Computers & Operations Research 139 (2022) 105631

Contents lists available at ScienceDirect

Computers and Operations Research

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

A simheuristic algorithm for the portfolio optimization problem with random returns and noisy covariances

Renatas Kizys [a,*], Jana Doering [b], Angel A. Juan [b,c], Onur Polat [d], Laura Calvet [b,e], Javier Panadero [b,c]

[a] Department of Banking and Finance, Southampton Business School, University of Southampton, Southampton SO17 1BJ, United Kingdom
[b] IN3 – Computer Science Dept., Open University of Catalonia, 08860 Castelldefels, Spain
[c] Euncet Business School, 08225 Terrassa, Spain
[d] Department of Public Finance, Bilecik University, 11230 Bilecik, Turkey
[e] Valencian International University, Pintor Sorolla 21, 46002 Valencia, Spain

ARTICLE INFO

Keywords:
Constrained portfolio optimization
Metaheuristics
Simulation
Financial assets
Variable neighborhood search
Biased randomization

ABSTRACT

The goal of the portfolio optimization problem is to minimize risk for an expected portfolio return by allocating weights to included assets. As the pool of investable assets grows, and additional constraints are imposed, the problem becomes *NP-hard*. Thus, metaheuristics are commonly employed for solving large instances of rich versions. However, metaheuristics do not fully account for random returns and noisy covariances, which renders them unrealistic in the presence of heightened uncertainty in financial markets. This paper aims to close this gap by proposing a simulation–optimization approach – specifically, a simheuristic algorithm that integrates a variable neighborhood search metaheuristic with Monte Carlo simulation – to deal with stochastic returns and noisy covariances modeled as random variables. Computational experiments performed on a well-established benchmark instance illustrate the advantages of our methodology and analyze how the solutions change in response to a varying degree of randomness, minimum required return, and probability of obtaining a return exceeding an investor-defined threshold.

1. Introduction

Financial investments play an essential role in our society through resource allocation, wealth creation, sustainable economic growth and ultimately improvements in welfare standards. They provide companies with the necessary funds to transform ideas and resources into profitable projects, social benefits and jobs. Most of the related questions in financial economics can be formulated as combinatorial optimization problems (COPs).

Optimization methods can be grouped into exact methods and heuristics/metaheuristics (Talbi, 2009). The first group entails procedures that guarantee the optimality of a solution. However, exact methods may require strict assumptions or simplifying formulations to render the problem solvable, thus reducing the informational value of real-life operational applications. Furthermore, when exact methods are used to solve large instances of *NP-hard* optimization problems, they require enormous computational times. Within the second group, heuristics are experience-based procedures, which typically provide reasonably good solutions within very short computational times (a few seconds or even less). By contrast, metaheuristics (Boussaïd et al., 2013; Michalewicz and Fogel, 2013; Doering et al., 2019) are general solving procedures, which are able to provide near-optimal solutions for a broad range of optimization problems within reasonable computational times (typically in the order of a few minutes).

At the core of modern portfolio theory, pioneered by Markowitz (1952), the portfolio optimization problem (POP) defines the investment decision as the strategy of: *(i)* selecting financial assets; and *(ii)* determining the optimal weights allocated to those assets, which results in a desired (also referred to as expected or required) portfolio return and a minimum level of risk. The POP entails a quadratic objective function that aggregates the weighted covariances of the constituent asset returns, and a required rate of return. It is worth noting that other risk measures, such as value-at-risk or Sharpe ratio can be employed too. The basic version of the POP implies that portfolio weights must add up to one and take on non-negative values, which rules out short-selling. This basic version of the problem can be solved through exact methods and, in fact, these methods have been predominantly used in the POP literature (Bonami and Lejeune, 2009; Mansini et al., 2014). However, metaheuristics are increasingly employed to deal with more

* Corresponding author.
E-mail address: r.kizys@soton.ac.uk (R. Kizys).

https://doi.org/10.1016/j.cor.2021.105631
Received 7 June 2020; Received in revised form 5 November 2021; Accepted 8 November 2021
Available online 20 November 2021

realistic and complex versions of the problem (Soler-Dominguez et al., 2017; Ray et al., 2019; Kalayci et al., 2019), in which additional constraints are considered. In particular, pre-assignment, quantity, and cardinality constraints have received overwhelming attention in the related literature (Chang et al., 2000; Soleimani et al., 2009; Lwin et al., 2017; Meghwani and Thakur, 2017). Concretely, pre-assignment constraints pre-select certain assets, irrespective of their risk-return characteristics. Quantity constraints confine the weight allocated to an asset in the portfolio between a floor and a ceiling. These constraints simultaneously limit the exposure to specific assets and rule out investments in negligible quantities – a practice that may be prohibitively costly, since the transaction costs may reduce or erase the benefit. While recognizing that quantity constraints arise as a result of the investor's discretionary decisions, these constraints have received a growing volume of interest. Kolm et al. (2014) and Babaei et al. (2015) are recent examples. For instance, Kolm et al. (2014) argue that the inclusion of quantity constraints: *(i)* can lead to improved performance, *(ii)* can help to contain portfolio volatility, and *(iii)* can decrease downside risk and shortfall probability. Behr et al. (2013) further assert that tightening these constraints helps to ensure that portfolio weights are not driven by the sampling error that arises from parameter estimates based on historical data. Finally, cardinality constraints determine a lower and an upper bound for the number of assets included in the portfolio. Diversification is to some extent attained through the allocation of resources to a minimum number of imperfectly correlated assets, while the upper bound is dictated by the trade-off between diversification and incurred costs. Beyond a certain threshold, the marginal benefits of portfolio diversification decrease (Maringer, 2005), since portfolios with a large number of assets are more costly in terms of complexity, managerial effort, and non-negligible transaction costs. All in all, these additional constraints make the problem *NP-hard* (Bienstock, 1996), thus requiring the use of metaheuristics.

Contrary to the well-established real-life constraints, the growing body of literature assumes constant expected returns and covariances. This empirically unsupported assumption poses a key limitation when real-life approaches are sought, and the main contribution of this paper is to address this limitation. Indeed, since asset returns are random variables that obey certain probability density functions, and future returns are unpredictable, the required return may not be attained with certainty. More concretely, we relax the above simplifying assumptions and consider expected returns and covariances as random variables. In terms of the latter, we reasonably assume that covariances are unknown in practice. To estimate them, sample statistics computed from historical observations are employed. Hence, these estimates contain a certain degree of noise or uncertainty. The resulting problem is referred to as the stochastic POP (SPOP). This approach can be aligned with research that seeks to forecast volatility, which then feeds into the POP (Becker et al., 2015). Fig. 1 shows two resulting portfolios with different required returns for the Hang Seng stock market data with a medium level of covariance stochasticity (in this illustrative example, required returns are assumed constant). As expected, the higher the required return, the higher the expected risk and the lower the number of assets selected (only those with a relatively high expected return are considered).

The relation between the required return and the expected risk (i.e., the constrained efficient frontier or CEF) is analyzed in more detail in Fig. 2 for four different levels of stochasticity (zero, low, medium, and high). Circles on the vertical axis indicate the required returns of the two solutions analyzed in the previous figure, while crossings denote the expected risk of the first solution for each level of stochasticity.

In this paper, we propose a simheuristic algorithm to solve the SPOP. As described in Juan et al. (2018), simheuristic algorithms integrate metaheuristics with simulation techniques in order to deal with the random nature of stochastic COPs. In particular, we use

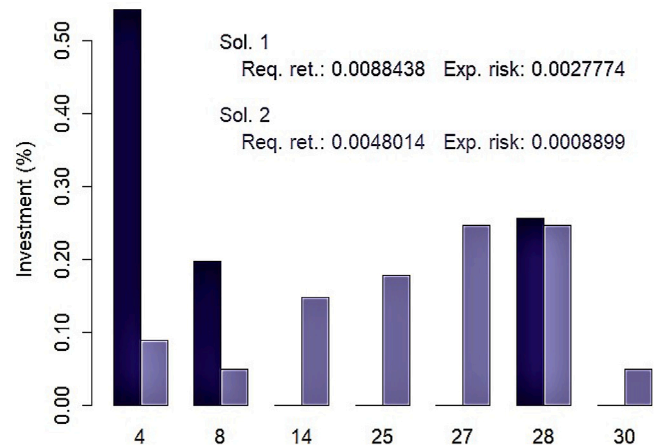

Fig. 1. Representation of two solutions for the SPOP.

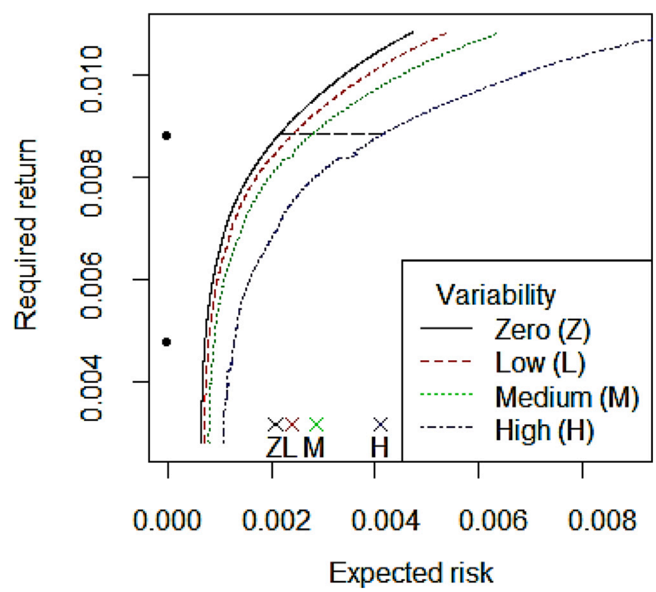

Fig. 2. CEFs of Hang Seng Stock Market (Hong Kong) data.

an extension of the variable neighborhood search (VNS) metaheuristic (Hansen et al., 2010) that integrates Monte Carlo simulation (MCS) techniques. While the metaheuristic generates promising portfolios for a deterministic version of the problem – which considers expected values – simulation techniques are applied to: *(i)* estimate the expected risk of these portfolios under uncertainty conditions; *(ii)* carry out a risk analysis on each portfolio; and *(iii)* provide feedback to the metaheuristic in order to better guide the searching process. All in all, the contribution of this article is threefold. First, we derive a mathematical formulation for a stochastic POP. Second, we develop an efficient solving methodology for the stochastic POP. Third, we illustrate the methodology by solving an adapted benchmark instance.

