Proceedings of the Operational Research Society Simulation Workshop 2023 (SW23)
C. Currie and L. Rhodes-Leader, eds.
https://doi.org/10.36819/SW23.020

SIMHEURISTIC AND LEARNHEURISTIC FOR SOLVING STOCHASTIC AND/OR DYNAMIC PORTFOLIO OPTIMIZATION PROBLEMS

Mr. Yuda Li
Dept. of Computer Science, Universitat Oberta de Catalunya
Barcelona, Spain
yli1@uoc.edu

Dr. Onur Polat
Dept. of Applied Statistics and Operations Research, Universitat Politcnica de Valncia
Alcoy, Spain
Dept. of Public Finance, Bilecik Şeyh Edebali University
Bilecik, Turkey
opolat@upvnet.upv.es

Dr. Angel A. Juan
Dept. of Applied Statistics and Operations Research, Universitat Politcnica de Valncia
Alcoy, Spain
juanp@upv.es

Dr. Laura Calvet

Dept. of Telecomm. and Systems Engineering, Autonomous University of Barcelona
Sabadell, Spain
Laura.Calvet.Linan@uab.cat

Dr. Renatas Kizys

Dept. of Banking and Finance, Southampton Business School, University of Southampton
Southampton, United Kingdom
r.kizys@soton.ac.uk

ABSTRACT

Constructing portfolio by proper asset selection to maximize return and minimize risk has been considered an essential task for investment activities. Rich portfolio optimizations with realistic constraints are NP-hard problems and are commonly solved using metaheuristics. However, financial markets are characterized by their high volatility and uncertainty, and metaheuristics do not fully account for these random and/or dynamic components, which renders them unrealistic in the presence of heightened uncertainty and dynamism in financial markets. Therefore, this paper proposes a simulationoptimization approach specifically, a simheuristic algorithm to deal with the stochastic version of the problem and a learnheuristic algorithm for solving the dynamic version of the problem. Computational experiments are performed on a benchmark instance to illustrate the advantages of the proposed methodologies and analyze how the solutions change in response to a different degree of stochasticity, dynamism, and minimum required return.

1 Introduction

Financial decisions have the uppermost significance in the creation of wealth, enhancement of welfare standards, and sustainable economic growth. They play an essential role in providing funds to firms, transforming ideas and resources into profitable projects, and eventually serving social benefits for societies. Such improvements are characterized mainly by the formulation of optimization problems in financial economics.

Developed by Markowitz (1952), the portfolio optimization problem (POP) consists of the investment decision as a strategy of (i) determining financial assets; (ii) computing the appropriate weights allocated to those financial assets in the desired portfolio return, and allocating a minimum level of risk. The POP is carried out through a quadratic objective function that aggregates the weighted covariances of the associated asset returns, which is then minimized subject to the desired portfolio return. Predominantly, the exact methods have been employed to solve this basic version of the POP. However, exact methods become inefficient when dealing with realistic and large-scale combinatorial optimization problems (COPs) due to their NP-hard nature. In this context, metaheuristics have been employed as alternative solution techniques to overcome these drawbacks. Metaheuristics are general solving procedures, that are able to solve near-optimal solutions to COPs in reasonable computing times. Particularly, metaheuristics have been extensively implemented to deal with complex portfolio optimization problems, where pre-assignment, quantity, and cardinality constraints are considered (Armananzas and Lozano 2005, Doering et al. 2019).

Pre-assignment constraints entail the pre-selection of some assets, regardless of their risk-return features. Quantity constraints prescribe the fractions attained to an asset in the portfolio within a floor and ceiling constraint. The quantity constraints render the administration of a portfolio manageable by minimizing lot sizes and lowering transaction costs (Cesarone et al. 2013). Finally, cardinality constraints dictate a lower and upper bound for the number of assets covered in the portfolio. Cardinality constraints not only limit the number of assets in the portfolio to cope with but provide a certain threshold of diversification (Gaspero et al. 2011).

