

Optimal Portfolio Insurance under Nonlinear Transaction Costs

Vasilios N. Katsikis, Spyridon D. Mourtas

Department of Economics, Division of Mathematics and Informatics, National and Kapodistrian University of Athens, Sofokleous 1 Street, 10559 Athens, Greece
E-mail: vaskatsikis@econ.uoa.gr; spirosmourtas@gmail.com

Received: 12 August 2020; Accepted: 11 September 2020; Available online: 15 October 2020

Abstract: The minimization of the costs related to portfolio insurance is a very important investment strategy. In this article, by adding the transaction costs to the classical minimum cost portfolio insurance (MCPI) problem, we define and study the MCPI under transaction costs (MCPITC) problem as a nonlinear programming (NLP) problem. In this way, the MCPI problem becomes more realistic. Since such NLP problems are commonly solved by heuristics, we use the Beetle Antennae Search (BAS) algorithm to provide a solution to the MCPITC problem. Numerical experiments and computer simulations in real-world data sets confirm that our approach is an excellent alternative to other evolutionary computation algorithms.

Keywords: Portfolio insurance; Transaction costs; Nonlinear programming.

1. Introduction

Investment costs and fees can have a significant effect on the average returns on portfolios, particularly over the longer term. Paying lower costs means that more return on your investment can flow to you, allowing you to reinvest more money and compound it into the future. Consequently, minimizing fees and costs form a crucial focus of the strategy of each investor. For example, a Markowitz-based portfolio selection with minimum transaction lots, cardinality constraints and regarding sector capitalization problem is presented in [1]. To solve the mixed-integer nonlinear programming (NP-Hard) problem therein, a corresponding genetic algorithm is utilized. In [2], the authors develop and examine a mixed-method, integrated qualitative and quantitative portfolio selection model, applied to a “tritium extraction facility” project concept. The model therein was developed by building on an existing analytical hierarchy process theory using statistical principles and was successfully applied to a nuclear tritium extraction facility case study.

In this paper, we define and study the minimum-cost portfolio insurance (MCPI) problem as a two-period model. The dominant financial problem here is the MCPI under transaction costs (MCPITC) problem, which is resulting from the combination of the MCPI problem and the portfolio selection under transaction costs (PSTC) problem. The MCPITC is an NLP problem, we approach it with an altered acceptance of the Beetle Antennae Search (BAS) algorithm and we compare the results with the results of some of the best evolutionary computation algorithms at present. These algorithms are the Shuffled Frog Leaping Algorithm (SFLA) from [3], the standard Firefly Algorithm (FA) from [4] and the Genetic Algorithm (GA) MATLAB function. Note that we apply our modified version of BAS in a MATLAB environment.

Beetle Antennae Search is a nature inspired algorithm that is competent of efficient global optimization and has been extensively used in several scientific fields in recent years (see [5-9]). For example, in [7], a constrained portfolio optimization problem is tackled by an altered version of BAS, named Quantum Beetle Antennae Search (QBAS). Therein, the authors carved BAS into the quantum realm and used quantum mechanics to formulate QBAS, as quantum computing outsmart traditional computing in efficiency and speed. In [9], by combining a back propagation neural network (BPNN) and BAS, an intelligent technique was presented for predicting the unconfined compressive strength (UCS) of the coalcrete. More precisely, the BPNN's architecture was initially tuned by BAS, and afterwards, the optimized BPNN-BAS model was used for nonlinear relationship modeling. Note that BAS is directly applicable only to unconstrained optimization. Inhere, our modified version of BAS keeps the solutions in the feasible region by using the penalty function method. In this way, BAS is able to manage convex or nonconvex constraints more effectively. Generally, the advantages of BAS are that it is less time consuming than the previously mentioned evolutionary algorithms, with akin efficacy.

The main contributions of this paper can be summarised as follows:

- 1) the definition and the study of the MCPITC problem, as a NLP problem,
- 2) the modification of BAS, which enables it to handle constrained optimization problems,

3) the comparison between BAS, SFLA, FA and GA on a NLP problem in the field of finance.

Main results of the present paper are summarized as follows. Section 2 describes the MCPI, PSTC and the MCPITC optimization problems. In addition, a suitably altered meta-heuristic algorithm addresses the MCPITC optimization problem. Section 3 comprises of the numerical experiments, using data from the real world and testing the efficacy of BAS, SFLA, FA and GA in different portfolios setups. Finally, section 4 presents the concluding remarks.