The remainder of this paper is organized as follows. Section 2 presents a literature review. In Section 3, the mathematical models for the POP and the SPOP are formulated. Section 4 proposes a solving methodology for the SPOP. Sections 5 and 6 explain the computational experiments carried out and analyze the results. Finally, Section 7 concludes.

2. Literature review

Implicit in the mean–variance efficient frontier (MVEF) proposed by Markowitz (1952) are the assumptions that: *(i)* first and second moments of a return distribution do not vary over time; and *(ii)* current and past asset price changes can be used to predict future movements in asset prices. It is worth noting that the first assumption has received little empirical support in the related body of research, whereas the second assumption rules out the efficient market hypothesis in the semi-strong efficiency form. Even so, the MVEF has received an overwhelming volume of interest from academics and practitioners for nearly seven decades and has spawn a large number of theoretical extensions and practical applications (Kolm et al., 2014). Notwithstanding the aforementioned shortcomings, there has been little attempt to reconcile the MVEF with the random and uncertain nature of returns on financial investments. Early research addresses uncertainty in the parameters (i.e. risk, return, and return covariances) through a robust model formulation of the mean–variance optimization (Bertsimas and Pachamanova, 2008). However, these formulations present a deterministic uncertainty model that adopts different uncertainty sets.

Huang (2012) addresses this issue and defines returns not as random, but as uncertain and exposed to subjective imprecision. The author thus combines the MVEF with uncertainty theory and bases returns mainly on expert opinions rather than historical data. A genetic algorithm (GA) is used to optimize the resulting mean–variance and mean-semivariance models. However, Qin (2015) argues that a distinction between different assets needs to be made, and addresses the real-world complexity of financial markets by solving a hybrid POP that distinguishes between random and uncertain returns. If there is ample historical information available on an asset, then the asset return is deemed a random variable. If, however, there is a lack of historical information on the asset – which applies to newly listed securities on a stock exchange – then the return is treated as fuzzy or uncertain. In the latter case, to estimate the first and second moments of asset returns experts' opinions are sought. In the presence of assets with random and uncertain returns, the author derives an optimal solution of the POP that considers assets from Shanghai Stock Exchange. Along similar lines, Qin (2017) formulates a general form of the random fuzzy mean-absolute deviation (MAD) model for portfolio optimization in a hybrid complex environment, and solves the POP by employing GA. Although the aforementioned work is an interesting and significant extension of the MVEF, it ignores several fundamental issues. First, in addition to the presence of random and uncertain asset returns, the investor might be less concerned with keeping a hypothetical expected return on or above a certain threshold. Indeed, it is unlikely that all the individual assets will take on expected returns at the end of the investment period. Instead, it would be more natural and practical for the investor to maximize the probability that the actual portfolio return is higher or equal to a certain threshold. Second, in real life, investors face additional constraints, such as cardinality, quantity and pre-assignment constraint. These issues are echoed in Nazemi et al. (2015). They formulate and solve several POPs under the assumption that returns behave as normal, rectangular and trapezoidal uncertain variables. This assumption allows reducing the POP to a linear programming (LP) problem. A neural network model is then designed to solve a dynamic version of the POP. An alternative measure of return uncertainty is introduced by Ning et al. (2015). In the standard MVEF, they consider a POP with a triangular entropy as an additional constraint that controls the level of uncertainty in the POP, whilst leaving unresolved the above mentioned issues. In a similar vein, Omidi et al. (2017) treat asset returns as normal and trapezoidal uncertain variables by means of a LP problem. To this end, the authors build on a neural network model to solve the POP. The first issue is partially alleviated by Huang (2007), who supports the argument of maximizing actual return (as opposed to expected return) on a portfolio of assets. Consequently, the author designs a hybrid intelligent algorithm to solve a stochastic POP, which seeks to maximize the actual portfolio return with a certain probability under the constraint that the ratio of the expected portfolio return to the portfolio variance is greater than a pre-set tolerance level. Whilst the article considers stochastic asset returns, it continues to treat second moments as constant. Moreover, the author's assumption that the investors face no further constraints in the POP still remains an interesting area of opportunity. Along similar lines, Liu et al. (2012) treat asset returns as fuzzy random variables, and formulates several POPs in hybrid uncertain decision systems. The first optimization problem consists of minimizing the portfolio variance subject to a chance constraint. The chance constraint specifies a required rate of return that can be attained under a prescribed chance level. The second problem caters to tail risk aversion. Specifically, if only the tail of a chance distribution matters for a risk averse investor, then she will maximize the chance that a fuzzy random return on a portfolio is at least as high as the required return under the maximum level of risk. To solve the above optimization problems, these authors employ a combination of MCS and a particle swarm optimization (PSO) algorithm. The simulation method is used to simulate probability density functions, whereas the PSO algorithm is utilized to solve stochastic programming problems. They further compare the computational results obtained with the chance-variance SPOP and with a conventional POP, and identify notable differences in optimal portfolio weights across the two settings. While the authors continue to treat the second moments as constant, they account for uncertainty in the constraints (chance or variance constraint depending on the investor's risk awareness) by further employing MCS in the simulation of probability density functions of the constraints. Zhang et al. (2012) further explore fuzzy modeling of returns and combine it with the mean-semivariance-entropy approach in a multi-period setting, in which they explicitly consider four dimensions to the POP; namely, the level of risk and return, transaction costs, and the degree of diversification of the portfolio. The resulting model is solved using a hybrid algorithm that combines a GA and simulated annealing (SA). The entropy measure for diversification is further employed in a mean–variance entropy approach by Chen (2015). Unlike Zhang et al. (2012), Chen (2015) further employs uncertainty theory in describing investment risk as an uncertain portfolio variance, and solves the resulting model using an artificial bee colony algorithm (ABC). The shortcomings of previous research are diminished by modeling the second moments as uncertain variables and considering further constraints, namely transaction costs. Recognizing that not only returns are subject to uncertainty, Nguyen et al. (2014) develops a model, in which the uncertainty of fuzzy portfolio returns is minimized, while the Sharpe ratio is initiated in a fuzzy context and is subsequently maximized. The resulting model is solved twice, employing a fuzzy approach and a GA; both provide superior solutions to the conventional POP based on the works by Markowitz (1952). More recently, Saborido et al. (2016) consider the mean-downside risk-skewness (MDRS) model, a cardinality-constrained portfolio selection model. This model optimizes the expected return, the downside-risk, and the skewness of a given portfolio, in which uncertain future returns are approximated by means of LR-fuzzy numbers. The authors construct three evolutionary algorithms; namely, the non-dominated sorting genetic algorithm (NSGAII), the multi-objective evolutionary algorithm based on decomposition (MOEA/D), and the global weighting achievement scalarizing function genetic algorithm (GWASF-GA) for solving the POP. Likewise, Liagkouras and Metaxiotis (2018) analyze the multi-period POP with transaction costs and fuzzy variables accounting for future returns. The study employs a multi-objective evolutionary algorithm (MOEA) to solve the POP with data from the FTSE 100 stock market index; it also assesses the proposed algorithm's performance with the NSGAII and the MOEA/D algorithms. The experimental results validate the superiority of MOEA over the NSGAII and the MOEA/D for all examined periods. Muthuraman and Zha (2008) solve a continuous-time portfolio optimization problem, in which the investor seeks to maximize the long-term expected growth rate, in a scenario with proportional transaction costs. The stock price is assumed to follow a geometric Brownian motion, considering a vector of local stock returns, and a positive definite symmetric matrix that represents the covariance structure. The authors focus on approaches based on partial differential equations, whose run time grows super-exponentially with the number of assets. They convert the free boundary problem into a sequence of fixed boundary problems, developing an approach that scales polynomially in dimension. Their algorithm is able to solve problem instances of seven assets within a reasonable time (several days). Bačević et al. (2019) consider a deterministic POP that features cardinality constraints, tracking error, active share, and turnover. The authors propose a VNS-based framework for solving the problem and present several neighborhood structures and rebalancing strategies. However, they neither use benchmark instances nor share their instances; indeed, no external validation is carried out. Anil Akbay et al. (2020) deal with a deterministic cardinality-constrained POP. The authors propose a hybrid approach that involves a VNS algorithm to select the combination of assets, and use quadratic programming to calculate the weights. A simple parallelization strategy is implemented to speed up the search. The approach is compared to other solution methods in the literature using five well-known datasets. According to the results, their methodology is efficient.

Based on previous work that recognizes the importance of stochastic modeling of asset returns, this paper takes a further important step in acknowledging stochasticity not only in stock returns, but also in covariances. Furthermore, it combines a novel simheuristic approach with rich constraints that have not previously been considered in stochastic POPs; namely pre-selection, quantity, and cardinality constraints. Calvet et al. (2016a) is an exception. Similar to our paper, Calvet et al. (2016a) solve the stochastic POP, which consists of pre-selection, quantity and cardinality constraints. However, differently from our paper, which assumes that both expected returns and covariances are stochastic, in Calvet et al. (2016a) only expected returns are assumed stochastic. The authors propose a simheuristic approach based on an iterated local search metaheuristic, and illustrate its use with a small experiment that tests how the solutions change in terms of expected risk when the required return (10 values are tested) and the probability of satisfying the constraint associated with the required return (two values) are allowed to vary. A more realistic and complex scenario with stochastic expected returns and covariances is examined in Calvet et al. (2016b). Specifically, the authors propose a simheuristic algorithm that relies on a SA metaheuristic. However, it should be noted that Calvet et al. (2016b) constitutes a preliminary work that presents detailed results for 10 values of the minimum required return. It also compares the results for two different probabilities of obtaining each value of the minimum required return no lower than a given threshold. By contrast, our paper presents a more efficient algorithm (as discussed in Section 6), as well as a more comprehensive and exhaustive set of experiments, in which best deterministic and stochastic solutions are compared considering three stochasticity levels and 20 values of the required return. In addition, it includes an analysis of results where computational times and confidence intervals for average risks pertaining to the best solutions are examined.

3. Problem definition

This section describes the POP and SPOP. It also outlines mathematically the objective function and constraints for each problem, which are informed by Tollo and Roli (2008).