In contrast to the real-world trading constraints, a plethora of studies assumes constant covariances. This work presents two more realistic versions of the POP: the stochastic POP (SPOP) and the dynamic POP (DPOP). In the SPOP the covariances are modeled as random variables, which may be estimated by historical data using certain statistical measures. Therefore, the predictions encompass a certain threshold of uncertainty (noise). In the DPOP, the covariances matrix is deterministic but non-static (i.e., while we know the estimated value for the covariance of each pair of assets, the specific value that this covariance takes is influenced by synergy effects of assets in the portfolio). We deal with the SPOP by implementing the simheuristic algorithm proposed by Kizys et al. (2022). Furthermore, we propose a novel learnheuristic algorithm, which is an integration of a machine learning algorithm with a well-known metaheuristic, the Variable Neighborhood Search (VNS) (Hansen et al. 2019), to address the DPOP. Both methodologies carry considerable benefits compared with the presumption of deterministic/static risk. A set of computational experiments is carried out to illustrate and validate the solving approaches.

We proceed with the study as follows: Section 2 reviews related studies in the field. Section 3 presents the mathematical model and the problem definition for the POP. Section 4 describes the SPOP and the simheuristic solving approach, provides computational experiments, and analyses the results. Section 5 provides details for the learnheuristic algorithm and analyzes the results of computational experiments. Finally, Section 7 draws the main findings of the work and discusses future research lines.

2 Literature Review

Seven decades ago, Markowitz (1952) proposed the first mathematical formalization of the idea of diversification of investments. This work postulated that an investor should maximize expected portfolio return while minimizing portfolio variance of return. Since then, multiple authors have contributed to the field of portfolio optimization by considering different risk measures, realistic constraints, and datasets, and solving methodologies (ranging from exact methods for small problem instances to approximate methods for bigger and more realistic problem instances) (Kolm et al. 2014). Portfolio optimization continues to be an interesting and challenging task that attracts the attention of researchers. The reader interested in recent reviews is referred to Milhomem and Dantas (2020), Kalayci et al. (2019), and Doering et al. (2019).

For instance, recently, Kizys et al. (2022) present a simheuristic algorithm (Juan et al. 2015) that integrates a variable neighborhood search metaheuristic with Monte Carlo simulation to address the portfolio optimization problem with pre-selection, quantity and cardinality constraints, as well as stochastic returns and noisy covariances modeled as random variables. Zhai et al. (2020) build a multiconstraint portfolio optimization model based on the second-order stochastic dominance rule, the higher moments of return series, and the Shannon entropy, in addition to other investment constraints. The authors employ the whale optimization algorithm to solve the problem, which is compared against other swarm intelligence optimization algorithms. The numerical results used data from the FTSE100 index stocks during 2018. Sehgal and Mehra (2020) proposed a model for robust portfolio optimization with second order stochastic dominance constraints, in which the input returns of each asset at every scenario is varied in a bounded and symmetric interval. The cutting plane algorithm is employed to deal with the optimization problem. The performance of the model is assessed by experimenting with datasets drawn from S&P 500, S&P BSE 500, Nikkei 225, S&P Global 100, FTSE 100, and BOVESPA index.

3 Problem Definition

This section provides a mathematical formulation of the constrained POP. It includes a description of the notation employed and the mathematical formulations, which are based on Kizys et al. (2022).

3.1 Formal Description

In the classical POP, there is a set $A = \{a_1, a_2, \ldots, a_n\}$ of n assets. Each asset a_i ($\forall i \in \{1, 2, \ldots, n\}$) is characterized by an expected return r_i. The covariance between two assets a_i and a_j ($\forall i, j \in \{1, 2, \ldots, n\}$) is represented by σ_{ij}.

A solution for the POP can be encoded as a vector $X = (x_1, x_2, \ldots, x_n)$, where each element $x_i (0 \leq x_i \leq 1)$ reveals the weight or fraction of the investment allocated to the asset a_i.

The main goal is to minimize the portfolio risk, while meeting the following constraint: the expected return has to be greater than an investor-given threshold R. The constrained version of the POP describe a more realistic scenario by including pre-selection, quantity, and cardinality constraints. Pre-selection constraints reveal whether an asset a_i has been pre-selected by the investor and, therefore, it has to be included in the solution ($x_i > 0$). They are described by means of the parameter p_i: $p_i = 1$ if a_i is pre-selected, and $p_i = 0$ otherwise. Quantity constraints set a lower and an upper bounds for the weights x_i, ε_i and δ_i ($0 \leq \varepsilon_i \leq \delta_i \leq 1$), respectively. Finally, the cardinality constraint specifies the minimum and maximum number of assets included in the solution, k_{min} and k_{max} ($1 \leq k_{min} \leq k_{max} \leq n$).