2. Optimal insured portfolio under nonlinear transaction costs

The reduction of the insurance cost is one way to minimize the expenditure of a portfolio (see [10-18]). For example, based on Meton’s optimal investment-consumption model, a continuous-time portfolio insurance model is presented in [16]. The results therein show that investors’ optimal strategies of portfolio insurance are not dependent on their wealth, but on the market risk. In [18], assuming that investors’ preferences are expressed in terms of target sets to be reached at each time period over a specified finite horizon, a portfolio construction proposed and approached by stochastic reachability. In general, a portfolio insurance strategy is a dynamic hedging process which gives the investor the potential to limit downside risk while allowing upside participation to maximize a portfolio's terminal value over a given investment horizon.

In this section, we define the MCPI, the PSTC and the MCPITC problems. The MCPI problem minimizes a portfolio's insurance costs and at the same time keeps its payoff over a price on the market. The PSTC problem minimizes the transaction costs of the portfolio while at the same time trying to get an optimal payoff. If we consider the MCPI problem along with the PSTC problem, then we can identify the MCPITC problem, which minimizes both the cost of the portfolio insurance and its transaction costs and, at the same time, attempts to capture the optimal payoff while keeping the payoff above the floor price. In addition, we can solve the MCPITC problem by an altered version of the BAS algorithm.

2.1 MCPITC problem definition

We consider the financial problem of portfolio insurance as in [6, 7, 10-14]. The marketed securities space is $X = [x_1, x_2, \dots, x_n] \in \mathbb{R}^{m \times n}$ where $x_i \in \mathbb{R}^m$ denotes the i -security, for $i = 1, 2, \dots, n$, and contains information from m consecutive measurements on its price. A portfolio is a vector $\theta = (\theta_1, \theta_2, \dots, \theta_n)$ of \mathbb{R}^n where θ_i denotes the number of shares of the i -security. If $\theta = (\theta_1, \theta_2, \dots, \theta_n)$ is the initial portfolio that is not zero at the first period, then the following formula gives its payoff at the second period:

$$G(\theta) = \sum_{i=1}^n \theta_i x_i, \tag{1}$$

where $x_i \in \mathbb{R}^m$, $i = 1, 2, \dots, n$. The G operator is called the payoff operator and is one-to-one. If we also have a “floor” price $\phi \in \mathbb{R}$ then the payoff after insurance price p on the θ portfolio at the ϕ “floor” is the supremum $G(\theta) \vee \phi \cdot \mathbf{1}$, where $\mathbf{1} \in \mathbb{R}^m$ is a vector comprises of ones and “ \vee ” denotes the supremum operator. The minimum-cost portfolio insurance η at the floor ϕ and the insurance price p is the solution to the MCPI problem:

$$\min_{\eta} \quad p^T \cdot \eta \tag{2}$$

$$\text{subject to} \quad X \cdot \eta \geq G(\theta) \vee \phi \cdot \mathbf{1}. \tag{3}$$

The problem (2)-(3) can be rewritten in a LP form as follows:

$$\min_{\eta} \quad p^T \cdot \eta \tag{4}$$

$$\text{subject to} \quad -X \cdot \eta \leq \min\{-G(\theta), -\phi \cdot \mathbf{1}\} \tag{5}$$

$$\mathbf{0} \leq \eta \leq X(1, \cdot) \cdot \theta \cdot \left[\frac{1}{x_1(1)}, \dots, \frac{1}{x_n(1)} \right]^T \tag{6}$$

where $\mathbf{0} \in \mathbb{R}^n$ is a vector comprises of zeros. Furthermore, the last part of (6) is the maximum amount of every stock that an investor can hold by investing all the portfolio’s payoff into each one of them. Also, $x_i(1)$ denotes the first element of the vector x_i , $i = 1, \dots, n$. The optimal insured portfolio of (4)-(6) is equal to $\eta = [\eta_1, \dots, \eta_n]$.

In addition, transaction costs can be used to define a variety of costs. In a financial sense, transaction costs include brokerage commissions, fund loads, taxes, bid-ask spreads, etc. By setting the sum of the transaction costs relating to individual transactions as

$$\kappa = \sum_{i=1}^n \kappa_i, \tag{7}$$

where κ_i is the function of transaction cost for the asset i , we assume that the cost of transaction is separable as presented in [19]. Considering that a fixed price is common, for any non-zero trade and that there may be one or more situations at which the cost of the transaction per share will reduce in value, we consider a mere model that involves fixed plus linear costs. Thus, the buying cost and the selling cost would possibly be concave. By setting the cost rates associated with buying and selling asset i as γ^+ , γ^- , and the fixed costs relating to purchasing and selling i -asset as β^+ , β^- , respectively, a fixed-plus-linear function of transaction cost is formulated as follows:

$$\kappa_i = \begin{cases} 0, & \eta_i = \theta_i \\ \beta_i^+ + \gamma_i^+(\eta_i - \theta_i)x_i(1), & \eta_i > \theta_i \\ \beta_i^- + \gamma_i^-(\theta_i - \eta_i)x_i(1), & \eta_i < \theta_i \end{cases} \quad (8)$$

where $x_i(1)$ denotes the first price of the i -stock. Note that (8) is explicitly nonconvex with the exception of zero fixed costs case. In this way, we are extending our approach to deal with more complex transaction cost functions.

