FACTA UNIVERSITATIS (NIŠ)
Ser. Math. Inform. Vol. 31, No 2 (2016), 349–362

A NEW COMPUTATIONAL METHOD FOR FINDING THE CHEAPEST HEDGE

Vasilios N. Katsikis*

Abstract. In this article, we investigate the computational efficiency of an order theoretic approach applied to the problem of finding the cheapest hedge under portfolio constraints. In particular, we design a new computational method for computing the cheapest hedge and we discuss advantages of this method compared to the standard linear programming techniques. Numerical results as well as a new Matlab function for computing the cheapest hedge are provided.
Keywords: Portfolio dominance, Portfolio optimization

1. Introduction and notation

When markets are complete, there are as many states in the world as the non-redundant available securities. Therefore, if we form a matrix of non-redundant securities, then by taking the inverse value of the payoff matrix at the desired insured-payoff we can calculate the replicating portfolio. On the other hand, when markets are incomplete the desired insured payoff may not be marketed since there are more states in the world than available securities. In such a case we are interested in tradable portfolios which can pay at least as much at every state of the world as the desired payoff. That is, an investor is interested to purchase a portfolio that combines available securities and whose payoff dominates the desired insured payoff and has the lowest insurance premium.

The problem of finding the least costly portfolio, usually termed as the cheapest hedge or the minimum-premium insurance portfolio, whose payoff dominates the insured payoff is a finite minimization problem. This problem is extensively studied by many researchers and many different techniques have been developed for its solution (see, for example, [4, 5, 6, 7, 16, 18, 19]). Also, recent literature on optimization techniques provides a large number of important papers in this area under different perspectives, such as [20, 21].

Received February 08, 2016; accepted March 02, 2016
2010 *Mathematics Subject Classification.* 90C90, 91B99
*The author gratefully acknowledges that this work was carried out at the Department of Economics, University of Athens, Greece, and supported by Special Account Research Grant(SARG) 11698.

An alternative method for computing the cheapest hedge was presented in [1]. The innovation of this method focuses on the use of Riesz spaces and in particular on the existence of pseudo-complete markets in an incomplete market. Here we present a new computational tool (see the `mpiportfolio` function in the Appendix) that implements this method and we discuss its computational efficiency versus the standard linear programming procedure. To the best of our knowledge, a computational analogue of this very interesting approach to the cheapest hedge problem has not yet been presented or studied in the literature. The idea of finding the cheapest hedge is based on the notion of the portfolio dominance ordering i.e., a portfolio x dominates a portfolio y if $Ax \geq Ay$, where A denotes the payoff matrix. In incomplete markets portfolio dominance has no lattice structure (see [1, 2, 8, 9, 10, 11, 12, 13, 14, 15]), but it is possible to define pseudo-complete markets by considering proper invertible submatrices of the payoff matrix. Then, for each pseudo-complete market, its positive portfolio dominance cone is a lattice cone and we are in position to define a potentially insuring portfolio. The remarkable result of [1] states that one of the potentially insuring portfolios is a minimum-premium insurance portfolio.

Throughout the paper, we will understand $\mathbb{R}^m$ as the coordinate-wise ordered vector lattice $\bigoplus_{i=1}^m \mathbb{R}$. The *point-wise order* relation in $\mathbb{R}^m$ is defined by

$$x \leq y \text{ if and only if } x(i) \leq y(i), \text{ for each } i = 1, ..., m.$$

The positive cone of $\mathbb{R}^m$ is defined by $\mathbb{R}^m_+ = \{x \in \mathbb{R}^m | x(i) \geq 0, \text{ for each } i\}$ and if we suppose that X is a vector subspace of $\mathbb{R}^m$ then X ordered by the point-wise ordering is an *ordered subspace* of $\mathbb{R}^m$, with positive cone $X_+ = X \cap \mathbb{R}^m_+$. By $\{e_1, e_2, ..., e_m\}$ we shall denote the standard basis of $\mathbb{R}^m$. It is easy to see that the portfolio dominance order i.e.,

$$x \succcurlyeq y \text{ if } Ax \geq Ay$$

makes $\mathbb{R}^m$ a partially ordered vector space.