3.2 Mathematical Model

The constrained 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:

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

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

$$\varepsilon_i z_i \leq x_i \leq \delta_i z_i, \forall i \in \{1, 2, \ldots, n\} \tag{4}$$

$$0 \leq \varepsilon_i \leq \delta_i \leq 1, \forall i \in \{1, 2, \ldots, n\} \tag{5}$$

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

$$p_i \leq z_i, \forall i \in \{1, 2, \ldots, n\} \tag{7}$$

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

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

Equation (1) represents the objective function, which represents the riskiness of the portfolio and is to be minimized. Equation (2) forces that the expected return of the investment is equal or greater than the threshold R. Equation (3) guarantees that the portfolio investment equals existing and pre-defined resources. For each asset a_i, Equation (4) sets lower and upper bounds (ε_i and δ_i, respectively) for x_i in case the asset is selected. Whether the asset a_i is included in the solution or not is represented by means of an auxiliary variable ($z_i = 1$ if positive; $z_i = 0$ otherwise). Equation (5) bounds the lower and upper bounds by zero and one inclusive. Equation (6) relates variables x_i with z_i, ensuring that if x_i is greater than 0, then z_i is 1 (M is a large positive value such that $Mx_i \geq 1$ for all i if $x_i > 0$). The pre-assignment constraint is defined by Equation (7). It relates variable z_i with parameter p_i; if the asset a_i is pre-selected (i.e., $p_i = 1$), it must be included in the portfolio (i.e., $z_i = 1$). Equation (8) sets the minimum and maximum number of assets to be included in the portfolio. Finally, Equation (9) defines z_i as a binary variable.

4 Stochastic Portfolio Optimization Problem

In this paper, the difference between the POP and the SPOP resides the modeling of asset covariances. In the POP, they are represented by the expected values. However, the SPOP proposes a realistic stochastic uncertainty and hence suggests these as random variables. Therefore, in the second case, covariances (C_{ij}) in the objective function are considered random variables that follow a given probability distribution:

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

here Γ represents a specific statistical measure (such as the mean, or the variance).

4.1 Simheuristics

Simheuristics are combinations of metaheuristics with simulation and have been employed extensively by scholars to deal with NP-hard stochastic optimization problems. In this vein, studies on vehicle routing problems with stochastic times (Taş et al. 2013, Wang and Lin 2013, Guimarans et al. 2018), flow-shop scheduling problems with random processing times (Juan et al. 2014, Wu et al. 2018, Lee and Kim 2022), and inventory routing problems with stochastic demands (Gruler et al. 2018, Raba et al. 2020) can be mentioned.

In this work, we employ the simheuristic algorithm proposed by Kizys et al. (2022), which integrates a VNS with Monte Carlo simulation (MCS). The number of neighbors K is set to 3, and a movement in each neighbor constitutes changing 25%, 35%, and 45% of the assets, respectively. The algorithm is described as follows:

1. First, an initial solution (*initSol*) is constructed by selecting the k_{min} assets that provide the highest rate of return, including the s assets pre-selected by the investor, then, the optimal weights allocated to each asset is determined by a quadratic programming solver.
2. A list *bestSols* is developed for storing the l best-found solutions regarding the expected risk. Thereafter, *initSol* is copied into *currentSol* and k is set to one. In the next step, the expected risk of *currentSol* is derived in terms of MCS, and the solution is stored in the *bestSols* list.
3. A new solution (*newSol*) is constructed by 'shaking' the *currentSol*. This procedure randomly erases a number of non-pre-selected assets in the solution and randomly introduces new assets until reaches k_{max}. Furthermore, a local search is employed for the solution. *newSol* is compared against *currentSol*. If the former is better concerning the risk corresponding to the deterministic version of the problem, the expected risk for the stochastic version is computed with sim_{short} runs. If the expected risk of *newSol* is lower than that of *currentSol*, then *newSol* replaces

currentSol, k is set to one, and *bestSols* is updated. If not, the solution is removed. If *newSol* is not better, k is increased in one unit if $k < K$ or set to one otherwise. This step is repeated until the maximum time constraint T_{loop} is reached.