According to that, we set as the PSTC problem the minimization of the total transaction costs under the constraints of the portfolio. Thus, the PSTC problem is formulated as follows:

$$\min_{\eta} \quad \kappa \quad (9)$$

$$\text{subject to} \quad -X \cdot \eta \leq \min\{-G(\theta), -\phi \cdot \mathbf{1}\} \quad (10)$$

$$\eta_i \geq 0 \quad \forall i, \quad (11)$$

where κ is defined in (7).

Consequently, combining the problem of (4)-(6) with the problem (9)-(11), the MCPITC problem can be defined in the following NLP form:

$$\min_{\eta} \quad p^T \cdot \eta + \kappa \quad (12)$$

$$\text{subject to} \quad -X \cdot \eta \leq \min\{-G(\theta), -\phi \cdot \mathbf{1}\} \quad (13)$$

$$\mathbf{0} \leq \eta \leq X(1, :) \cdot \theta \cdot \left[\frac{1}{x_1(1)}, \dots, \frac{1}{x_n(1)} \right]^T, \quad (14)$$

The following algorithmic procedure can be easily implemented as a MATLAB function for calculating (12).

Algorithm 1 Algorithmic procedure for (12).

Input: The insurance prices p , the stock prices X , the portfolio η , the initial portfolio θ , the fixed costs β^- , β^+ and the costs rates γ^- , γ^+ .

1: function $f = \text{minfunc}(\eta, \theta, p, X, \beta^-, \beta^+, \gamma^-, \gamma^+)$

2: Set $f = p^T \eta + \text{sum}((\eta > \theta) \cdot (\beta^+ + \gamma^+(\eta - \theta) \cdot X(1, :))) + (\eta < \theta) \cdot (\beta^- + \gamma^-(\theta - \eta) \cdot X(1, :))$

3: return f

4: end function

Output: The function in (12).

2.2 BAS algorithm approach on the MCPITC problem

In [20], the penalty functions operate in a series of sequences, changing each time the penalty parameters, while starting a new sequence from the predecessors. During construction of any sequence the following penalty function is maximized:

$$V(z, R) = f(z) - \Omega(R, d(z), r(z)), \quad (15)$$

where Ω is the penalty term and $f(z)$ is the objective function. Moreover, R is a set of penalty parameters, $d(z)$ is the inequality constraint function and $r(y)$ is the equality constraint function. Note that, for equality or inequality constraints, different penalty terms are used. More details about the penalty function methods can be found in [20] and, in general, the key benefit of this approach is that it enables any convex or nonconvex constraints to be complied with.

As mentioned before, we approach the solution of the MCPITC problem through a modified BAS algorithm. Since BAS is directly applicable only to unconstrained optimization, it is necessary to use some additional methods that will keep solutions in the feasible region. Our modified version of BAS incorporates the aforementioned

penalty function. Therefore, we just use penalty function (15) and the initial value of the penalty parameter is settled by the user. Note that R remains constant over all the sequences generated by BAS. We can therefore build an even more effective modified BAS algorithm than the primary one by simply considering a penalty function as shown in (16). This approach helps BAS to manage convex or nonconvex constraints more effectively. Nevertheless, the concept of the BAS algorithm remains similar to the one set out in [21].

The following penalty function is used,

$$\Omega = R\langle d_j(z) \rangle^2, \tag{16}$$

where $d_j(z)$ is a penalty term handling the j -th inequality constraint and $\langle \cdot \rangle$ denotes the bracket-operator with $\langle w \rangle = w$, if w is negative, otherwise $\langle w \rangle = 0$. The bracket operator is an external penalty term, since the infeasible points are inserted with a negative value. This operator is mainly used to deal with the constraints on inequality. The corresponding algorithmic procedure, which handles the equality/inequality constraint of MCPITC, is shown in algorithm 2.

Algorithm 2 Penalty function algorithm for MCPITC.

Input: The requirements of Algorithm 1 plus the penalty parameter R , b the right part of Eq. (13), the lower limit η^- and upper limit η^+ of Eq. (14), respectively.