For $x, y \in \mathbb{R}^m$, $x \vee y$ (resp. $x \wedge y$) is the component-wise maximum (resp. minimum) of x and y defined by

$$(x \vee y)(i) = \max\{x(i), y(i)\} \text{ (resp. } (x \wedge y)(i) = \min\{x(i), y(i)\}), \text{ for all } i = 1, ..., m.$$

In our economy there are two time periods, $t = 0, 1$, where $t = 0$ denotes the present and $t = 1$ denotes the future. We consider that at $t = 1$ we have a finite number of states indexed by $s = 1, 2, ..., m$, while at $t = 0$ the state is known to be $s = 0$.

Let us denote by A, the payoff matrix i.e., the matrix whose columns are the non-redundant security vectors $x_1, x_2, ..., x_n$, that is $A = \left[x_i(j)\right]_{i=1,2,...,n}^{j=1,2,...,m} \in \mathbb{R}^{m \times n}$. Since the vectors x_i, $i = 1, \dots n$ are non-redundant, it is clear that the matrix A is of full rank. We shall denote the *asset span* by $X = Span(A)$, so X is the vector subspace of $\mathbb{R}^m$ generated by the vectors x_i. Economically speaking, that is, X consists of those income streams that can be generated by trading on the financial market. If $m = n$, then markets are said to be *complete* and the asset span coincides with the space $\mathbb{R}^m$. On the other hand, if $n < m$, the markets are *incomplete*, meaning that some state contingent claim cannot be replicated by a portfolio. As usual, a *portfolio* is a column vector $\theta = (\theta_1, \theta_2, ..., \theta_n)^T$ of $\mathbb{R}^n$ and the *payoff* of a portfolio θ is the vector $x = A\theta \in \mathbb{R}^m$, which offers payoff $x(i)$ in state i, where $i = 1, ..., m$. We shall denote by $\mathbf{1}$ the *risk-less (or risk-free) bond* i.e., the vector $\mathbf{1} = (1, 1, ..., 1)$. For notation not defined here the interested reader may refer to [1, 2, 8, 9, 10, 11, 12, 13, 14, 15, 17] and the references therein.

2. Computation of the cheapest hedge

The point-wise ordering in $\mathbb{R}^m$, induces the partial ordering $\geq_A$ in the portfolio space $\mathbb{R}^n$ and is defined as follows: for each $\theta, \phi \in \mathbb{R}^n$ we have

$$\theta \geq_A \phi, \text{ if and only if } A\theta \geq A\phi.$$

This ordering is known as the *portfolio dominance ordering*. An *insurance portfolio* is a portfolio u such that for a given portfolio θ and a floor k, then the payoff Au dominates, in each state, the quantity $\max\{A\theta, \mathbf{k}\}$ (where $\mathbf{k} = k \cdot \mathbf{1},\ k \in \mathbb{R}$). The space $\mathbb{R}^n$ endowed with the portfolio dominance ordering relation $\geq_A$ becomes a partial ordered vector space. Let us denote by C the pointed convex cone generated by the relation $\geq_A$, i.e.,

$$C = \{\theta \in \mathbb{R}^n : \theta \geq_A 0\},$$

then for any two portfolios θ, η we can write $\theta \vee_C \eta$ to mean the supremum (i.e., the least upper bound) of the two-point set $\{\theta, \eta\}$ with respect to the ordering $\geq_A$. Note that this supremum does not necessarily exists.

Any vector $q = (q_1, q_2, \ldots, q_n) \in \mathbb{R}^n$, where q_i is the price of security i, is called a *security price*. In our study we assume that the risk-less bond, $\mathbf{1} = (1, 1, ..., 1)$, is marketed and that any security price vector q is positive in the portfolio dominance ordering, i.e., $q \cdot \theta > 0$ whenever the portfolio θ satisfies $A\theta > 0$. This type of prices is known as *arbitrage-free prices*. The minimum-premium insurance portfolio minimization problem or else the problem of finding the cheapest hedge is

$$\begin{aligned} &\min q \cdot \eta \\ &s.t.:\ \eta \in \mathbb{R}^n,\ A\eta \geq A\theta \text{ and } A\eta \geq \mathbf{k} \end{aligned} \tag{2.1}$$

Under the above considerations, the solution set of this problem is non-empty and also it is a compact and convex subset of $\mathbb{R}^n$ (see [1]). Also, it is well known that there are several techniques to solve such problems, ranging from classical linear programming methods to more sophisticated methods as in [22]. Our approach here provides an alternative method, based on portfolio dominance ordering and the existence of pseudo-complete markets.