4. Finally, for each solution in *bestSols*, a larger number of runs (sim_{large}) is simulated to obtain a more accurate estimation of expected risk.

5 Dynamic Portfolio Optimization Problem

In this section, we consider a dynamic version of the POP in which the covariances matrix is deterministic but non-static, i.e.: while we know the estimated value for the covariance of each pair of assets, the specific value that this covariance takes is influenced by the synergy effects of assets s_i and s_j ($\forall i, j \in \{1, 2, \ldots, n\}$) in the portfolio, i.e., some combinations of assets might increase or decrease the initial covariance estimates. Therefore the original deterministic objective function is transformed into:

$$\min\ f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \sigma_{ij} x_i x_j s_i s_j \tag{11}$$

5.1 Learnheuristic

To deal with this dynamic environment, the paper employs a learnheuristic algorithm, which is a hybrid algorithm that combines metaheuristic algorithms with machine learning to deal with optimization problems with dynamic inputs (Calvet et al. 2017). In these optimization problems, the inputs (either located in the objective function or in the set of constraints) are not fixed in advance and they vary in a predictable way according to the current status of the solution. Therefore, learnheuristic relies on machine learning techniques to learn the relationships between inputs and solution characteristics from historical data and metaheuristic algorithms to find high-quality solutions using the predicted inputs. In this sense, these optimization problems represent an extension of the classical static optimization problems, in which all inputs are given in advance and are immutable. Learnheuristics have been employed in different application areas. Calvet et al. (2016) used learnheuristic to solve a multi-depot vehicle routing problem with the assumption that customers show a different willingness to consume depending on how well the assigned depot fits their preferences. To solve this problem, they employed machine learning models trained with historical data to estimate the demand of the customers depending on the assigned depot and included this prediction as part of an enriched objective function as a way to better guide the stochastic local search inside the metaheuristic framework. Arnau et al. (2018) employed learnheuristic for solving a vehicle routing problem with dynamic traveling times which depend on the structure of the routing plan. Bayliss et al. (2020) proposed a learnheuristic algorithm to solve an aerial-drone team orienteering problem with the travel times between targets depending on drones flight path between previous targets.

Pseudocode 1 illustrates a simple description of the basic learnheuristic framework for solving our DPOP. First, a white box learner is generated using a machine learning model trained with historical data, and an initial dummy solution is saved as the current best solution. Then, at each iteration, the VNS algorithm described in Subsection 4.1 is used to select the assets and construct the portfolio. Once the portfolio is constructed, the white box learner is employed to predict the synergy effect of assets in the portfolio, and the optimal weight of assets is allocated according to the objective function (Equation 11) using the predicted synergy coefficients. This iteration is repeated until the maximum time constraint T_{loop} is reached. Finally, the best solution obtained is modeled in the black box simulator, and the actual risk is obtained to compare with the white box predicted risk.

6 Computational Experiments

6.1 DPOP

The proposed algorithm was executed as a Python application. We used a standard personal computer, Intel Core i7 CPU at 2.5 GHz and 12GB RAM with Windows 10 to perform all tests. The same dataset and metaheuristic described in the previous section are used to conduct the experiments. In order to evaluate our learnheuristic approach, the set of asset combinations that can trigger synergies are created, the ID of these assets are: (4, 8), (21, 22, 30), (14, 15), (19, 20, 21), and (27, 29), with their respective

Algorithm 1 Basic structure of the learnheuristic.