- 1: **function** $V = \text{penfunc}(\eta, \theta, p, X, b, R, \eta^-, \eta^+, \beta^-, \beta^+, \gamma^-, \gamma^+)$
- 2: Set $\Omega = R(\text{sum}((-X\eta > b).(-X\eta + b)^2 + (\eta^- > \eta).(\eta^- - \eta)^2 + (\eta > \eta^+).(\eta - \eta^+)^2))$
- 3: Set $V = \text{minfunc}(\eta, \theta, p, X, \beta^-, \beta^+, \gamma^-, \gamma^+) + \Omega$
- 4: **return** P
- 5: **end function**

Output: The outcome of Penalty Function.

According to [21], the way in which beetles track food using two antennas is determined by the strength of the odor they sense on the antennas. The BAS algorithm mimics these behaviors in order to discover an optimum solution to the problem. Beetle search behaviour can be designed to solve the problem of optimizing an objective function. This approach allows the implementation of new optimization algorithms (see [22-25]).

Using the MATLAB functions `minfunc`, which implements Algorithm 1, and `penfunc`, which implements Algorithm 2 we modified the BAS algorithm in the following Algorithm 3.

Algorithm 3 MCPITC problem through BAS algorithm.

Input: The requirements of Algorithms 1 and 2 plus the parameters d, δ, tol and $kmax$.

- 1: **function** $[\eta, f_\eta] = \text{basfunc}(\theta, p, X, b, R, \eta^-, \eta^+, \beta^-, \beta^+, \gamma^-, \gamma^+, d, \delta, tol, kmax)$
- 2: Set $y_1 = \theta$ and $y_2 = (\text{nan})\text{ones}(\text{size}(\theta))$
- 3: Set $k = 0$ and $len = \text{length}(\theta)$
- 4: **While** $k < kmax$ **and** $\|\text{penfunc}(y_2, \dots) - \text{penfunc}(y_1, \dots)\|_2 > tol$
- 5: Set $b = \text{rands}(k, 1)$ and $b = \frac{b}{2^{-5z} + \|b\|}$
- 6: Set $y_r = y_1 - db$ and $y_l = y_1 + db$
- 7: Set $y = |y_1 + \delta b(\text{sign}(\text{penfunc}(y_r, \dots) - \text{penfunc}(y_l, \dots)))|$
- 8: **if** $\text{penfunc}(y, \dots) < \text{penfunc}(y_1, \dots)$ **then**
- 9: Set $y_2 = y_1$ and $y_1 = y$
- 10: **end if**
- 11: Set $d = 0.988d + 0.001$ and $\delta = 0.988\delta$
- 12: **end while**
- 13: **return** $\eta = y_1, f_\eta = \text{minfunc}(y_1, \theta, p, X, \dots)$
- 14: **end function**

Output: η, f_η .

The dots in the inputs of minfunc and penfunc (see Algorithm 3), imply that all other inputs remain the same as the ones stated in Algorithms 1 and 2.

3. Numerical examples

In this section 3 numerical examples are presented. Furthermore, some additional information must be given for the following numerical examples as preliminaries.

First, the function κ_i has the same coefficients for all numerical examples of this section. Thus, κ_i is defined as follows:

$$\kappa_i = \begin{cases} 0, & \eta_i = \theta_i \\ 0.1 + 0.06(\eta_i - \theta_i)x_i(1), & \eta_i > \theta_i \\ 0.1 + 0.04(\theta_i - \eta_i)x_i(1), & \eta_i < \theta_i \end{cases}$$

where we have set $\beta_i^+ = \beta_i^- = 0.1$, $\gamma_i^+ = 0.06$ and $\gamma_i^- = 0.04$.

Second, we presume that our portfolio's insurance costs include a fixed premium plus a risk (i.e., rate of the variance of the assets). Assuming that ζ is the fixed price and α is the price rates relating to the risk of assets, then the function of the fixed-plus-linear insurance prices is formulated as follows:

$$p_i = \zeta + \alpha \cdot \text{Var}\left[\frac{x_i}{\max(x_i)}\right], \quad i = 1, 2, \dots, n, \tag{17}$$

where $\text{Var}[Y]$ denotes the variance of Y . Since $x_i \in \mathbb{R}^m$, we estimate the risk of the i -asset from the variance of its m normalized prices. In the following examples, the values of α and ζ are chosen in such a way that the insurance price is appropriate for the corresponding price of the stock. It should be noted that small changes in the value of α can cause large changes in the insurance prices.

Also, note that insurance premium is called the amount of money a company or person has to pay for an insurance plan. In fact, the insurance premium for each asset depends on the degree of risk it bears, where risk is the probability that the actual return may vary from the anticipated return. Consequently, the insurance prices become more realistic.