3.1. Notation and description

In the POP, suppose there is a set $A = \{a_1, a_2, \dots, a_n\}$ of n assets. Each asset a_i ($\forall i \in \{1, 2, \dots, n\}$) is characterized by an expected return r_i. The covariance between two assets a_i and a_j ($\forall i, j \in \{1, 2, \dots, n\}$) is denoted by σ_{ij}. A solution to this problem consists of a vector $X = (x_1, x_2, \dots, x_n)$, where each element $x_i (0 \le x_i \le 1)$ represents the weight or fraction of the investment allocated to the asset a_i. The aim is to minimize the portfolio variance such that an expected return greater than an investor-given threshold, R, is attained. To cater for real-life scenarios, a rich version of the POP considers pre-selection, quantity, and cardinality constraints. Pre-selection constraints dictate whether an asset a_i is pre-selected by the investor and included in the solution (i.e., $x_i > 0$) by means of the parameter p_i: $p_i = 1$ if a_i is pre-selected, and $p_i = 0$ otherwise. Quantity constraints specify a lower and an upper bound for the weights x_i, ϵ_i and δ_i ($0 \le \epsilon_i \le \delta_i \le 1$), respectively. Finally, the cardinality constraint defines a minimum and a maximum number of assets included in the portfolio, k_{min} and k_{max} ($1 \le k_{min} \le k_{max} \le n$).

The difference between the POP and the SPOP considered in this paper lies in the modeling of asset returns and covariances. While in the POP they are assumed to take on expected values, the SPOP highlights the stochastic/uncertain nature of both expected returns and covariances, and hence treats them as random variables. This translates into a modified required return constraint where a return no lower than R is required by the investor with a probability of, at least, P_0.

3.2. Mathematical model

The POP can mathematically be defined as follows:

$$\min f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \sigma_{ij} x_i x_j \tag{1}$$

subject to:

$$\sum_{i=1}^{n} r_i x_i \ge R \tag{2}$$

$$\sum_{i=1}^{n} x_i = 1 \tag{3}$$

$$\epsilon_i z_i \le x_i \le \delta_i z_i, \forall i \in \{1, 2, \dots, n\} \tag{4}$$

$$0 \le \epsilon_i \le \delta_i \le 1, \forall i \in \{1, 2, \dots, n\} \tag{5}$$

$$z_i \le M x_i, \forall i \in \{1, 2, \dots, n\} \tag{6}$$

$$p_i \le z_i, \forall i \in \{1, 2, \dots, n\} \tag{7}$$

$$k_{min} \le \sum_{i=1}^{n} z_i \le k_{max} \tag{8}$$

$$z_i \in \{0, 1\}, \forall i \in \{1, 2, \dots, n\} \tag{9}$$

As aforementioned, the POP is solved by minimizing the objective function in Eq. (1), which quantifies investment risk. Eq. (2) guarantees that the expected return on the investment does not fall below the threshold R. Eq. (3) restrains the investment to existing and pre-defined resources. Further, an auxiliary variable is introduced, which indicates if the asset a_i is included in the solution ($z_i = 1$; $z_i = 0$ otherwise). For all assets a_i, Eq. (4) sets a lower and an upper bounds (ϵ_i and δ_i, respectively) for x_i if the asset is selected (i.e., $z_i = 1$). The two bounds are themselves constrained to vary between zero and one inclusive (Eq. (5)). In Eq. (6), M is a very large positive value such that $M x_i \ge 1$ for all i if $x_i > 0$. Eq. (7) defines the pre-assignment constraint, where z_i depends on the parameter p_i. If the asset a_i is pre-selected (i.e., $p_i = 1$), it must be included in the solution (i.e., $z_i = 1$) irrespective of its risk-return characteristics. Eq. (8) ensures that the number of assets included in the portfolio ranges between a minimum and a maximum integer values. Finally, Eq. (9) defines z_i as a binary variable. The mathematical formulation of the SPOP entails two modifications:

First, covariances (C_{ij}) in the objective function are regarded as random variables that follow a given probability distribution (e.g., the one that best fits the historical data available):

$$f(x) = \Theta \left[\sum_{i=1}^{n} \sum_{j=1}^{n} C_{ij} x_i x_j \right] \tag{10}$$

Second, Eq. (2) is replaced with the following probabilistic constraint:

$$P\left(\sum_{i=1}^{n} R_i x_i \geq R\right) \geq P_0 \tag{11}$$

where Θ refers to a specific statistic (such as the mean, the median, or the variance), and R_i refers to the asset return modeled as a random variable. It ensures that the portfolio return will be no lower than the threshold return R with a probability of, at least, P_0.

4. Proposed methodology for the SPOP

In this section, we propose a simheuristic algorithm to solve the SPOP. Generally, this class of algorithms relies on two facts. First, a stochastic COP can be considered as a generalization of the deterministic COP; this signifies that the latter can be interpreted as a special case of the former when variances are zero. Second, although stochastic COPs have not been extensively examined (Bianchi et al., 2009), there is a vast literature on deterministic COPs. Thus, our approach consists of *(a)* selecting an efficient algorithm for the deterministic version of a POP, and *(b)* extending it in a natural way by hybridizing it with simulation techniques. This section introduces the proposed simheuristic as well as relevant elements of its implementation in order to allow reproducibility.

4.1. VNS metaheuristic and implementation components

As a base framework we employ the VNS metaheuristic, which was pioneered by by Mladenović and Hansen (1997). Since then, VNS has become a popular metaheuristic in combinatorial as well as global optimization. It has been used in a wide range of research fields such as scheduling, vehicle routing, telecommunications, biology, and artificial intelligence. For extensive reviews on applications, the reader is referred to Moreno-Vega and Melián (2008). In a nutshell, the VNS metaheuristic proposes systematic changes to a neighborhood in order to find a local minimum by intensifying the search, and to escape from a valley by diversifying. It relies on three facts (Hansen et al., 2010): *(i)* a local minimum with respect to one neighborhood structure is not necessarily so for another; *(ii)* a global minimum is a local minimum with respect to all possible neighborhood structures; and *(iii)* for many problems, local minima with respect to one or more neighborhoods are relatively close to each other. Pseudocode 1 shows a simple version of the VNS used in this work. Its inputs entail the problem instance, the number of neighborhoods considered (K), and the maximum computational time (T). Frequently, K is set to two or three, and the neighborhoods are nested. First, the variable t, which measures time, is initialized at zero. Subsequently, an initial solution is obtained and stored in $currentSol$. An outer loop sets the current neighborhood to the first one and controls that the time-based constraint is satisfied. Inside, another loop builds and tests new solutions. Within this loop, the current solution is initially 'shaken' (or perturbed), generating a solution from the kth neighborhood of $currentSol$. The resulting solution is stored in $newSol$, which is then improved by means of a local search. If there is an improvement (i.e., $newSol$ is preferred over $currentSol$), $newSol$ is copied into $currentSol$ and the current neighborhood is set to the first. This constitutes a descendent phase aimed to find a local minimum. Otherwise, the next neighborhood is analyzed (i.e., k is set to $k+1$). The inner loop is executed until the last neighborhood is explored (i.e., $k = K$). Finally, $currentSol$ is returned.

Our implementation of the VNS metaheuristic includes biased randomization techniques (Juan et al., 2020), which entail the introduction of randomization in classical deterministic heuristics in order to obtain a number of solutions. Heuristics include at least one step, in which one element is to be selected from a list sorted according to a specific criterion. Typically, the first one is expected to be the optimal option to construct a high-quality solution. For instance, in the permutation flow-shop problem (Fernandez-Viagas and Framinan, 2015), which determines a sequence of jobs with the aim of minimizing a time-related measure, the classical NEH heuristic creates a list of jobs that are ordered from more to less time-consuming, iteratively extracts the first one, and then chooses the best allocation in the solution. While this is an intuitively logical procedure, it does not necessarily lead to the best solution. In Dominguez et al. (2016) and Gonzalez-Neira et al. (2017), the authors propose assigning a probability to each element in the sorted list according to the reference measure (i.e., the higher in the list an item is ranked, the higher the probability of choosing the specific item). Consequently, different solutions are generated when the procedure is executed repeatedly, and it can be expected that some of them will be of significantly higher quality than the one obtained with a deterministic heuristic. In order to efficiently implement these ideas in a code, they make use of skewed probability distributions, such as the geometric or the decreasing triangular ones. All in all, the biased randomization approach enables us to obtain multiple solutions based on a deterministic heuristic. As a further advancement, our algorithm includes MCS, which is applied during the search to assess the performance of a number of solutions in a stochastic environment by generating scenarios with specific values for each random variable, and computing the expected value of the objective function (i.e., the expected portfolio risk). The open-source quadratic programming solver ojAlgo (http://ojalgo.org), developed in Java, is called to determine the optimal weights allocated to a given set of assets. Additionally, a cache memory (implemented as a hash map data structure) is used in order to avoid calling the solver repeatedly for a specific set of assets.

Algorithm 1 Basic structure of the VNS metaheuristic.

```
VNS(instance, K, T)
t ← 0
initSol ← initialSolution(instance)
currentSol ← initSol
while {t < T} do
    k ← 1
    while {k ≤ K} do
        newSol ← shake(currentSol,k)
        newSol ← localSearch(newSol)
        if {newSol > currentSol} then
            currentSol ← newSol
            k ← 1
        elsek ← k+1
        end if
    end while
    t ← elapsedTime
end while
return currentSol
```

4.2. Details of the proposed methodology

Combining metaheuristic optimization with simulation (in any of its forms), simheuristics have been employed during the last years to address *NP-hard* stochastic optimization problems in areas such as: flow-shop scheduling problems with random processing times (Gonzalez-Neira et al., 2017; Hatami et al., 2018), vehicle routing problems with stochastic travel times (Guimarans et al., 2018), inventory routing problems with stochastic demands (Onggo et al., 2019), and stochastic project portfolio selection problems (Panadero et al., 2020). The flowchart diagram of our simheuristic approach is depicted in Fig. 3 and described next:

1. Consider a SPOP instance defined by n assets. Each asset a_i ($\forall i \in \{1, 2, \ldots, n\}$) yields a rate of return R_i, which is a random

variable distributed with either an empirical or theoretical probability distribution. Returns on any two assets a_i and a_j ($\forall i, j \in \{1, 2, \ldots, n\}$) covary with a covariance C_{ij}. The covariance is assumed random; it is directly related to the correlation P_{ij}, and inversely related to the standard deviations S_i and S_j according to $P_{ij} = \frac{C_{ij}}{S_i S_j}$