learnheuristic(*instance*, *T*)
t ← 0
WhiteBoxPredictor ← getMLPredictor(instance)
bestSol ← dummySolution(instance)
while {t < T} **do**
    portfolio ← constructPortfolio(instance)
    newSol ← allocateWeight(portfolio, WhiteBoxPredictor)
    if cost(newSol) ≤ cost(bestSol)} **then**
        bestSol ← newSol
    end if
    t ← elapsedTime
end while
blackBox(bestSol)
return bestSol

synergies coefficients 0.9, 1.2, 0.8, 0.85, and 1.15. The dataset is tested under 3 different levels of dynamism (i.e. the original coefficient is multiplied by 0.8, 1.0, and 1.2, respectively). The maximum time constraint T_{loop} is set to 15. The white box learner of learnheuristic can be any machine learning model, in this paper we employed a decision tree regressor model. Table 1 summarizes the results of our experiment. The first column displays the required return, showing only the first and the last 10 values. Columns 2, 3, and 4 are the final risk of our learnheuristic approach under the low, medium, and high levels of dynamism. Columns 5, 6, and 7 detail the gaps between the final risk of learnheuristic approach and the final risk of the static approach under the 3 different levels of dynamism. The final risk of the dynamic solutions is subtracted from that of the static solutions, therefore, a negative gap represents an improvement in the final risk of the solution. These results are also presented as the right sub-plot of Figure 1.

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

Required Return	Risk			Gaps (%)		
	Low	Medium	High	Low	Medium	High
0.002861137	0.0006111	0.0005727	0.0005463	-7.20%	-14.20%	-19.55%
0.002941981	0.0006204	0.0005774	0.0005714	-7.29%	-31.16%	-16.43%
0.003022826	0.000613	0.0005847	0.0005445	-10.77%	-29.66%	-22.94%
0.003103671	0.0006031	0.0005802	0.0005459	-19.60%	-28.87%	-36.87%
0.003184516	0.000621	0.0005766	0.0005514	-19.87%	-26.90%	-24.65%
0.003265361	0.0006217	0.0005909	0.000564	-15.22%	-12.27%	-33.24%
0.003346206	0.0006093	0.0005786	0.0005557	-18.84%	-28.43%	-23.39%
0.003427051	0.0006004	0.0005917	0.0005655	-24.65%	-23.34%	-35.47%
0.003507896	0.0006208	0.0005971	0.0005559	-9.26%	-2.45%	-38.05%
0.010056641	0.0029361	0.0028098	0.0026863	0.00%	0.00%	0.00%
0.010137479	0.0030279	0.0028977	0.0027703	0.00%	0.00%	0.00%
0.010218315	0.0031239	0.0029895	0.0028581	0.00%	0.00%	0.00%
0.01029915	0.003224	0.0030854	0.0029498	0.00%	0.00%	0.00%
0.010379986	0.0033283	0.0031852	0.0030452	-20.94%	-26.38%	-32.19%
0.010460822	0.0034368	0.003289	0.0031445	0.00%	0.00%	0.00%
0.010541657	0.0035495	0.0033969	0.0032476	0.00%	0.00%	0.00%
0.010622493	0.0036664	0.0035087	0.0033545	0.00%	0.00%	0.00%
0.010703329	0.0037874	0.0036245	0.0034652	0.00%	0.00%	0.00%
0.010784164	0.0039126	0.0037443	0.0035798	0.00%	0.00%	0.00%
0.010865	0.0047755	0.0047755	0.0047755	0.00%	0.00%	0.00%
			Average:	-7.68%	-11.18%	-14.14%

As we can observe, the final risk of the dynamic solutions is always lower than the final risk of static solutions, the average gap of dynamic solutions is -7.68%, -11.18% -14.14% for low, medium, and high levels of dynamism, respectively. The performance of the static solutions has shown a clearly decreasing trend as the synergy effect of assets increases. Furthermore, the gap of solutions is higher when the required return is relatively lower, this is not surprising as a lower return would allow more choice of assets and a more diversified portfolio, leading to a higher probability of creating synergy effects. Thus, the learnheuristic approach carries considerable benefits compared with the assumption of static risk.

6.2 SPOP

The proposed algorithm was executed as a JAVA application. We run our algorithm for the experimental stock market data from the repository ORlib (http://people.brunel.ac.uk/~mastjjb/jeb/orlib/portinfo.html). These datasets were gathered from Datastream and cover global financial and macroeconomic data. In this work, we used the Hang Seng (Hong Kong) stock market index at a weekly frequency between March 1992 and September 1997. Missing stock data were discarded. The data constitutes the mean and the standard deviation of returns from the stock market, and the correlation coefficients for all possible pairs of assets. We follow Gaspero et al. (2011) and divide the portfolio frontier into 100 equidistant points on the axis indicating the portfolio expected return.