Third, in the numerical examples of this section, we have set $R = 1e5$, $d = 2$, $\delta = 2$, $tol = 1e - 6$ and $kmax = 1200$ in Algorithm 3. For BAS to produce more accurate results, we set the penalty value, the maximum iterations value and the tolerance value high. In addition, the parameters of the beetle's antennae d and δ are chosen in such a way that the optimal solution to the problem should be within their range. Note that d and δ are highly sensitive parameters and must always be within the range of the optimal solution.

Fourth, in the following examples, we use the default settings in GA. We use SFLA with maximum number of iterations 1000, memplex size 10, number of memplexes 5, number of offsprings 3 and step size 2. Furthermore, we use FA with population size 20, randomness strength 1, attractiveness constant 1, absorption coefficient 0.01, randomness reduction factor 0.97 and maximum number of iterations 500.

Lastly, the Tabs. 1, 2, 3 comprise of ticker symbols of stocks. A ticker symbol is an arrangement of characters that represents particular securities listed on an exchange or otherwise traded publicly. Every security listed has a unique ticker symbol, which facilitates the vast array of trade orders that flow every day across the financial markets.

3.1 Example A

Tab. 1 comprises of the stocks' ticker symbols which we use in the space of marketed securities X in this example.

Market			
AAL	AMD	F	GE

More precisely, let $X = [x_1, x_2, x_3, x_4]$, where X composed of the stocks daily close prices of Tab. 1 for the time period 1/7/2020 to 31/7/2020. In this time period, there exists 22 observations, so, $X \in \mathbb{R}^{22 \times 4}$. Given a portfolio $\theta = [2, 10, 2, 1]^T$, a floor $\phi = 580$ and $\alpha = 3000$, $\zeta = 5$ in (17), we present the findings in Fig. 1 where:

1) Fig. 1a displays the payoff of the portfolio η , which is $X \cdot \eta$, created by BAS, SFLA, FA and GA. Also, Fig. 1a displays the payoff of the portfolio θ as well as the floor price;

2) Fig. 1b displays the cost of the portfolio η , which is the outcome of (12). Note that (12) comprises of the insurance cost plus the transaction cost of the portfolio η . Also, Fig. 1b displays the insurance cost of the portfolio

θ , since its transaction cost is zero;

3) Fig. 1c displays the payoff-after-cost (PAC), which is $PAC = \text{payoff} - \text{cost}$, of the portfolios η and θ . The time consumptions of BAS, SFLA, FA and GA, in numerical example 3.1, are presented in Tab. 4.

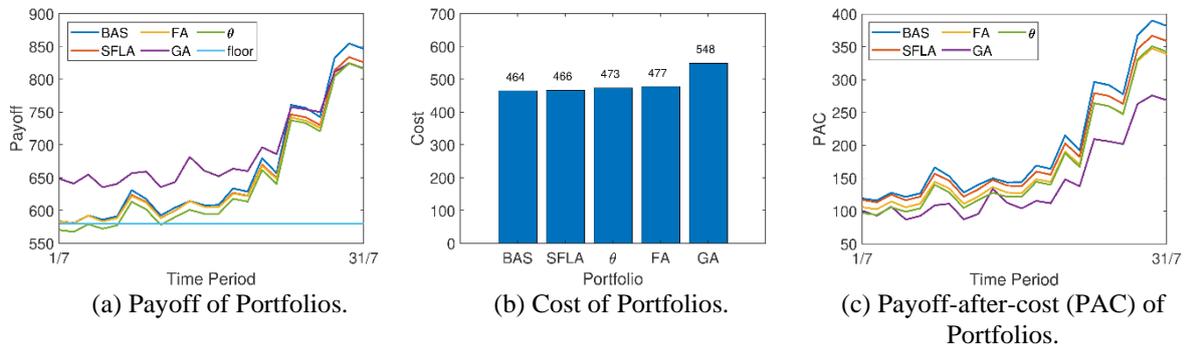


Figure 1. The payoff, the cost and the payoff-after-cost of four stocks (see example A).

3.2 Example B

Tab. 2 comprises of the stocks' ticker symbols which we use in the space X in this example.