2. In this step, the original stochastic POP is transformed into a POP instance by replacing the random variables with their expected values r_i and σ_{ij}.
3. In this step, an initial solution ($initSol$) is constructed by selecting k_{min} assets that offer the highest rate of return, having previously included s assets pre-selected by the investor, and calling the solver. Subsequently, simulation techniques are employed to compute the probability of satisfying the minimum required return constraint in the stochastic environment described by the original instance. In particular, a short number of scenarios (sim_{short}) is used to simulate returns. The solution is stored, and the algorithm progresses to the fourth step, provided that the constraint is satisfied. If this is not the case, a feasible solution is searched using a randomized and iterative procedure. This procedure involves two sub-steps. In the first sub-step, the pre-selected assets compose a portfolio. In the second sub-step, the non-preselected assets are ordered according to the expected rate of return, and a random number, between $k_{min} - s$ and $k_{max} - s$, are selected using biased randomization, relying on a geometric distribution with a parameter β (Estrada-Moreno et al., 2019). All weights are set to the minimum value initially. Thereafter, each weight is set to the maximum value (taking into account for an asset a_i the following elements: ε_i, δ_i, and the fraction that remains unallocated, i.e., $1 - \sum_{i=1}^{n} x_i$) in the previously established order. If an initial solution can be constructed in this step, one moves to step 4. It is worthwhile to remark that we avoid using the solver in this step because the focus is on finding an initial feasible solution considering the stochastic environment and not the solution that carries the lowest risk. The time allocated to searching for a feasible solution is limited by T_{init}, and the algorithm stops if no feasible solution is obtained.
4. A list $bestSols$ is created for storing l best-found solutions in terms of expected risk. Then, $initSol$ is copied into $currentSol$ and k is set to one. Following this, the expected risk of $currentSol$ is computed by means of MCS, and the solution is incorporated in the created $bestSols$ list.
5. An iterative procedure commences, and steps 6 and 7 are executed within a given time window (T_{loop}).
6. A new solution ($newSol$) is created by 'shaking' the current one. This procedure consists of randomly erasing a number of non pre-selected assets in the solution and randomly introducing new assets until k_{max} is reached. The number of assets erased is determined by k. Moreover, a local search is applied to the solution. It aims to improve the solution by replacing the asset that carries the lowest weight with another one from the list of non-selected assets.
7. $newSol$ is compared with $currentSol$. If the former is better in terms of risk associated with the deterministic version of the problem, $newSol$ is considered to be a promising portfolio and the minimum required return constraint for the stochastic environment is checked for. If the return constraint is satisfied, the expected risk is computed for the stochastic version of the problem. If the expected risk of $newSol$ is better than that of $currentSol$, then $newSol$ replaces $currentSol$, k is set to one, and $bestSols$ is updated. If it is not satisfied, the solution is discarded. If $newSol$ is not better, k is increased by one unit if $k < K$, or set to one otherwise.
8. When the iterative procedure terminates, the algorithm returns $bestSols$. For each solution, a sample of risk measures is obtained by simulating a large number of scenarios (sim_{large}). We perform a risk analysis where solutions are compared by using risk distributions. Concretely, the risk analysis is based on the expected values and the variances of the distributions, and the reliabilities (or probabilities of satisfying the minimum required return constraint). Accordingly, the Pareto dominant solutions (i.e., solutions not dominated by another portfolio for one measure while the other measures are at least equally good) are reported to the decision-maker.

In addition to being a notoriously simple and natural, our algorithm is also efficient, which can generate solutions in real time (see Section 5). However, since MCS techniques tend to be time-consuming, their use is minimized by conducting only a few runs to assess promising solutions (Rabe et al., 2020), and a more thorough simulation at the end of the procedure. Crucially, our algorithm can provide decision makers with reliable solutions, which can be used as a basis for portfolio construction, optimization and management.

5. Computational experiments

The outlined algorithm was implemented as a Java application. A standard personal computer, Intel Core i5 CPU at 3.2 GHz and 4 GB RAM with Windows 7 was used to perform all tests. Our algorithm was executed ten times using different seeds; only the best results were stored. We experimented with stock market data from the repository ORlib (http://people.brunel.ac.uk/~mastjjb/jeb/orlib/portinfo.html). These datasets were described by e.g., Chang et al. (2000), and have been extensively analyzed by other scholars in this research field (Schaerf, 2002; Armañanzas and Lozano, 2005; Moral-Escudero et al., 2006; Fernández and Gómez, 2007; Gaspero et al., 2011). These datasets were originally retrieved from Datastream, which sources global financial and macroeconomic data. Specifically, we used the Hang Seng (Hong Kong), FTSE 100 (UK), and Nikkei 225 (Japan) stock market indices at weekly frequency that span the period from March 1992 to September 1997. Stocks with missing values were dropped. The datasets comprise the mean return and the return standard deviation for constituent stocks from the stock market index, as well as the correlation coefficients for all possible pairs of assets. Following Gaspero et al. (2011), the portfolio frontier was divided into 100 equidistant points on the axis representing the expected portfolio return.

These three benchmark instances are deterministic. In order to evaluate our simheuristic approach, they were adapted by substituting the deterministic returns and covariances with random variables. More specifically, we considered three complementary scenarios, which add varying degrees of uncertainty to the deterministic POP:

1. S_i (stochastic standard deviation) follows a $LN(\mu_S, \sigma_S)$, where LN represents a log-normal distribution, and μ_S and σ_S are the mean and the standard deviation, respectively, of the natural logarithm of S_i. They are assumed to take on the values of σ_i and $c\sigma_i$, respectively, where σ_i is the standard deviation of the variable, c is an input.
2. P_{ij} (stochastic correlation) is assumed to follow a truncated normal distribution $TN(\mu_P, \sigma_P, l, u)$, where μ_P is the mean, σ_P is the standard deviation, and l and u are the lower and upper bounds, respectively. μ_P, the mean correlation between the returns on assets a_i and a_j, is assumed to take on value ρ_{ij}, and σ_P is an input. To assure that the correlation varies between -1 and 1, l and u are set to -1 and 1, respectively. A special case is when $i = j$, in which case l and u are equal to 1 (i.e., $P_{ij} = 1$).
3. R_i (stochastic portfolio return) follows a $N(\mu_R, \sigma_R)$, where μ_R and σ_R are the mean and the standard deviation of the variable, respectively, i.e., r_i and S_i, respectively.

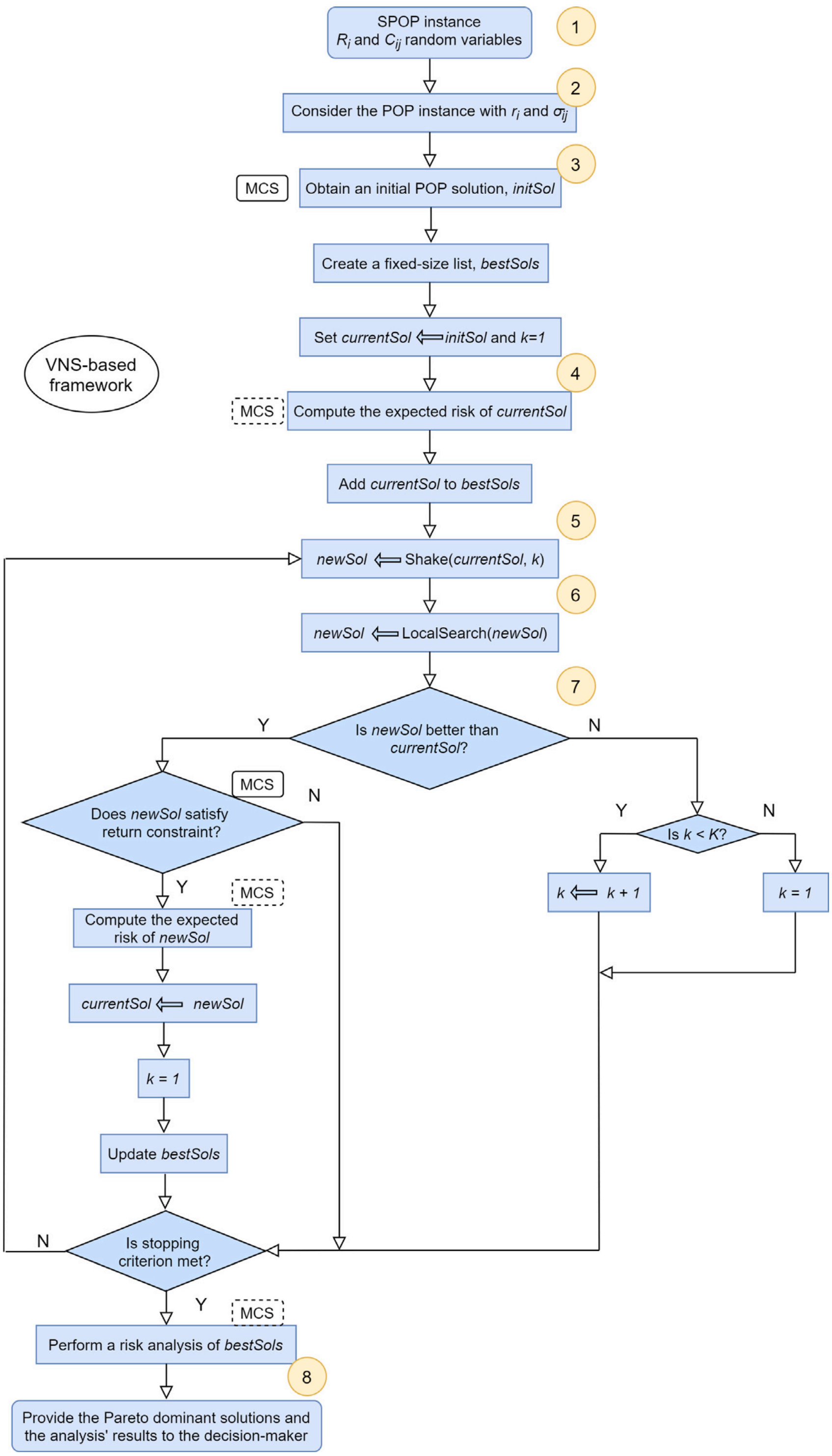

Fig. 3. Flowchart of the simheuristic algorithm.

Three values for $c(0.01, 0.025, 0.08)$ and $\sigma_P(\sqrt{0.00002}, \sqrt{0.0002}, \sqrt{0.002})$ were tested to account for three different levels of stochasticity, from lowest to highest. These values were informed by preliminary tests that aimed to explore a 'reasonable' range of values for each parameter. Fig. 4 displays the probability distributions of the standard deviation with a mean of 0.0472 (average standard deviation of asset returns) (left) and the distributions for the correlation with a mean of 0.5562 (average standard correlation between asset returns) (right). It is worth noting that our simulation-based methodology admits any probability distribution, either theoretical or empirical (in a real-life scenario, empirical data would be employed to find the best fit distribution for each random variable considered). In order to illustrate our approach, three computational experiments were carried out. The first experiment considers stochastic covariances (Scenarios 1 and 2). Building on the first experiment, the second one additionally introduces stochastic returns (Scenarios 1–3). While the first two experiments center on the Hang Seng (Hong Kong) stock market index (which comprises 31 assets), the third experiment evaluates the performance of our approach on the FTSE 100 (UK) and Nikkei 225 (Japan) stock indices (with 89 and 225 assets, respectively).