This benchmark dataset is deterministic. In order to perform our simheuristic algorithm, we considered the following modifications and assumptions.

1. Standard deviation, S_i follows a $LN(\mu_S, \sigma_S)$, where LN denotes 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. We assume that they take the values σ_i and $c\sigma_i$, respectively, where σ_i represents the standard deviation of the variable, and c is an input.
2. Correlation C_{ij} follows a truncated normal distribution $TN(\mu_C, \sigma_C, lb, ub)$, where μ_C is the mean, σ_C is the standard deviation, and lb and ub are the lower and upper bounds, respectively. μ_C is the correlation between the returns on assets a_i and a_j, ρ_{ij}, and σ_C is an input. We set lb and ub to -1 and 1, respectively, to ensure that the correlation varies between -1 and 1.

$c(0.01, 0.025, 0.08)$ and $\sigma_M(\sqrt{0.00002}, \sqrt{0.0002}, \sqrt{0.002})$ were used to determine three different levels of stochasticity, from lowest to highest. Following Kizys et al. (2022), β is randomly selected from a uniform distribution with parameters 0.05 and 0.25. Furthermore, sim_{short}, sim_{large}, and T_{loop} are set to 2500, 12500, and 15, respectively.

We assume stochastic covariances and plot the relationship between the required return and the expected risk for three different levels of stochasticity (low, medium, and high) for Hang Seng stock market data as the left sub-plot of Figure 1. In this figure, we only consider the first and the last 10 values of required returns and compare the best-found solutions obtained for the deterministic/static and stochastic/dynamic environments.

Unsurprisingly, the higher the returns, the higher the risks (level of stochasticity). Moreover, negative gaps between stochastic and deterministic solutions reveal that the simheuristic approach provides considerable benefits compared with the assumption of constant expected risk. Furthermore, the performance of the BSS ameliorates when covariances are more uncertain.

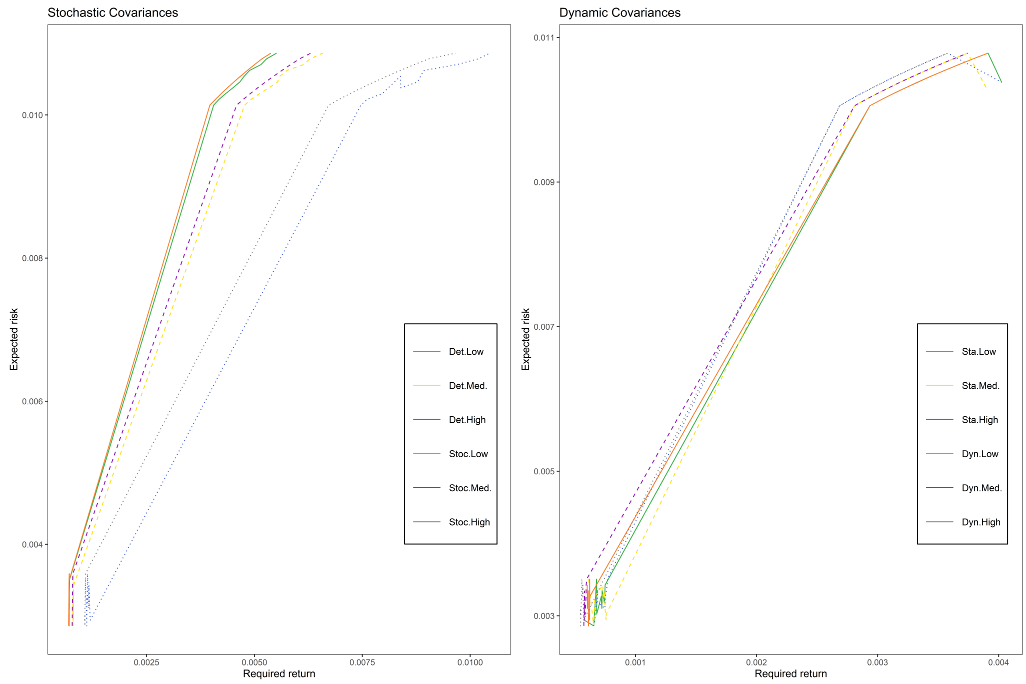

Figure 1: Risk curves for Hang Seng stock market with stochastic, dynamic covariances.