If A is the $m \times n$ payoff matrix, i.e., the matrix whose columns are the non-redundant security vectors $x_1, x_2, ..., x_n$ then a pseudo-complete market is defined as follows.

Definition 2.1. Let $I = \{i_1, i_2, \ldots, i_n\} \subset \{1, 2, \ldots, m\}$, with $i_1 < i_2 < \ldots < i_n$, we say that the matrix

$$A_I = \begin{bmatrix} x_1(i_1) & x_2(i_1) & \ldots & x_n(i_1) \\ x_1(i_2) & x_2(i_2) & \ldots & x_n(i_2) \\ \vdots & \vdots & & \vdots \\ x_1(i_n) & x_2(i_n) & \ldots & x_n(i_n) \end{bmatrix}$$

defines a *pseudo-complete market* if A_I is invertible.

The existence of pseudo-complete markets is obvious since $rank(A) = n$. For every pseudo-complete market defined by the matrix A_I we may also define the corresponding portfolio ordering, denoted by $\geq_I$ such that

$$\theta \geq_{A_I} \phi, \text{ if and only if } A_I\theta \geq A_I\phi.$$

This ordering is a lattice ordering i.e., the cone $C_I = \{\theta \in \mathbb{R}^m : \theta \geq_{A_I} 0\}$ is a lattice cone and then if η, θ are two portfolios the supremum $\eta \vee_{A_I} \theta$ exists and is given by the relation

$$\eta \vee_{A_I} \theta = A_I^{-1} \max\{A_I\eta, A_I\theta\}.$$

Let θ be any portfolio and k is a floor price, then a *potentially insuring portfolio* is a portfolio η_{A_I} such that

$$A\eta_{A_I} \geq \max\{A\theta, \mathbf{k}\}.$$

The set of all potentially insuring portfolios shall be denoted by $\mathcal{P}_{\theta,k}$.

Theorem 2.1. Aliprantis et al. [1] *For any portfolio θ, any arbitrage-free price q, and any floor k we have the following:*

1. *There exists at least one potentially insuring portfolio $\theta \vee_{A_I} \boldsymbol{k}$ which is a minimum-premium insurance portfolio for θ at floor k.*

2. *A minimum-premium insurance portfolio $\theta \vee_{A_I} \boldsymbol{k}$ is the i.e. potentially insuring portfolio. That is, $q(\theta \vee_{A_I} \boldsymbol{k}) \leq q\eta$ for all $\eta \in P_{\theta,k}$.*

3. *The portfolio $\eta^* = \theta \vee_C \boldsymbol{k}$ exists if and only if $P_{\theta,\boldsymbol{k}}$ consists of only one portfolio η^*, which is automatically a minimum-premium insurance portfolio for any arbitrage-free price.*

3. The computational approach

In this section we shall present a computational method to find the minimum-premium insurance portfolio θ for any arbitrage-free price q and any floor k. Existence of such a portfolio is evident from Theorem 2.1. Also, we shall present the basic advantages against the standard linear programming procedure. It is important to notice that the proposed algorithmic procedure may result to multiple minimum-premium insurance portfolios corresponding to a single arbitrage-free price. So, besides other advantages that we will analyze in the sequel, one of the the main advantages of our method relies on the fact that, in such cases, there is a clear perspective on choosing between different solutions of the minimization problem according to prior knowledge or experience.

The following proposition is of great importance since it characterizes all the solutions of the minimization problem (2.1). The basic part of its proof is an easy consequence of the proof of Theorem 2.1, so it is omitted.

Proposition 3.1. *By θ, q and k we shall denote any portfolio, arbitrage-free price and floor, respectively. Also, we shall assume that the risk-less bond is marketed. Then,*

1. *The minimization problem* (2.1) *has at least one solution.*
2. *Convex combination of different solutions of problem* (2.1) *is also a solution.*
3. *If a portfolio η^* is a solution of* (2.1)*, then $\eta^* \in \mathcal{P}_{\theta,k}$.*

In view of Proposition 3.1 and the preceding discussion of section 2., we are in position to present the following algorithm.