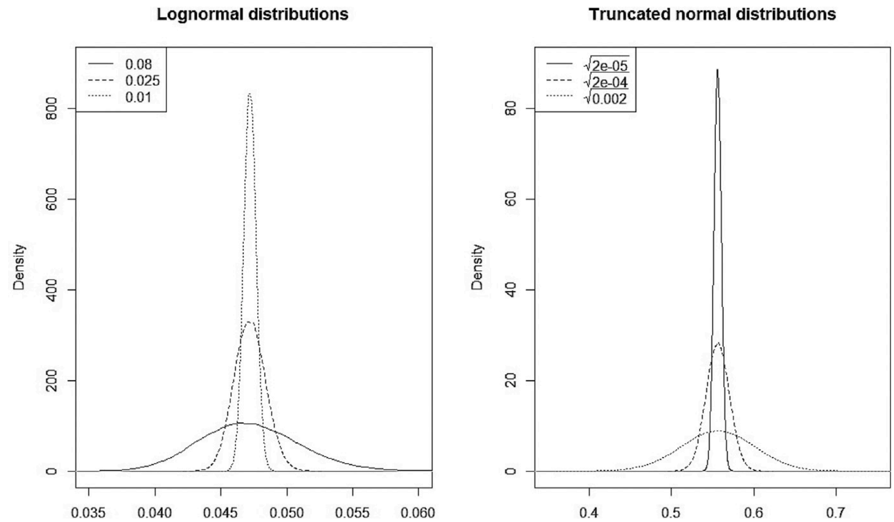

Fig. 4. Probability distributions.

Table 1
Hang Seng (Hong Kong) Stock Market with stochastic covariances.

Required return	Deterministic solution				Stochastic solutions			Time (s.)	Gaps (%)				
	Risk (1)	E.R. - Low (2)	E.R. - Med. (3)	E. R. - High (4)	E.R. - Low (5)	E.R. - Med (6)	E.R. - High (7)		(2-1)	(5-1)	(5-2)	(6-3)	(7-4)
0.002861137	0.0006424	0.0007055	0.0007975	0.001114	0.0006948	0.000777	0.0010713	0.211	0.0982	0.0815	−0.0152	−0.0257	−0.0383
0.002941981	0.0006429	0.0007062	0.0008052	0.001195	0.0006952	0.0007772	0.0010701	0.409	0.0985	0.0813	−0.0156	−0.0348	−0.1045
0.003022827	0.0006437	0.0007015	0.0007888	0.0011057	0.0006961	0.0007782	0.0010706	0.627	0.0897	0.0814	−0.0076	−0.0135	−0.0318
0.003103671	0.0006444	0.0007108	0.0008089	0.0011647	0.0006976	0.0007799	0.0010727	0.456	0.1031	0.0826	−0.0186	−0.0358	−0.079
0.003184516	0.0006455	0.0007054	0.0007978	0.00113	0.0006996	0.0007824	0.0010764	0.369	0.0929	0.0838	−0.0083	−0.0192	−0.0474
0.003265361	0.0006468	0.0007057	0.0007951	0.0011208	0.0007011	0.000785	0.0010821	2.03	0.0912	0.084	−0.0065	−0.0127	−0.0346
0.003346206	0.0006483	0.0007113	0.0008062	0.0011714	0.0007021	0.0007857	0.0010806	2.461	0.0972	0.083	−0.0129	−0.0255	−0.0775
0.003427051	0.00065	0.0007143	0.0008095	0.0011649	0.0007035	0.0007869	0.0010809	1.541	0.0989	0.0824	−0.015	−0.0279	−0.0721
0.003507896	0.0006517	0.0007161	0.0008037	0.0010852	0.0007103	0.0007887	0.0010822	5.966	0.0989	0.09	−0.0081	−0.0187	−0.0028
0.00358874	0.0006536	0.0007103	0.000799	0.0011332	0.0007076	0.0007912	0.0010848	0.581	0.0867	0.0826	−0.0038	−0.0098	−0.0427
0.010137479	0.0035774	0.0040554	0.0047712	0.0074653	0.0039666	0.0045855	0.0067181	0.001	0.1336	0.1088	−0.0219	−0.0389	−0.1001
0.010218315	0.0036908	0.004178	0.0049221	0.0076114	0.0040965	0.0047415	0.0069622	0.002	0.132	0.1099	−0.0195	−0.0367	−0.0853
0.010299151	0.0038091	0.0043348	0.0051217	0.0079629	0.0042323	0.0049052	0.0072194	0.001	0.138	0.1111	−0.0236	−0.0423	−0.0934
0.010379986	0.0039324	0.004503	0.0053405	0.0083909	0.0043741	0.0050764	0.0074899	0.001	0.1451	0.1123	−0.0286	−0.0494	−0.1074
0.010460822	0.0040605	0.0046604	0.0055344	0.0087777	0.0045219	0.0052551	0.0077734	0.001	0.1477	0.1136	−0.0297	−0.0505	−0.1144
0.010541657	0.0041937	0.004762	0.0055848	0.0083847	0.0046757	0.0054415	0.0080701	0.001	0.1355	0.1149	−0.0181	−0.0257	−0.0375
0.010622493	0.0043317	0.0048879	0.005757	0.0089222	0.0048354	0.0056353	0.00838	0.001	0.1284	0.1163	−0.0107	−0.0211	−0.0608
0.010703329	0.0044747	0.0051485	0.0061357	0.0096893	0.0050011	0.0058368	0.008703	0.001	0.1506	0.1176	−0.0286	−0.0487	−0.1018
0.010784164	0.0046226	0.0052806	0.006288	0.010186	0.0051728	0.0060458	0.0090391	0.001	0.1423	0.119	−0.0204	−0.0385	−0.1126
0.010865	0.0047755	0.0055134	0.0065833	0.010458	0.005381	0.0063068	0.0096622	0	0.1545	0.1268	−0.024	−0.042	−0.0761
							Average	0.733	0.1182	0.0992	−0.0168	−0.0309	−0.071

The parameter fine-tuning of our algorithm was performed taking into account suggestions of other authors and results from fast experimental tests. The recommended number of neighbors (K) is 3 (Hansen et al., 2010). A movement in each neighbor involves changing 25%, 35%, and 45% of the assets, respectively. 10 best stochastic solutions are used to perform intensive simulation and risk analysis. As suggested in Estrada-Moreno et al. (2019), β is randomly selected from a uniform distribution with parameters 0.05 and 0.25. Finally, sim_{short} and sim_{large} are set to 2500 and 12 500, respectively, T_{init} and T_{loop} are set to 5 and 15, respectively.

6. Analysis of results

6.1. First experiment: stochastic covariances

Table 1 summarizes the results of the first experiment, which uses the Hang Seng (Hong Kong) stock market index and where only the covariances are assumed stochastic.

The first experiment essentially contrasts and compares two types of solutions: *(i)* the best-found solution to the deterministic version of the problem (BDS); and *(ii)* the best-found solution to the stochastic version of the problem (BSS). We consider different levels of stochasticity in the covariances and expected returns. For each of these stochasticity levels (low, medium, and high) a different stochastic scenario is defined. It is worth noting that portfolio configurations obtained for the deterministic version of the problem can also be used as investment plans for a stochastic version of the problem – even when good solutions for the deterministic version might constitute sub-optimal solutions in a stochastic scenario. Put it differently, each portfolio configuration has a different risk measure (cost) for each different environment (deterministic or stochastic). In particular, we are interested in the following risk measures (costs) associated with the BDS portfolio configuration: the risk obtained when employing the BDS in a deterministic scenario, and the expected risk associated with the BDS in each of the stochastic scenarios. To some extent, the former could be considered as a lower bound for the BSS, while the latter could be considered as an upper bound for the BSS. The information gathered in columns is explained next. The first column displays the required return, showing only the first and the last 10 values. The next four columns depict the BDS. The following three columns contain the expected risk associated with the BSS for each of the stochastic environments analyzed. Also, average computational times needed to find the BSS are provided. The last five columns summarize several gaps: *(i)* the gap between the risk and the expected risk for a low level of stochasticity of the BDS; *(ii)* the gap between the risk of the BDS and the expected risk of the BSS in a low stochasticity environment, where the former is the lower-bound for the expected risk of both the BDS and the BSS; *(iii)* the gap between the expected risks for the BDS and the BSS considering a low stochasticity scenario, which quantifies the benefit of using the simheuristic approach instead of assuming constant values; *(iv)* the same gap as in *(iii)*, but in a medium stochasticity scenario; and *(v)* the same gap as in *(iii)* and *(iv)*, albeit in a high stochasticity scenario. Additionally, the average of each ratio was added at the bottom of the table. Fig. 5 illustrates boxplots of the expected risk gaps between the

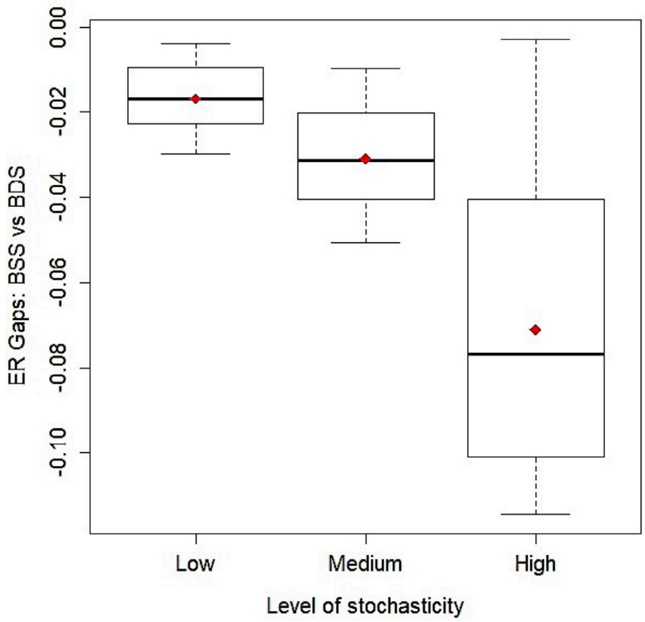

Fig. 5. Risk gaps between the best deterministic and stochastic solutions for different levels of stochasticity (environments).