7 Conclusions and Future Research

While the literature on rich POP is extensive, most of them only considers deterministic versions of POP. However, real-life POP faced by investors is becoming increasingly complex. This can be due, among other factors, to the increasingly complex financial markets full of uncertainty and dynamism. On the other hand, advances in computing power over the last few decades have enabled the implementation of more powerful and faster algorithms, as well as the analysis of massive amounts of various types of data. As a consequence, hybrid approaches for addressing hard combinatorial and/or continuous optimization problems are becoming more popular. Therefore, this paper proposed two hybrid approaches for solving SPOP and DPOP, respectively. The first approach proposed is a simheuristic algorithm, a combination of metaheuristics with simulation for solving SPOP with stochastic covariances. Computational experiments showed considerable benefits compared to the traditional deterministic approach. The second approach is a learnheuristic algorithm, a combination of metaheuristics and machine learning for solving DPOP with dynamic covariances. Computational experiments showed that learnheuristic approach can provide significantly better results than the static metaheuristic approach.

In this paper, SPOP and DPOP are solved separately. However, in real-life scenarios, stochastic and dynamic components may coexist in the environment. Therefore, it is necessary to solve the problem with stochastic and dynamic components together. In future work, we plan to combine simheuristic and learnheuristic, i.e., hybridizing machine learning-simulation-optimization for solving optimization problems with both stochastic and dynamic components.

Acknowledgements

This work has been supported by the TÜBTAK 2219-A program, (1059B192200770).

REFERENCES

Armananzas, R., and J. A. Lozano. 2005. "A multiobjective approach to the portfolio optimization problem". In *IEEE Congress on Evolutionary Computation*, Volume 2, 1388–1395. IEEE.

Arnau, Q., A. A. Juan, and I. Serra. 2018. "On the use of learnheuristics in vehicle routing optimization problems with dynamic inputs". *Algorithms* 11 (12): 208.

Bayliss, C., A. A. Juan, C. S. Currie, and J. Panadero. 2020. "A learnheuristic approach for the team orienteering problem with aerial drone motion constraints". *Applied Soft Computing* 92:106280.

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

Calvet, L., A. Ferrer, M. I. Gomes, A. A. Juan, and D. Masip. 2016. "Combining statistical learning with metaheuristics for the multi-depot vehicle routing problem with market segmentation". *Computers & Industrial Engineering* 94:93–104.

Cesarone, F., A. Scozzari, and F. Tardella. 2013. "A new method for mean-variance portfolio optimization with cardinality constraints". *Annals of Operations Research* 205 (1): 213–234.

Doering, J., R. Kizys, A. Juan, A. Fit, and O. Polat. 2019. "Metaheuristics for rich portfolio optimisation and risk management: Current state and future trends". *Operations Research Perspectives* 6:100121.

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

Gruler, A., J. Panadero, J. de Armas, J. A. M. Pérez, and A. A. Juan. 2018. "Combining variable neighborhood search with simulation for the inventory routing problem with stochastic demands and stock-outs". *Computers & Industrial Engineering* 123:278–288.

Guimarans, D., O. Dominguez, J. Panadero, and A. A. Juan. 2018. "A simheuristic approach for the two-dimensional vehicle routing problem with stochastic travel times". *Simulation Modelling Practice and Theory* 89:1–14.

Hansen, P., N. Mladenović, J. Brimberg, and J. A. M. Pérez. 2019. "Variable neighborhood search". In *Handbook of metaheuristics*, 57–97. Springer.

Juan, A. A., B. B. Barrios, E. Vallada, D. Riera, and J. Jorba. 2014. "A simheuristic algorithm for solving the permutation flow shop problem with stochastic processing times". *Simulation Modelling Practice and Theory* 46:101–117.

