini and S. M. Grazia, "Selecting portfolios with fixed costs and minimum transaction lots," *Annals of Operations Research*, Vol. 99, pp. 287-304, 2000.
[22] H. Konno and A. Wijayanayake, "Portfolio optimization problem under concave transaction costs and minimal transaction unit constraints," *Mathematical Programming*, Vol. 89, pp. 233-250, 2001.
[23] P. C. Lin, "Portfolio optimization and risk measurement based on non-dominated sorting genetic algorithm," *Journal of Industrial and Management Optimization*, Vol. 8, pp. 549-564, 2012.
[24] C. C. Lin and Y. T. Liu, "Genetic algorithms for portfolio selection problems with minimum transaction lots," *European Journal of Operational Research*, Vol. 185, pp. 393-404, 2008.
[25] J. Li and J. Xu, "Multi-objective portfolio selection model with fuzzy random returns and a compromise approach-based genetic algorithm," *Information Sciences*, Vol. 220, pp. 507-521, 2012.
[26] H. M. Markowitz, "Harry Markowitz: Selected Works," *World Scientific Publishing Company*, 2009.
[27] R. Mansini and M. G. Speranza, "Heuristic algorithms for the portfolio selection problem with minimum transaction lots," *European Journal of Operational Research*, Vol. 114, pp. 219-233, 1999.
[28] R. Mansini and M. G. Speranza, "An exact approach for portfolio selection with transaction costs and rounds," *IIE transactions*, Vol. 37, pp. 919-929, 2005.
[29] T. P. Patalia and G. Kulkarni, "Design of genetic algorithm for knapsack problem to perform stock portfolio selection using financial indicators," *International Conference on Computational Intelligence and Communication Networks*, pp. 289-292, 2011.
[30] A. D. Roy, "Safety first and the holding of assets," *Econometrica*, Vol. 20, pp. 431-449, Jul. 1952.
[31] H. Soleimani, H. R. Golmakani and M. H. Salimi, "Markowitz-based portfolio selection with

minimum transaction lots, cardinality constraints and regarding sector capitalization using genetic algorithm," *Expert Systems with Applications*, Vol. 36, pp. 5058-5063, 2009.
[32] A. T. H. Sim, M. Indrawan, S. Zutshi and B. Srinivasan, "Logic-based pattern discovery," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 22, No. 6, pp. 798-811, 2010.
[33] E. Vercher, "A possibilistic mean-downside risk-skewness model for efficient portfolio selection," *IEEE Transactions on Fuzzy Systems*, Vol. 21, pp. 585 - 595, 2011.
[34] E. Wah, Y. Mei and B. W. Wah, "Portfolio optimization through data conditioning and aggregation," *IEEE International Conference on Tools with Artificial Intelligence*, pp. 253-260, 2011.
[35] Y. Yue, Y. Wan, R. Jia and L. Jiang, ñDomain driven data mining for customer demand discovery,ò *International Conference on Service Systems and Service Management*, pp. 1-6, 2014.