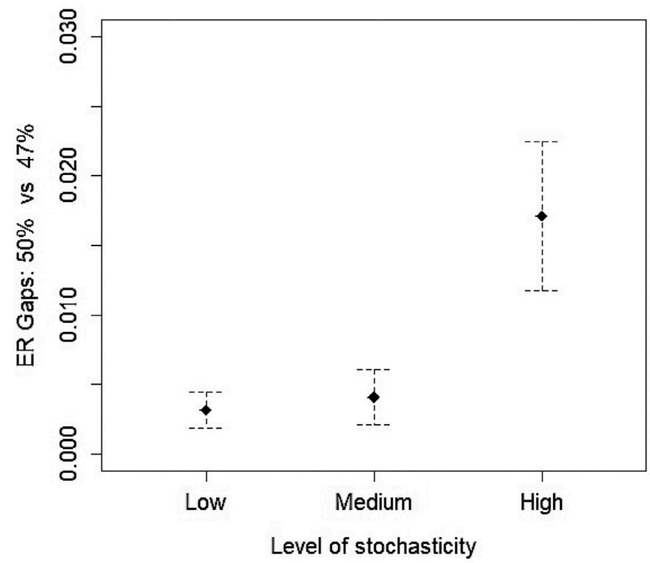

Fig. 6. Confidence intervals for the average expected risk gaps between solutions considering different probabilities.

BDSs and BSSs for the various stochasticity scenarios. The expected risk of the BDS is subtracted from that of the BSS; hence, a negative gap indicates an improvement in the expected risk of the solution. Mean values are represented with diamonds. Table 1 indicates that our algorithm is able to obtain a reasonably good BSS in 0.733 s on average. It also shows that the average gaps between the risk of the BDS and the expected risk of the BDS (when used as a portfolio configuration for the stochastic environment), and the BSS are quite large even for the low-stochasticity scenario (11.82% and 9.92%, respectively). As anticipated, the expected risk of the BSS is closer to the lower-bound (the risk) than the expected risk implied by the BDS. Thus, the simheuristic approach carries considerable benefits compared with the assumption of constant expected risk.

These benefits, illustrated in Fig. 5, are measured by the average gaps of −1.68%, −3.09%, and −7.10% for low, medium and high levels of stochasticity, respectively. It is important to remark that the gaps are never positive. Thus, the BSS shows a lower expected risk than the BDS when the latter is used to solve a stochastic version of the problem. Furthermore, the performance of the BDS deteriorates when covariances become more uncertain (i.e., as the stochasticity level increases). Intuitively, the BDS can be thought of as a restricted version of the BSS. Such a restriction is costly in terms of the expected risk, which justifies the observed gap between the BDS and the BSS. Importantly, our research findings are indicative of a potential bias in the solution to a deterministic POP when in fact covariances are stochastic. The size of this bias grows larger when covariances become more uncertain. To correct for this bias, a stochastic POP ought to be solved instead of the deterministic POP.

6.2. Second experiment: Stochastic covariances and returns

Results from the second experiment are displayed in Table 2.

As in the previous table, the first column displays the required return. Columns 2, 3, and 4 detail the expected risk of the BSSs under the low, medium, and high levels of stochasticity, under the assumption that the probability of attaining the required return is 50%. Columns 5, 6, and 7 visualize the gaps between the expected risk of the BSSs under the low, medium, and high levels of uncertainty, when the probabilities of attaining the required return are set to 50% and 47%, respectively. It is worth noting that our benchmarks extend the classical (deterministic) ones by considering probabilistic constraints. These probabilistic constraints force the selected portfolio to reach the required return with a given probability. This probability threshold can be allowed to vary by the portfolio manager within a ‘reasonable’ range of values; i.e., if the required return is too high or the probability threshold is too high, then the problem can become unfeasible. The average gaps are shown in the last row. The average computational times are reported in the last column.

Further, Fig. 6 displays the confidence intervals for the average expected risk gaps between the solutions for the two probability levels considered. As standard in most studies, a confidence level of 95% is assumed. These confidence intervals are generated from the observations provided by MCS. This figure shows that the expected risk gap between the BSS with the probabilities of 47% and 50% is relatively small, although it can grow larger for higher stochasticity levels. The average values for the different environments are: 0.31%, 0.39%, and 1.84%. Therefore, the gap increases with the level of stochasticity. The confidence intervals, depicted in Fig. 6, indicate that the average risk gaps are positive and significant. Moreover, while they overlap for the low and medium levels of stochasticity (although the point estimate is slightly higher for the latter), this is not the case with the confidence level for the high stochasticity level.

For illustrative purposes, we focused on the results associated with a specific required return, 0.0028611366, a probability of 50%, and the environment with a high level of stochasticity. Fig. 7 shows the confidence intervals for the average risks, and the reliabilities for the BDS and the best three BSSs. Here, the reliability of a given solution refers to the probability of obtaining a return higher than the threshold, and it is generated by MCS. The point estimates for the BSSs are lower compared with the BDS, but the differences among the three BSSs are relatively small. The lowest variability (i.e., the narrowest interval) is associated with the BSS2. The visualized solutions yield similar reliability levels too. Overall, the BSS3 can be discarded (i.e., it is Pareto-dominated), since it has the lowest reliability value, a relatively high point estimate, and a wide interval. Ultimately, the decision-maker will select the best solution that is taylored to his/her preferences. In this case, most decision-makers would probably choose BSS1, since it offers a relatively low average risk and a high reliability level.

Table 2
Hang Seng (Hong Kong) Stock Market with stochastic covariances and correlations.

Required	ER (50%)			ER gaps [%] (50%–47%)			Time (s)
Return	Low	Medium	High	Low	Medium	High	
0.002861137	0.0007006	0.0007872	0.0011358	0.00%	0.00%	4.38%	1.244
0.002941981	0.0007006	0.0007873	0.001135	0.00%	0.00%	4.44%	2.749
0.003022827	0.0007019	0.0007882	0.0011272	0.00%	0.00%	3.66%	2.647
0.003103671	0.0007027	0.00079	0.0011392	0.00%	0.00%	3.24%	1.7
0.003184516	0.0007068	0.0007925	0.0011075	0.33%	0.00%	1.23%	2.715
0.003265361	0.0007093	0.0007954	0.0011243	0.66%	0.00%	2.15%	1.438
0.003346206	0.0007085	0.000796	0.001137	0.09%	0.19%	2.67%	0.633
0.003427051	0.0007085	0.0007983	0.0011017	−0.11%	0.14%	0.00%	0.764
0.003507896	0.0007138	0.000799	0.0011144	0.54%	0.02%	0.05%	1.365
0.00358874	0.0007161	0.0008014	0.0011068	0.39%	0.00%	0.00%	1.726
0.010137479	0.0040004	0.0046595	0.0071471	0.00%	0.00%	1.05%	1.631
0.010218315	0.0041312	0.0048569	0.0074155	0.00%	0.82%	1.15%	2.8
0.010299151	0.004268	0.0049834	0.0076973	0.00%	0.00%	1.23%	1.421
0.010379986	0.0044359	0.0052031	0.0079926	0.57%	0.90%	1.31%	2.937
0.010460822	0.0045867	0.0053878	0.0083013	0.59%	0.93%	1.37%	4.313
0.010541657	0.0047435	0.0055803	0.0086235	0.61%	0.97%	1.42%	2.31
0.010622493	0.0049063	0.0057805	0.0089592	0.63%	1.00%	1.46%	3.76
0.010703329	0.0050752	0.0059884	0.0093083	0.65%	1.02%	1.49%	1.836
0.010784164	0.0052501	0.006204	0.0096709	0.67%	1.05%	1.52%	0.773
0.010865	0.0054126	0.0063772	0.0098998	0.37%	0.70%	3.06%	1.729
			Average	0.30%	0.39%	1.84%	

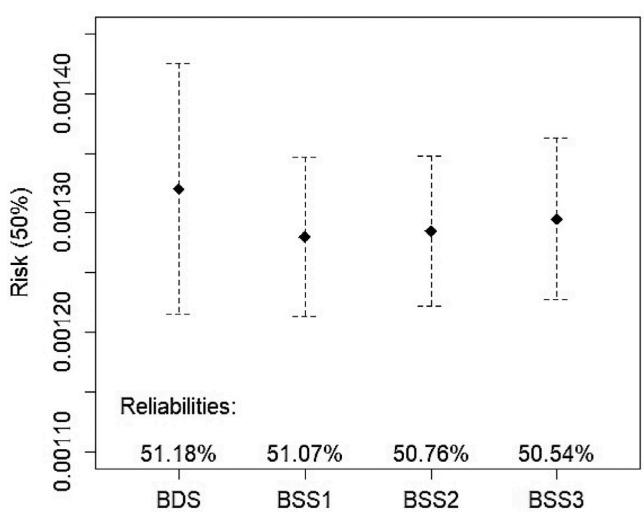

Fig. 7. Confidence intervals for the average risks from the best deterministic and stochastic solutions.

6.3. Third experiment: Additional instances

Results from the third experiment are displayed in Tables 3 and 4. The former refers to the FTSE 100 (UK) stock market and the latter to the Nikkei 225 (Japan) stock market.

In these tables, the first column displays the required return. Columns 2 and 3 detail the expected risk of the BSSs under the low level of stochasticity, given the probabilities of 47% and 50% for attaining the required return, respectively. Column 4 reports the expected risk gaps between these two probability levels. Columns 5, 6, and 7 provide analogous information for the medium level of stochasticity, while the last three columns depict the results for the high level of stochasticity. In the last row of Tables 3 and 4, the average expected gap is shown. The average computational time for each execution was 3.5 s. The average results from using 10 seeds are used.

As expected, all risk gaps are strictly positive. For the FTSE 100 (UK) stock market index, the gaps are relatively similar across the different levels of stochasticity, ranging from 5.7248 to 5.8140, while there are more significant changes for the Nikkei 225 (Japan) stock market index (ranging from 13.2286 to 22.0106). All in all, it can be concluded that there are larger differences between the expected risk of the BSS when the probabilities for attaining the required return are set to 47% and 50%. These differences seem to increase when the level of uncertainty increases (the size of the differences depends on the stock market), and the algorithm is able to report results for relatively big instances in a few seconds.

7. Conclusions

This work addresses a rich, *NP-hard*, and stochastic variant of the portfolio optimization problem (SPOP), in which expected returns and covariances are considered to be random variables instead of constant values as it is standard in the existing literature. The literature review on the SPOP highlights an emerging research topic that mainly relies on uncertainty theory, where only few small illustrative examples are analyzed. Subsequently, the mathematical formulations for both the rich POP and the rich stochastic POP are described. Since real-life financial transactions take place in uncertain environments, the added degree of stochasticity to these elements is conducive to a lower gap between theory and practice. To solve the SPOP, we propose a simheuristic algorithm that combines the VNS metaheuristic (which guides the search of promising solutions) with Monte Carlo simulation techniques. Three computational experiments are performed to illustrate its use, and to analyze how the solutions change in terms of expected risk when the level of stochasticity, the minimum required return, and the probability of obtaining a return greater or equal to the threshold are allowed to vary. Our methodology is able to solve real-sized stochastic instances within relatively short time spans. We also show that, even in an environment with a relatively low level of stochasticity, our simheuristic approach can provide significantly better results than a classical metaheuristic approach that generates solutions for the deterministic POP. It should be noted that solutions to the deterministic and stochastic POPs may become even more contrasting in a dynamic setting, as the horizon of a portfolio investment grows (Laborda and Olmo, 2017). This is an interesting avenue for future research.