Require: The matrix A, i.e., the payoff matrix with the non-redundant security vectors $x_1, x_2, ..., x_n$ specified as columns, the portfolio vector θ, the floor k and the price vector q.
1: Compute the insured payoff.
2: Compute the pseudo-complete markets A_I.
3: Compute the set $\mathcal{P}_{\theta,k} = \{\eta_I \in \mathbb{R}^m : \eta_I = A_I^{-1} \max\{A_I\theta, \mathbf{k}\}\}$, of all potentially insuring portfolios.
4: If $\mathcal{P}_{\theta,k}$ consists of only one portfolio, say η, stop the procedure. Then η is the minimum-premium insurance portfolio. Or else, continue to the next step.
5: Find the least costly portfolios with respect to the price vector q.
6: Compute the output, that is the minimum-premium insurance portfolio, from the previous step, for any arbitrage-free price.

Algorithm 3. corresponds to the Matlab function `mpiportfolio` presented in the Appendix. The `mpiportfolio` function is our basic tool in order to find the potentially insuring portfolios and then the minimum-premium insurance portfolio.

4. Numerical experiments

In this section, we highlight the kind of analysis that can be efficiently performed with the presented approach. Also, we test the proposed algorithmic procedure (Algorithm 3.) against the standard Matlab function `linprog`, for linear programming problems. In what follows, it is important to keep in mind that the validity of Theorem 2.1 requires that we are working with arbitrage-free prices. In addition, note that the arbitrage-free prices are exactly the C-strictly positive vectors, that is, the arbitrage-free prices are elements of the cone generated by the rows of the payoff matrix A.

4.1. Basic examples

We are now ready to present and discuss two examples of particular interest for our analysis. The low dimensionality of these examples have been chosen for better representation purposes. The first one is an example of an incomplete market with only one potentially insuring portfolio.

Example 4.1. *Suppose that there are ten states of the world and our market is described by the following non-redundant securities:*

- *A corporate bond with payoff* $x_1 = (2, 2, 4, 3, 0, 0, 0, 0, 1, 1)$.
- *Two shares with payoffs* $x_2 = (0, 0, 1, 1, 2, 3, 1, 3, 4, 4)$ *and* $x_3 = (3, 3, 0, 0, 0, 0, 4, 0, 0, 0)$.
- *Two treasury bonds with payoffs* $x_4 = (1, 1, 0, 1, 0, 1, 0, 1, 0, 0)$ *and* $x_5 = (0, 0, 1, 0, 1, 0, 1, 0, 1, 1)$.
- *A municipal bond with payoff* $x_6 = (0, 0, 0, 0, 0, 0, 6, 0, 0, 0)$.
- *A call option written on the share* x_2 *with strike price of* 3. *That is, the security* $x_7 = (x_2 - \mathbf{3})^+ = \max\{(x_2 - \mathbf{3}), \mathbf{0}\} = (0, 0, 0, 0, 0, 0, 0, 0, 1, 1)$.

Thus, the market is described by the returns matrix

$$A = \begin{bmatrix} 2 & 0 & 3 & 1 & 0 & 0 & 0 \\ 2 & 0 & 3 & 1 & 0 & 0 & 0 \\ 4 & 1 & 0 & 0 & 1 & 0 & 0 \\ 3 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 & 0 & 0 \\ 0 & 3 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 4 & 0 & 1 & 6 & 0 \\ 0 & 3 & 0 & 1 & 0 & 0 & 0 \\ 1 & 4 & 0 & 0 & 1 & 0 & 1 \\ 1 & 4 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

Note that the risk-less bond **1** *is marketed. Consider the portfolio*

$$\theta = (0, 3, 0, 0, 0, 0, 0)$$

of three shares of security 2 *at floor* $k = 10$. *By using the* `SUBlatSUB` *function presented in [9] one can easily see that the asset span* $X = [x_1, x_2, x_3, x_4, x_5, x_5, x_6, x_7]$ *forms a vector sublattice of* $\mathbb{R}^{10}$. *At this point we may follow two different ways in order to calculate the "best" insurance portfolio. One way is to use Theorem 3.2 from [3] alongside with the computational methods presented in [8, 9, 12]. In particular, the Matlab function* `mcpinsurance`, *from [12], generates the minimum-cost insured portfolio by using the code*

```
>> eta=mcpinsurance(A,floorvector,theta)
```

where $floorvector = floor * \mathbf{1}$.