More precisely, let $X = [x_1, x_2, x_3, x_4, x_5, x_6]$, where X composed of the stocks daily close prices of Tab. 2 for the time period 2/3/2020 to 30/4/2020. In this time period, there exists 43 observations, so, $X \in \mathbb{R}^{43 \times 6}$. Given a portfolio $\theta = [2, 8, 0, 1, 1, 3]^T$, a floor $\phi = 400$ and $\alpha = 2400$, $\zeta = 7$ in (17), we present the findings in Fig. 2 where:

1) Fig. 2a displays the payoff of the portfolio η , which is $X \cdot \eta$, created by BAS, SFLA, FA and GA. Also, Fig. 2a displays the payoff of the portfolio θ as well as the floor price;

2) Fig. 2b displays the cost of the portfolio η , which is the outcome of (12). Note that (12) comprises of the insurance cost plus the transaction cost of the portfolio η . Also, Fig. 2b displays the insurance cost of the portfolio θ , since its transaction cost is zero;

3) Fig. 2c displays the payoff-after-cost (PAC), which is $PAC = \text{payoff} - \text{cost}$, of the portfolios η and θ . The time consumptions of BAS, SFLA, FA and GA, in numerical example 3.2, are presented in Tab. 4.

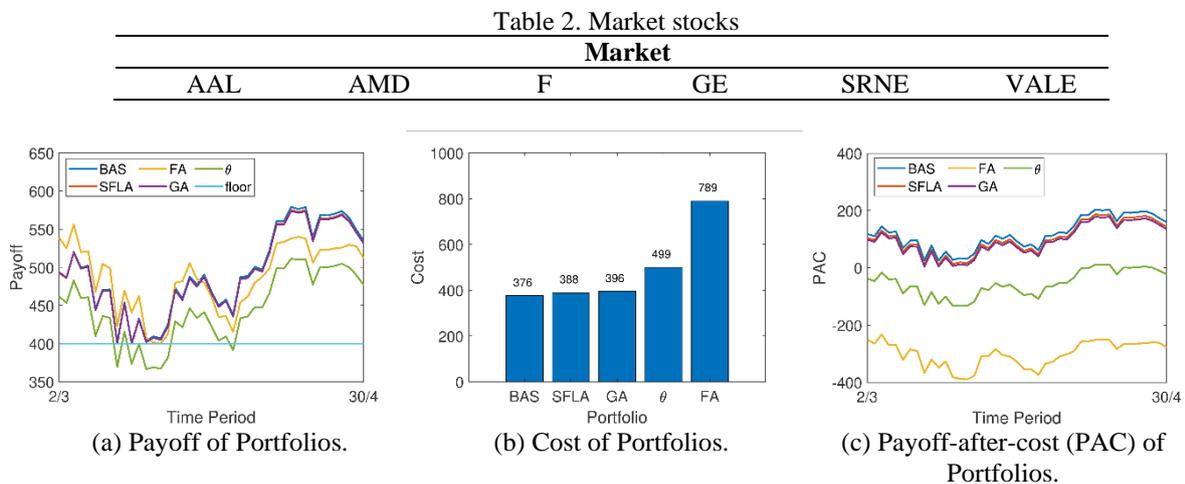


Figure 2. The payoff, the cost and the payoff-after-cost of six stocks (see example B).

3.3 Example C

Tab. 3 comprises of the stocks' ticker symbols which we use in the space X in this example.

Table 3. Market stocks

Market							
AAL	AMD	F	GE	NIO	PINS	SRNE	VALE

More precisely, let $X = [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8]$, where X composed of the stocks daily close prices of Tab. 3 for the time period 1/5/2020 to 31/7/2020. In this time period, there exists 64 observations, so, $X \in \mathbb{R}^{64 \times 8}$. Given

a portfolio $\theta = [2,7,1,1,2,4,1,0]^T$, a floor $\phi = 570$ and $\alpha = 800, \zeta = 5$ in (17), we present the findings in Fig. 3 where:

- 1) Fig. 3a displays the payoff of the portfolio η , which is $X \cdot \eta$, created by BAS, SFLA, FA and GA. Also, Fig. 3a displays the payoff of the portfolio θ as well as the floor price;
 - 2) Fig. 3b displays the cost of the portfolio η , which is the outcome of (12). Note that (12) comprises of the insurance cost plus the transaction cost of the portfolio η . Also, Fig. 3b displays the insurance cost of the portfolio θ , since its transaction cost is zero;
 - 3) Fig. 3c displays the payoff-after-cost (PAC), which is $PAC = \text{payoff} - \text{cost}$, of the portfolios η and θ .
- The time consumptions of BAS, SFLA, FA and GA, in numerical example 3.3, are presented in Tab. 4.