The flexibility and robustness of the proposed approach in solving the rich, NP-hard, and stochastic version of the POP render it possible

Table 3
FTSE 100 (UK) with stochastic covariances and correlations.

Required return	Low stochasticity			Medium stochasticity			High stochasticity		
	ER - P: 0.47	ER - P: 0.50	Gap (%)	ER - P: 0.47	ER - P: 0.50	Gap (%)	ER - P: 0.47	ER - P: 0.50	Gap (%)
0.002420865	0.0002401	0.0002419	0.7736	0.0002861	0.0002875	0.5064	0.0004532	0.0004512	0.0000
0.002479328	0.0002401	0.0002418	0.7131	0.0002873	0.0002866	0.0000	0.0004544	0.0004532	0.0000
0.002537791	0.0002393	0.0002407	0.6058	0.0002857	0.0002886	1.0139	0.0004510	0.0004627	2.6053
0.002596256	0.0002415	0.0002433	0.7425	0.0002879	0.0002902	0.8213	0.0004577	0.0004627	1.0768
0.002654720	0.0002414	0.0002428	0.5746	0.0002880	0.0002917	1.2833	0.0004566	0.0004734	3.6744
0.002713184	0.0002431	0.0002434	0.1207	0.0002884	0.0002924	1.3772	0.0004606	0.0004668	1.3512
0.002771647	0.0002436	0.0002559	5.0705	0.0002884	0.0002923	1.3573	0.0004608	0.0004618	0.2030
0.002830111	0.0002433	0.0002452	0.7790	0.0002903	0.0002906	0.0936	0.0004617	0.0004863	5.3253
0.002888575	0.0002449	0.0002586	5.5798	0.0002917	0.0002982	2.2316	0.0004712	0.0004749	0.7892
0.007624416	0.0011858	0.0013801	16.3880	0.0014998	0.0016916	12.7877	0.0026548	0.0030115	13.4355
0.007682888	0.0012358	0.0013552	9.6644	0.0015661	0.0017029	8.7353	0.0027813	0.0031060	11.6753
0.007741359	0.0012849	0.0014719	14.5566	0.0016226	0.0019022	17.2294	0.0028597	0.0033866	18.4247
0.007799830	0.0013401	0.0015654	16.8162	0.0016998	0.0019968	17.4744	0.0030174	0.0035059	16.1884
0.007858284	0.0014014	0.0015511	10.6857	0.0017853	0.0020351	13.9937	0.0031914	0.0036761	15.1881
0.007916738	0.0014689	0.0016034	9.1557	0.0018791	0.0020907	11.2596	0.0033817	0.0036180	6.9885
0.007975191	0.0015425	0.0017087	10.7770	0.0019813	0.0021688	9.4664	0.0035884	0.0038949	8.5417
0.008033645	0.0016223	0.0017271	6.4623	0.0020918	0.0021982	5.0894	0.0038115	0.0040234	5.5595
0.008092098	0.0017083	0.0017802	4.2146	0.0022107	0.0023226	5.0647	0.0040509	0.0042021	3.7322
0.008150551	0.0017988	0.0018133	0.8062	0.0023335	0.0023745	1.7588	0.0042913	0.0043563	1.5146
0.008209000	0.0019060	0.0019062	0.0094	0.0024908	0.0024909	0.0028	0.0046399	0.0046403	0.0081
Average			5.7248			5.5773			5.8140

Table 4
Nikkei 225 (Japan) with stochastic covariances and correlations.

Required return	Low stochasticity			Medium stochasticity			High stochasticity		
	ER - P: 0.47	ER - P: 0.50	Gap (%)	ER - P: 0.47	ER - P: 0.50	Gap (%)	ER - P: 0.47	ER - P: 0.50	Gap (%)
0.000107896	0.0003568	0.0003642	2.0875	0.0004259	0.0004413	3.6120	0.0006879	0.0006933	0.7919
0.000146922	0.0003547	0.0003654	2.9968	0.0004225	0.0004361	3.2060	0.0006642	0.0006954	4.6869
0.000185947	0.0003546	0.0003636	2.5273	0.0004252	0.0004387	3.1852	0.0006764	0.0007014	3.6970
0.000224972	0.0003547	0.0003650	2.9221	0.0004211	0.0004408	4.6879	0.0006713	0.0006909	2.9137
0.000263997	0.0003562	0.0003702	3.9221	0.0004214	0.0004444	5.4518	0.0006832	0.0007067	3.4510
0.000303022	0.0003562	0.0003805	6.8070	0.0004208	0.0004455	5.8677	0.0006677	0.0007202	7.8621
0.000342048	0.0003594	0.0003642	1.3195	0.0004256	0.0004525	6.3148	0.0006722	0.0007242	7.7251
0.000381073	0.0003560	0.0003731	4.8037	0.0004298	0.0004449	3.5173	0.0006682	0.0007009	4.8875
0.000420099	0.0003590	0.0003817	6.3401	0.0004303	0.0004417	2.6532	0.0006743	0.0006977	3.4669
0.003581248	0.0007727	0.0010825	40.1039	0.0009062	0.0012751	40.7079	0.0013976	0.0021721	55.4165
0.003620236	0.0008093	0.0010660	31.7148	0.0009478	0.0012557	32.4840	0.0014572	0.0022281	52.8989
0.003659224	0.0008571	0.0010873	26.8560	0.0010062	0.0014839	47.4716	0.0015542	0.0026588	71.0737
0.003698213	0.0009161	0.0011583	26.4383	0.0010814	0.0014659	35.5562	0.0016884	0.0025299	49.8446
0.003737203	0.0009852	0.0012427	26.1346	0.0011702	0.0016916	44.5603	0.0018499	0.0028002	51.3695
0.003776202	0.0010653	0.0013161	23.5352	0.0012745	0.0016856	32.2487	0.0020437	0.0027783	35.9419
0.003815201	0.0011588	0.0013909	20.0344	0.0013988	0.0016153	15.4792	0.0022815	0.0029845	30.8142
0.003854199	0.0012661	0.0014988	18.3760	0.0015439	0.0018430	19.3746	0.0025616	0.0031692	23.7195
0.003893136	0.0014186	0.0015544	9.5748	0.0017514	0.0020315	15.9934	0.0029694	0.0036564	23.1334
0.003932068	0.0016795	0.0018136	7.9818	0.0021204	0.0022781	7.4372	0.0037337	0.0039758	6.4847
0.003971000	0.0020544	0.0020564	0.0954	0.0026642	0.0026670	0.1067	0.0049044	0.0049060	0.0322
Average			13.2286			16.4958			22.0106

to extend future research in several directions. The first direction could extend our algorithm to a multi-period SPOP, in which the investor would solve a SPOP by optimally rebalancing his or her portfolio of assets. Following Calvet et al. (2017), the second direction could comprise a hybrid approach that combines statistical learning with a metaheuristic-based approach (referred to as *learnheuristics*) to generate high-quality solutions for a dynamic SPOP. A third direction could entail higher order moments of portfolio returns, in addition to the mean and standard deviation. In this regard, Chen et al. (2020) solve a multi-objective optimization problem that entails the third and fourth distributional moments. In Chen et al. (2020) the investor likes (dislikes) higher values of skewness (kurtosis). Hence, he or she maximizes expected portfolio return and skewness, but minimizes portfolio variance and kurtosis. A fourth direction would consider different measures of risk (e.g., value at risk and/or conditional value at risk) and a varying degree of investor risk aversion (Ling et al., 2017), or stochastic dominance constraints (Malavasi et al., 2021).

CRediT authorship contribution statement

Renatas Kizys: Conceptualization, Formal analysis, Investigation, Methodology, Visualization, Writing – original draft, Writing – review & editing. **Jana Doering:** Formal analysis, Investigation, Visualization, Writing – original draft. **Angel A. Juan:** Conceptualization, Formal analysis, Investigation, Methodology, Project administration, Supervision, Resources, Writing – original draft, Writing – review & editing. **Onur Polat:** Formal analysis, Investigation, Visualization, Writing – original draft, Writing – review & editing. **Laura Calvet:** Data curation, Formal analysis, Funding acquisition, Investigation, Software, Visualization, Writing – original draft, Writing – review & editing. **Javier**

Panadero: Data curation, Formal analysis, Investigation, Software, Validation.

Acknowledgments

This work has been partially funded by the Erasmus+ SEPIE program, Spain (2019-I-ES01-KA103-062602).

References

Anil Akbay, M., Kalayci, C.B., Polat, O., 2020. A parallel variable neighborhood search algorithm with quadratic programming for cardinality constrained portfolio optimization. Knowl.-Based Syst. 198, 105944.

Armañanzas, R., Lozano, J.A., 2005. A multiobjective approach to the portfolio optimization problem. In: IEEE Congress on Evolutionary Computation, Vol. 2. IEEE Press, pp. 1388–1395.

Babaei, S., Sepehri, M.M., Babaei, E., 2015. Multi-objective portfolio optimization considering the dependence structure of asset returns. European J. Oper. Res. 244 (2), 525–539.

Bačević, A., Vilimonović, N., Dabić, I., Petrović, J., Damnjanović, D., Džamić, D., 2019. Variable neighborhood search heuristic for nonconvex portfolio optimization. Eng. Econ. 64 (3), 254–274.

Becker, R., Clements, A., Doolan, M., Hurn, A., 2015. Selecting volatility forecasting models for portfolio allocation purposes. Int. J. Forecast. 31 (3), 849–861.

Behr, P., Guettler, A., Miebs, F., 2013. On portfolio optimization: Imposing the right constraints. J. Bank. Financ. 37 (4), 1232–1242.

Bertsimas, D., Pachamanova, D., 2008. Robust multiperiod portfolio management in the presence of transaction costs. Comput. Oper. Res. 35 (1), 3–17.

Bianchi, L., Dorigo, M., Gambardella, L.M., Gutjahr, W.J., 2009. A survey on metaheuristics for stochastic combinatorial optimization. Nat. Comput. 8 (2), 239–287.