Then, the solution is the portfolio

$$\eta = (0, 0, 0, 10, 10, 0, 2).$$

On the other hand if we use the proposed method i.e., the `mpiportfolio` *Matlab function from the Appendix we have,*

```
>> eta=mpiportfolio(A,floor,theta,price)
```

and the results are exactly the same as with the `mcpinsurance` *function. Note that in the case where the market has a lattice structure (i.e., it is a vector sublattice or a lattice-subspace) then the solution is price-independent. So, in the present example the choice of the arbitrage-free price vector, for the correct performance of* `mpiportfolio,` *is free. On the other hand, notice that by using the Matlab function* `linprog` *and for different arbitrage free prices the corresponding minimization problem must be solved repeatedly each time to find the minimum-premium insurance portfolios. So, the knowledge that our market has a vector lattice structure in conjunction with the proposed method can reduce significantly the computation time. A graphical illustration of this example is provided in Figure 4.1.*

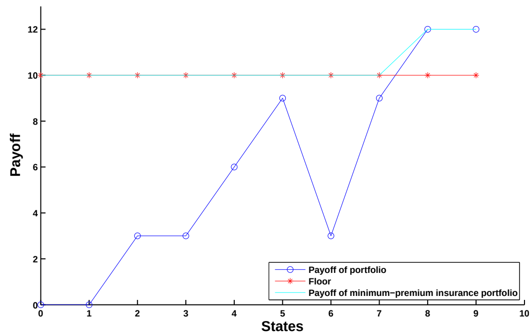

FIG. 4.1: The unique minimum-premium insurance portfolio contains 10 treasury bonds x_4 and 10 treasury bonds x_5 as well as 2 call options written on the share x_2 with strike price of 3

We continue with an example of an incomplete market with price-dependent insurance. It is easy to see, by using the `SUBlatSUB` function from [9], that this market is not a lattice-subspace, so our solution is price dependent. Furthermore, we shall see that for certain choices of price vectors we may have multiple solutions, i.e., more than one minimum-premium insurance portfolios for the same price. The interesting thing, regarding our approach, is that one can easily compute several minimum-premium insurance portfolios. This is an advantage against the classical optimization techniques since the interested user can choose between different minimum-premium insurance portfolios according to his/her prior knowledge or experience.

Example 4.2. *Suppose that there are seven states of the world and our market is described by the following non-redundant securities:*

- *A corporate bond with payoff* $x_1 = (0.25, 0.25, 0.25, 0.25, 0, 0, 0)$.
- *Two shares with payoffs* $x_2 = (0, 0, 0, 0, 1, 1, 1)$ *and* $x_3 = (2, 1, 0, 0, 0, 0, 0)$.
- *A treasury bond with payoff* $x_4 = (1, 5, 3, 0, 0, 0, 0)$.

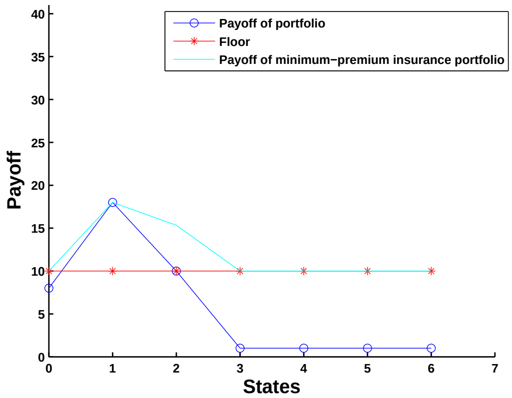

FIG. 4.2: Payoff of minimum-premium insurance portfolio $\eta_1 = (40, 10, -\frac{8}{9}, \frac{16}{9})$, that is $A \cdot \eta_1 = (10, 18, \frac{46}{3}, 10, 10, 10, 10)$

Thus, the market is described by the returns matrix

$$A = \begin{bmatrix} 1/4 & 0 & 2 & 1 \\ 1/4 & 0 & 1 & 5 \\ 1/4 & 0 & 0 & 3 \\ 1/4 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

Consider the portfolio $\theta = (4, 1, 2, 3)$ at floor $k = 10$. Again, by using the code presented in [9]*, we get that the asset span $X = [x_1, x_2, x_3, x_4]$ does not have a vector lattice structure. So, Theorem 3.2 from* [3] *is not valid anymore and we have price-dependent insurance. The proposed method,* `mpiportfolio` *function, by using the code*

```
>> eta=mpiportfolio(A,floor,theta,price)
```

for the arbitrage-free price vector $p = (44, 11, 5, 25)$, provides two alternatives. In particular, we have the solutions $\eta_1 = (40, 10, -\frac{8}{9}, \frac{16}{9})$ and $\eta_2 = (40, 10, 8, 0)$ and it holds $p \cdot \eta_1 = p \cdot \eta_2 = 1910$. This example confirms the already mentioned advantage of this method. That is, we may compute more than one available minimum-premium insurance portfolios. In particular we may compute s different minimum-premium insurance portfolios with $1 \leq s \leq \begin{pmatrix} 7 \\ 4 \end{pmatrix} = 35$, while in the contrary the classical optimization techniques provide only one for the given price vector p.