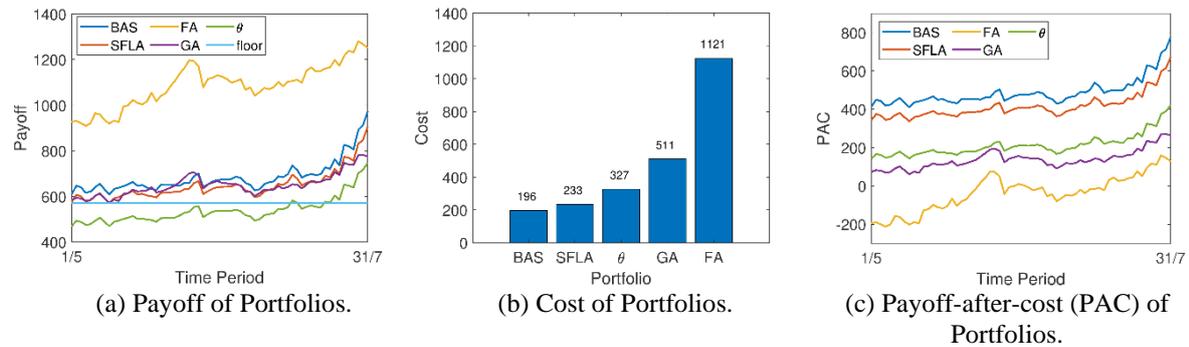


Figure 3. The payoff, the cost and the payoff-after-cost of eight stocks (see example C).

3.4 Superiority of the proposed method

We compared the performance of BAS, as presented in Algorithm 3, against the SFLA, FA and GA in MATLAB environment. The corresponding numerical results are shown in Tab. 4, which shows the average execution time of examples 3.1, 3.2 and 3.3. Note that, all numerical examples were executed using the MATLAB R2018b environment on a laptop HP-15-da0103nv, running on Windows 10 64 bit Operating System.

The results that are depicted in Figs. 1, 2, 3 show that BAS successfully solves the MCPITC problem and generates the solutions η of numerical examples 3.1, 3.2 and 3.3, respectively. In numerical example 3.1, GA is the worst-performing method and, in numerical examples 3.2, 3.3, FA is the worst-performing method. The payoffs of the portfolios η are shown in Figs. 1a, 2a, 3a and the costs of the portfolios η are displayed in Figs. 1b, 2b, 3b in numerical examples 3.1, 3.2, 3.3, respectively. Note that, the cost of the portfolio η is the outcome of (12), that is, the insurance cost plus the transaction cost of the portfolio η . Furthermore, in Figs. 1b, 2b, 3b is noticeable that the solution of BAS has the least cost on average contrasted to the solutions produced by SFLA, FA, GA. Also, Figs. 1c, 2c, 3c shows the payoff-after-cost (PAC) of the portfolios η and θ . Note that we see the clear payoff of the solutions through the PAC graph, which is $PAC = \text{payoff} - \text{cost}$, and thus we have a better insight. There, we observe that the solution of BAS gives, on average, the best outcome compared with the solutions of SFLA, FA, GA, in all numerical examples presented in this section. The time consumed in these examples is displayed in Tab. 4, and indicates that the BAS is much faster, on average, compared with SFLA, FA, GA, in all numerical examples presented in this section. In addition, the conclusion arising from Tab. 4 and Figs. 1, 2, 3 is that BAS performs more efficiently than the SFLA, FA and GA, in all numerical examples presented in this section and also worked excellently in solving the MCPITC problem.

Table 4. Examples 3.1, 3.2 and 3.3 execution time

Example	BAS	SFLA	FA	GA
3.1	0.04s	8.34s	1.1s	2.15s
3.2	0.04s	7.07s	1.03s	38.7s
3.3	0.03s	7.02s	0.94s	41.18s

4. Conclusion

The MCPITC problem is introduced in this paper. In addition, the BAS algorithm has provided a solution to the MCPITC problem, where three numerical examples have demonstrated its efficacy in such a financial NLP problem. In line with our numerical simulations, we concluded that the BAS method offers such a solution to the MCPITC problem that it makes it a highly competitive substitute, or even better than the SFLA, FA and GA methods. Experimental findings show the reliability of the proposed approach in various and realistic portfolio configurations on real-world datasets, and illustrate its usefulness in practical scenarios.