Bienstock, D., 1996. Computational study of a family of mixed-integer quadratic programming problems. Math. Program. 74 (2), 121–140.

Bonami, P., Lejeune, M.A., 2009. An exact solution approach for portfolio optimization problems under stochastic and integer constraints. Oper. Res. 57 (3), 650–670.

Boussaïd, I., Lepagnot, J., Siarry, P., 2013. A survey on optimization metaheuristics. Inform. Sci. 237, 82–117.

Calvet, L., de Armas, J., Masip, D., Juan, A.A., 2017. Learnheuristics: hybridizing metaheuristics with machine learning for optimization with dynamic inputs. Open Math. 1 (15), 261–280.

Calvet, L., Kizys, R., Juan, A., de Armas, J., 2016a. A SimILS-based methodology for a portfolio optimization problem with stochastic returns. In: International Conference on Modeling and Simulation in Engineering, Economics and Management. pp. 3–11.

Calvet, L., Kizys, R., Juan, A., de Armas, J., 2016b. An adaptive SA-based simheuristic for the stochastic portfolio optimization problem. Lect. Model. Simul. AMSE 2016, 11–21.

Chang, T.-J., Meade, N., Beasley, J.E., Sharaiha, Y.M., 2000. Heuristics for cardinality constrained portfolio optimisation. Comput. Oper. Res. 27 (13), 1271–1302.

Chen, W., 2015. Artificial bee colony algorithm for constrained possibilistic portfolio optimization problem. Physica A 429, 125–139.

Chen, B., Zhong, J., Chen, Y., 2020. A hybrid approach for portfolio selection with higher-order moments: Empirical evidence from Shanghai Stock Exchange. Expert Syst. Appl. 145, 113104.

Doering, J., Kizys, R., Juan, A.A., Fitó, A., Polat, O., 2019. Metaheuristics for rich portfolio optimisation and risk management: current state and future trends. Oper. Res. Pers. 6, 100121.

Dominguez, O., Guimarans, D., Juan, A.A., de la Nuez, I., 2016. A biased-randomised large neighbourhood search for the two-dimensional vehicle routing problem with backhauls. European J. Oper. Res. 255 (2), 442–462.

Estrada-Moreno, A., Savelsbergh, M., Juan, A.A., Panadero, J., 2019. Biased-randomized iterated local search for a multiperiod vehicle routing problem with price discounts for delivery flexibility. Int. Trans. Oper. Res. 26 (4), 1293–1314.

Fernández, A., Gómez, S., 2007. Portfolio selection using neural networks. Comput. Oper. Res. 34 (4), 1177–1191.

Fernandez-Viagas, V., Framinan, J.M., 2015. NEH-based heuristics for the permutation flowshop scheduling problem to minimise total tardiness. Comput. Oper. Res. 60, 27–36.

Gaspero, L.D., Tollo, G.D., Roli, A., Schaerf, A., 2011. Hybrid metaheuristics for constrained portfolio selection problems. Quant. Finance 11 (10), 1473–1487.

Gonzalez-Neira, E.M., Ferone, D., Hatami, S., Juan, A.A., 2017. A biased-randomized simheuristic for the distributed assembly permutation flowshop problem with stochastic processing times. Simul. Model. Pract. Theory 79, 23–36.

Guimarans, D., Dominguez, O., Panadero, J., Juan, A.A., 2018. A simheuristic approach for the two-dimensional vehicle routing problem with stochastic travel times. Simul. Model. Pract. Theory 89, 1–14.

Hansen, P., Mladenović, N., Pérez, J.A.M., 2010. Variable neighbourhood search: methods and applications. Ann. Oper. Res. 175 (1), 367–407.

Hatami, S., Calvet, L., Fernández-Viagas, V., Framiñán, J.M., Juan, A.A., 2018. A simheuristic algorithm to set up starting times in the stochastic parallel flowshop problem. Simul. Model. Pract. Theory 86, 55–71.

Huang, X., 2007. A new perspective for optimal portfolio selection with random fuzzy returns. Inform. Sci. 177 (23), 5404–5414.

Huang, X., 2012. A risk index model for portfolio selection with returns subject to experts' estimations. Fuzzy Optim. Decis. Mak. 11 (4), 451–463.

Juan, A.A., Corlu, C.G., Tordecilla, R.D., de la Torre, R., Ferrer, A., 2020. On the use of biased-randomized algorithms for solving non-smooth optimization problems. Algorithms 13 (1), 8.

Juan, A.A., Kelton, W.D., Currie, C.S., Faulin, J., 2018. Simheuristics applications: dealing with uncertainty in logistics, transportation, and other supply chain areas. In: Proceedings of the 2018 Winter Simulation Conference. IEEE Press, pp. 3048–3059.

Kalayci, C.B., Ertenlice, O., Akbay, M.A., 2019. A comprehensive review of deterministic models and applications for mean-variance portfolio optimization. Expert Syst. Appl..

Kolm, P.N., Tütüncü, R., Fabozzi, F.J., 2014. 60 Years of portfolio optimization: Practical challenges and current trends. European J. Oper. Res. 234 (2), 356–371.

Laborda, R., Olmo, J., 2017. Optimal asset allocation for strategic investors. Int. J. Forecast. 33 (4), 970–987.

Liagkouras, K., Metaxiotis, K., 2018. Multi-period mean–variance fuzzy portfolio optimization model with transaction costs. Eng. Appl. Artif. Intell. 67, 260–269.

Ling, A., Sun, J., Xiu, N., Yang, X., 2017. Robust two-stage stochastic linear optimization with risk aversion. European J. Oper. Res. 256 (1), 215–229.

Liu, Y., Wu, X., Hao, F., 2012. A new Chance–Variance optimization criterion for portfolio selection in uncertain decision systems. Expert Syst. Appl. 39 (7), 6514–6526.

Lwin, K.T., Qu, R., MacCarthy, B.L., 2017. Mean-VaR portfolio optimization: A nonparametric approach. European J. Oper. Res. 260 (2), 751–766.

Malavasi, M., OrtobelliLozza, S., Trck, S., 2021. Second order of stochastic dominance efficiency vs mean variance efficiency. European J. Oper. Res. 290 (3), 1192–1206.

Mansini, R., Ogryczak, W., Speranza, M.G., 2014. Twenty years of linear programming based portfolio optimization. European J. Oper. Res. 234 (2), 518–535.

Maringer, D.G., 2005. Portfolio Management with Heuristic Optimization. Springer.

Markowitz, H., 1952. Portfolio selection. J. Finance 7 (1), 77–91.

Meghwani, S.S., Thakur, M., 2017. Multi-criteria algorithms for portfolio optimization under practical constraints. Swarm Evol. Comput. 37, 104–125.

Michalewicz, Z., Fogel, D.B., 2013. How to Solve It: Modern Heuristics. Springer Science & Business Media.

Mladenović, N., Hansen, P., 1997. Variable neighborhood search. Comput. Oper. Res. 24 (11), 1097–1100.

Moral-Escudero, R., Ruiz-Torrubiano, R., Suárez, A., 2006. Selection of optimal investment portfolios with cardinality constraints. In: IEEE Congress on Evolutionary Computation. Press IEEE, pp. 2382–2388.

Moreno-Vega, J.M., Melián, B., 2008. Introduction to the special issue on variable neighborhood search. J. Heuristics 14 (5), 403–404.

Muthuraman, K., Zha, H., 2008. Simulation-based portfolio optimization for large portfolios with transaction costs. Math. Finance: Int. J. Math. Stat. Financ. Econ. 18 (1), 115–134.

Nazemi, A., Abbasi, B., Omidi, F., 2015. Solving portfolio selection models with uncertain returns using an artificial neural network scheme. Appl. Intell. 42 (4), 609–621.

Nguyen, T.T., Gordon-Brown, L., Khosravi, A., Creighton, D., Nahavandi, S., 2014. Fuzzy portfolio allocation models through a new risk measure and fuzzy sharpe ratio. IEEE Trans. Fuzzy Syst. 23 (3), 656–676.

Ning, Y., Ke, H., Fu, Z., 2015. Triangular entropy of uncertain variables with application to portfolio selection. Soft Comput. 19 (8), 2203–2209.

Omidi, F., Abbasi, B., Nazemi, A., 2017. An efficient dynamic model for solving a portfolio selection with uncertain chance constraint models. J. Comput. Appl. Math. 319, 43–55.

Onggo, B.S., Panadero, J., Corlu, C.G., Juan, A.A., 2019. Agri-food supply chains with stochastic demands: A multi-period inventory routing problem with perishable products. Simul. Model. Pract. Theory 97, 101970.

Panadero, J., Doering, J., Kizys, R., Juan, A.A., Fito, A., 2020. A variable neighborhood search simheuristic for project portfolio selection under uncertainty. J. Heuristics 26 (3), 353–375.

Qin, Z., 2015. Mean-variance model for portfolio optimization problem in the simultaneous presence of random and uncertain returns. European J. Oper. Res. 245 (2), 480–488.

Qin, Z., 2017. Random fuzzy mean-absolute deviation models for portfolio optimization problem with hybrid uncertainty. Appl. Soft Comput. 56, 597–603.

Rabe, M., Deininger, M., Juan, A.A., 2020. Speeding up computational times in simheuristics combining genetic algorithms with discrete-event simulation. Simul. Model. Pract. Theory 102089.

Ray, J., Bhattacharyya, S., Singh, N.B., 2019. Portfolio optimization and asset allocation with metaheuristics: A review. In: Metaheuristic Approaches to Portfolio Optimization. IGI Global, pp. 1–26.

Saborido, R., Ruiz, A.B., Bermúdez, J.D., Vercher, E., Luque, M., 2016. Evolutionary multi-objective optimization algorithms for fuzzy portfolio selection. Appl. Soft Comput. 39, 48–63.

Schaerf, A., 2002. Local search techniques for constrained portfolio selection problems. Comput. Econ. 20 (3), 177–190.

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

Soler-Dominguez, A., Juan, A.A., Kizys, R., 2017. A survey on financial applications of metaheuristics. ACM Comput. Surv. 50 (1), 1–23.

Talbi, E.-G., 2009. Metaheuristics: From Design to Implementation, Vol. 74. John Wiley & Sons.

Tollo, G.D., Roli, A., 2008. Metaheuristics for the portfolio selection problem. Int. J. Oper. Res. 5 (1), 13–35.

Zhang, W.-G., Liu, Y.-J., Xu, W.-J., 2012. A possibilistic mean-semivariance-entropy model for multi-period portfolio selection with transaction costs. European J. Oper. Res. 222 (2), 341–349.