In Figure 4.2 and Figure 4.3 we can see the two choices for portfolio insurance for the same price $p = (44, 11, 5, 25)$. Also, we tested the `linprog` Matlab function which provides one minimum-premium insurance portfolio.

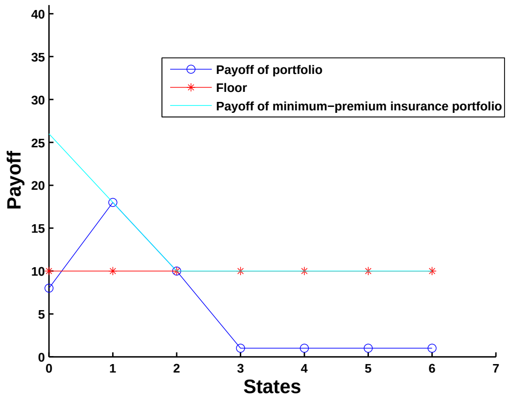

FIG. 4.3: Payoff of minimum-premium insurance portfolio $\eta_2 = (40, 10, 8, 0)$, that is $A \cdot \eta_2 = (26, 18, 10, 10, 10, 10, 10)$

4.2. Numerical experiments with large data sets

In this subsection we analyze numerical data arising during the computation of minimum-premium insurance portfolio by applying a Matlab implementation of Algorithm 3. (see the appendix). In order to test the time efficiency as well as the accuracy of our method we compare our results with those obtained using the Matlab function `linprog`. Our series of test examples exploit payoff matrices, A, randomly generated under the restrictions that a) the risk-less bond is marketed and b) the matrices A are of full rank. Let us denote by $minval_1$ the minimum value computed by Algorithm 3. and by $minval_2$ the minimal value computed by the `linprog` Matlab function. Then the number

$$MINVE = minval_1 - minval_2,$$

is a measure for comparison of the minimal values achieved by applying both methods. The more negative the number MINVE is, the larger the overestimation of the minimal value by the `linprog` is. Also, we shall denote by N_p the number of different minimum-premium insured portfolios that can be found by using the two methods. Note that the `linprog` can give us only $N_p = 1$. All the numerical tasks have been performed by using the Matlab R2015a environment on an Intel(R) Core(TM) i7-3770 CPU @ 3.40 GHz 64-bit system with 16 GB of RAM running on Windows 7 Professional SP1 Operating System. So, we shall present results for large data sets, i.e for incomplete markets generated by a large number of securities and states. The cumulative results are included in Table 4.1. As we can see in Table 4.1, the negative values of MINVE, in each tested case, suggests that the `linprog` function overestimates the minimal value of the minimization problem. Also, according to the numerical results we observe that the proposed method, Algorithm 3., performs quite well both in terms of accuracy as well as in CPU time response. Finally, in most of the cases we found more than one minimum-premium insurance

Table 4.1: Results for large data sets.

Method	Size m, n	CPU Time	MINVE	N_p
Algorithm 3.	50, 49	0.0228		4
Algorithm 3. and linprog			-04.0009e-06	
linprog	50, 49	0.0819		1
Algorithm 3.	100, 99	0.2026		1
Algorithm 3. and linprog			-08.0007e-07	
linprog	100, 99	0.8608		1
Algorithm 3.	150, 149	0.4651		1
Algorithm 3. and linprog			-4.0001e-06	
linprog	150, 149	0.4331		1
Algorithm 3.	200, 199	1.1638		4
Algorithm 3. and linprog			-0.0369	
linprog	200, 199	0.5242		1
Algorithm 3.	250, 249	2.3667		2
Algorithm 3. and linprog			-0.2092	
linprog	200, 199	0.5091		1
Algorithm 3.	300, 299	3.9350		7
Algorithm 3. and linprog			-0.0843	
linprog	300, 299	0.6234		1
Algorithm 3.	350, 349	6.2079		1
Algorithm 3. and linprog			-0.2080	
linprog	350, 349	0.7464		1

portfolios.