5. References

- [1] Soleimani H, Golmakani HR, Salimi MH. Markowitz-based portfolio selection with minimum transaction lots, cardinality constraints and regarding sector capitalization using genetic algorithm. *Expert Systems with Applications*. 2009;36(3):5058-5063.
- [2] Strang KD. Portfolio selection methodology for a nuclear project. *Project Management Journal*. 2011;42(2):81-93.
- [3] Eusuff M, Lansey K, Pasha F. Shuffled frog-leaping algorithm: a memetic meta-heuristic for discrete optimization. *Engineering optimization*. 2006;38(2):129-154.
- [4] Yang X-S. *Nature-inspired optimization algorithms*. Elsevier; 2014.
- [5] Khan AH, Cao X, Katsikis VN, Stanimirović P, Brajević I, Li S, Kadry S, Nam Y. Optimal portfolio management for engineering problems using nonconvex cardinality constraint: A computing perspective. *IEEE Access*. 2020;8:57437-57450.
- [6] Katsikis VN, Mourtas SD, Stanimirović PS, Li S, Cao X. Time-varying minimum-cost portfolio insurance under transaction costs problem via beetle antennae search algorithm (BAS). *Applied Mathematics and Computation*. 2020;385:125453.
- [7] Khan AT, Cao X, Li S, Hu B, Katsikis VN. Quantum beetle antennae search: a novel technique for the constrained portfolio optimization problem. *SCIENCE CHINA Information Sciences*. 2020.
- [8] Xie S, Chu X, Zheng M, Liu C. Ship predictive collision avoidance method based on an improved beetle antennae search algorithm. *Ocean Engineering*. 2019;192:106542.
- [9] Sun Y, Zhang J, Li G, Wang Y, Sun J, Jiang C. Optimized neural network using beetle antennae search for predicting the unconfined compressive strength of jet grouting coalcretes. *International Journal for Numerical and Analytical Methods in Geomechanics*. 2019;43(4):801-813.
- [10] Aliprantis CD, Brown DJ, Werner J. Minimum-cost portfolio insurance. *Journal of Economic Dynamics and Control*. 2000;24(11-12):1703-1719.
- [11] Katsikis VN. *Matlab - Modelling, programming and simulations*. ch. Computational and mathematical methods in portfolio insurance - A MATLAB-based approach. IntechOpen. 2010.
- [12] Katsikis VN, Mourtas SD. A heuristic process on the existence of positive bases with applications to minimum-cost portfolio insurance in C [a, b]. *Applied Mathematics and Computation*. 2019;349:221-244.
- [13] Katsikis VN. Computational methods in lattice-subspaces of C [a, b] with applications in portfolio insurance. *Applied Mathematics and Computation*. 2008;200(1):204-219.
- [14] Katsikis VN, Mourtas SD. ORPIT: A Matlab toolbox for option replication and portfolio insurance in incomplete markets. *Computational Economics*. 2019:1-1.
- [15] Katsikis VN. Computational methods in portfolio insurance. *Applied Mathematics and Computation*. 2007;189(1):9-22.
- [16] Yao Y. Optimization of dynamic portfolio insurance model. *Journal of Mathematical Finance*. 2012; 2: 181–188.
- [17] George J, Trainor W. Portfolio insurance using leveraged ETFs. Available at SSRN 3055199. 2017.
- [18] Pola G, Pola G. A stochastic reachability approach to portfolio construction in finance industry. *IEEE Transactions on Control Systems Technology*. 2012;20(1):189-195.
- [19] Lobo MS, Fazel M, Boyd S. Portfolio optimization with linear and fixed transaction costs. *Annals of Operations Research*. 2007;152(1):341-365.
- [20] Deb K. *Optimization for engineering design: Algorithms and examples*. second ed. PHI. 2013.
- [21] Jiang X, Li S. BAS: Beetle antennae search algorithm for optimization problems. arXiv preprint. 2017: vol. abs/1710.10724.
- [22] Khan AH, Cao X, Li S, Katsikis VN, Liao L. BAS-ADAM: An ADAM based approach to improve the performance of beetle antennae search optimizer. *IEEE/CAA Journal of Automatica Sinica*. 2020;7(2):461-471.
- [23] Zhu Z, Zhang Z, Man W, Tong X, Qiu J, Li F. A new beetle antennae search algorithm for multi-objective energy management in microgrid. In: 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA). IEEE. 2018. p.1599-1603.
- [24] Wu Q, Shen X, Jin Y, Chen Z, Li S, Khan AH, Chen D. Intelligent beetle antennae search for UAV sensing and avoidance of obstacles. *Sensors*. 2019;19(8):1758.
- [25] Xu X, Deng K, Shen B. A beetle antennae search algorithm based on Lévy flights and adaptive strategy. *Systems Science & Control Engineering*. 2020;8(1):35-47.



© 2020 by the author(s). This work is licensed under a [Creative Commons Attribution 4.0 International License](http://creativecommons.org/licenses/by/4.0/) (<http://creativecommons.org/licenses/by/4.0/>). Authors retain copyright of their work, with first publication rights granted to Tech Reviews Ltd.