4.3. Advantages and comments

According to the analysis of the previous sections we conclude to the following advantages and comments regarding the proposed method:

i) If $P_{\theta,k}$ consists of only one portfolio then automatically this portfolio is the minimum-premium insurance portfolio and no further investigation is needed.

ii) An important advantage of this method is that we are able to calculate, for the same arbitrage-free price, different minimum-premium insurance portfolios (see Example 4.2). This fact, gives us the opportunity to choose between optimum insurance portfolios that best fits our needs and experience. Note that, the linear programming method provides us only one optimum insurance portfolio.

iii) Our method is based upon the existence of pseudo-complete markets in an incomplete market which is always guaranteed by Theorem 2.1, case (1).

iv) If the lattice property holds then the solution we get, from the proposed method, is price-independent and this is an important advantage over the traditional linear programming methods (see Example 4.1). Also, recall that, when the market has a lattice structure[1] then the solution is price-independent .

v) According to Proposition 3.1, convex combination of different solutions of problem (2.1) is also a solution.

Finally, the computation of the set $\mathcal{P}_{\theta,k}$, is not always an easy task. In fact, for an $m \times n$ payoff matrix A, with $n < m$ the set $\mathcal{P}_{\theta,k}$ has $\begin{pmatrix} m \\ n \end{pmatrix}$ elements. This issue, in some cases, may cause a difficulty in calculations but, on the other hand, according to Proposition 3.1 we are able to find all the minimum-premium insured portfolios.

5. Conclusion

In this work we discuss the minimization problem of finding the minimum-premium insurance portfolio as a minimization problem over a finite set, $\mathcal{P}_{\theta,k}$. Moreover, we propose Algorithm 3. as the basic tool for solving the minimization problem (2.1). According to the proposed method, we are able to calculate, for the same arbitrage-free price, different minimum-premium insurance portfolios and then choose properly the minimum-premium insurance portfolio according to our knowledge or experience.

6. Appendix

The Matlab implementation of Algorithm 3. is given below.

```
function[insuredpayoff, mpiportfol, minvalue] = mpiportfolio(A,theta,floor,price)

%***************************%
%   General  Information.   %
%***************************%
% Synopsis:
% mpiportfolio = mpiportfolio(A,theta,floor,price)
% Input:
%   A =  an mxn payoff matrix, i.e., the matrix whose columns
%          are the non-redundant security vectors x1 ,x2 ,...,xn.
%   theta = a given portfolio (dimension = nx1).
%   floor = the real number that acts as a floor.
%   price = an arbitrage-free price vector, i.e., an element of the cone
%            generated by the rows of the payoff matrix A (dimension = nx1).
```

[1] The basic computational tools for testing the vector lattice property of a market (i.e., the market is a vector sublattice or lattice-subspace) are provided in [9].

```
%
% Output:
%   mpiportfolio = minimum-premium insurance portfolio. Also, we may ask
%   for the insured payoff and minimum value of the problem.
%
%*****************************************************%
%  Computation of minimum-premium insurance portfolio by  %
%  using Algorithm 1                                      %
%*****************************************************%
[m,n]=size(A);
%**********************************%
%    Compute the insured payoff.    %
%**********************************%
insuredpayoff= max([A*theta floor*ones(m,1)],[],2);
%*************************************%
%  Compute the pseudo-complete markets. %
%*************************************%
combos = nchoosek(1:m,n);
t = length(combos(:,1));
%*********************************************%
%  Compute the potentially insuring portfolios. %
%*********************************************%
min_prem_ins_port_matrix = zeros(n,t);
for i = 1:t
    Ai = A(combos(i,:),:);
    ranki = rank(Ai);
    if ranki == n
       portfolio = Ai\max([Ai*theta floor*ones(ranki,1)],[],2);
      min(A*portfolio-insuredpayoff)
      minimum = 1e-7+min(A*portfolio-insuredpayoff);
        if minimum >=0
          min_prem_ins_port_matrix(:,i) = portfolio;
       end
    end
end
%**********************************%
%  Find the least costly portfolio. %
%**********************************%
[~,j] = find(min_prem_ins_port_matrix);
 indices = unique(j)';
 cost = price'*min_prem_ins_port_matrix(:,indices);
 minvalue = min(cost);
 ind1 = eq(cost,minvalue);
 ind2 = ind1.*indices;
 [~,jj] = find(ind2);
```

```
ind3 = ind2(jj);
mpiportfol = min_prem_ins_port_matrix(:,ind3);
minvalue = price'*mpiportfol;
```

REFERENCES

1. C. D. Aliprantis, Y. Polyrakis, R. Tourky, *The cheapest hedge*, Journal of Mathematical Economics **37**, 2002, 269–295.
2. C.D. Aliprantis, R. Tourky, Markets that don't replicate any option, *Economics Letter,* **76**, 2002, 443–447.
3. C.D. Aliprantis, D.J. Brown, Minimum-cost portfolio insurance, *J. Economic Dynamics and Control,* **24**, 2002, 1703–1719.
4. M. Broadie, J. Cvitanic, H.M. Soner, Optimal replication of contingent claims under portfolio constraints, *Review of Financial Studies*, **11**, 1998, 59–81.
5. D.J. Brown, S.A. Ross, Spanning valuation and options, *Economic Theory* **1**, 1991, 3–12.
6. C. Edirisinghe, V. Naik, R. Uppal, Optimal replication of options with transaction costs and trading restrictions, *Journal of Financial and Quantitative Analysis* **28**, 1993, 117–139.
7. R. Green, R.A. Jarrow, Spanning and completeness in markets with contingent claims, *Journal Economic Theory* **41**, 1987, 202–210.
8. V.N. Katsikis, I. Polyrakis, Computation of vector sublattices and minimal lattice-subspaces of $\mathbb{R}^k$. Applications in finance. *Applied Mathematics and Computation* **218**, 2012, 6860–6873.
9. V.N. Katsikis, A Matlab-based rapid method for computing lattice-subspaces and vector sublattices of $\mathbb{R}^n$: Applications in portfolio insurance, *Applied Mathematics and Computation,* **215**, 2009, 961–972.
10. V.N. Katsikis, Computational methods in lattice-subspaces of $C[a, b]$ with applications in portfolio insurance, *Applied Mathematics and Computation,* **200**, 2008, 204–219.
11. V.N. Katsikis, Computational methods in portfolio insurance, *Applied Mathematics and Computation,* **189**, 2007, 9–22.
12. V.N. Katsikis. Computational and Mathematical Methods in Portfolio Insurance. A MATLAB-Based Approach., Matlab - Modelling, Programming and Simulations, (Ed.), ISBN: 978-953-307-125-1, InTech, 2010 (Book chapter).
13. V.N. Katsikis, Computational methods for option replication, *International Journal of Computer Mathematics,* **88, No. 13**, 2011, 2752–2769.
14. V.N. Katsikis, A new characterization of markets that don't replicate any option through minimal-lattice subspaces. A computational approach. *Filomat* 27:7, 2013, 1357–1372.
15. V.N. Katsikis. MATLAB Aided Option Replication, MATLAB - A Fundamental Tool for Scientific Computing and Engineering Applications - Volume 3, ISBN 978-953-51-0752-1, InTech, 2012 (Book chapter).
16. H. Leland, Who should buy portfolio insurance, *Journal of Finance* **35**, 1980, 581–594.

17. M. Magill, M. Quiinzi, *The theory of incomplete markets*, Vol. 1, MIT Press, Cambridge, 1996.
18. V. Naik, R. Uppal, Leverage constraints and the optimal hedging of stock and bond options, *Journal of Financial and Quantitative Analysis* **29**, 1994, 199–223.
19. S.A. Ross, Options and efficiency, *Quarterly Journal of Economics* **9**, 1976, 75–89.
20. Ivan P. Stanimirović, Milan Lj. Zlatanović, Marko D. Petković, On the linear weighted sum method for multi-objective optimization, *Facta Universitatis, Ser. Math. Inform.*, **26**, 2011, 49–63.
21. P.S. Stanimirović and I.P. Stanimirović, Implementation of polynomial multi-objective optimization in MATHEMATICA, *Struct. Multidiscip. Optim.* **36**, 2008, 411–428.
22. N.V. Stojković and P.S. Stanimirović, Two direct methods in linear programming, *European Journal of Operational Research* **131**, 2001, 417–439.

Vasilios N. Katsikis
National and Kapodistrian
University of Athens,
Department of Economics,
Division of Mathematics
and Informatics, Athens, Greece
Sofokleous 1 Street,
10559 Athens, Greece
vaskatsikis@econ.uoa.gr
http://users.uoa.gr/∼vaskatsikis