Juan, A. A., J. Faulin, S. E. Grasman, M. Rabe, and G. Figueira. 2015. "A review of simheuristics: Extending metaheuristics to deal with stochastic combinatorial optimization problems". *Operations Research Perspectives* 2:62–72.

Kalayci, C. B., O. Ertenlice, and M. A. Akbay. 2019. "A comprehensive review of deterministic models and applications for mean-variance portfolio optimization". *Expert Systems with Applications*.

Kizys, R., J. Doering, A. A. Juan, O. Polat, L. Calvet, and J. Panadero. 2022. "A simheuristic algorithm for the portfolio optimization problem with random returns and noisy covariances". *Computers & Operations Research* 139:105631.

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

Lee, J.-H., and H.-J. Kim. 2022. "Reinforcement learning for robotic flow shop scheduling with processing time variations". *International Journal of Production Research* 60 (7): 2346–2368.

Markowitz, H. 1952. "Portfolio selection". *Journal of Finance* 7 (1): 77–91.

Milhomem, D. A., and M. J. P. Dantas. 2020. "Analysis of new approaches used in portfolio optimization: A systematic literature review". *Production* 30.

Raba, D., A. Estrada-Moreno, J. Panadero, and A. A. Juan. 2020. "A reactive simheuristic using online data for a real-life inventory routing problem with stochastic demands". *International Transactions in Operational Research* 27 (6): 2785–2816.

Sehgal, R., and A. Mehra. 2020. "Robust portfolio optimization with second order stochastic dominance constraints". *Computers & Industrial Engineering* 144:106396.

Taş, D., N. Dellaert, T. Van Woensel, and T. De Kok. 2013. "Vehicle routing problem with stochastic travel times including soft time windows and service costs". *Computers & Operations Research* 40 (1): 214–224.

Wang, Z., and L. Lin. 2013. "A simulation-based algorithm for the capacitated vehicle routing problem with stochastic travel times". *Journal of Applied Mathematics* 2013.

Wu, X., X. Shen, and Q. Cui. 2018. "Multi-objective flexible flow shop scheduling problem considering variable processing time due to renewable energy". *Sustainability* 10 (3): 841.

Zhai, Q., T. Ye, M. Huang, S. Feng, and H. Li. 2020. "Whale optimization algorithm for multiconstraint second-order stochastic dominance portfolio optimization". *Computational Intelligence and Neuroscience* 2020.

AUTHOR BIOGRAPHIES

YUDA LI is a predoctoral researcher at the ICSO research group at Universitat Oberta de Catalunya (Spain). He holds a BSc in Aeronautical Management from the Universitat Autonoma de Barcelona and a Msc in Computational Engineering and Mathematics from the Universitat Rovira i Virgili. His email address is yli1@uoc.edu.

ONUR POLAT is an Associate Professor in Public Finance at Bilecik University. He is also a postdoctoral research fellow at the Dept. of Applied Statistics & Operations Research at the Universitat Politcnica de Valncia. He received an MSc in Financial Mathematics from the Middle East Technical University in 2009. He completed his PhD in Economics at Hacettepe University in 2017. He held a visiting scholar position at Illinois State University. His website is https://sites.google.com/view/onurpolat/opolat

LAURA CALVET received a BSc on Applied Statistics from Universitat Autnoma de Barcelona (UAB) in 2012. She completed her PhD at the Open University of Catalonia in 2017. She is currently a lecturer of Supply Chain Management in the Department of Telecommunications and Systems Engineering at the UAB. Her website is: https://sites.google.com/view/lauracalvetlinan.

RENATAS KIZYS is an Associate Professor in Finance within Southampton Business School at the University of Southampton. He received BA and MA degrees in Economics from the University of Vilnius, Lithuania, and MSc and PhD in Economics from the University of Alicante, Spain. He also graduated from the Advanced Studies Program in International Economic Policy Research at the Kiel Institute for the World Economy, Germany. Prior to joining the University of Southampton, Renatas Kizys spent 8 years working at the University of Portsmouth, where he held various positions, ranging from Lecturer to Reader. He spent more than three years working as an Assistant Professor at the Technological Institute of Monterrey, Mexico. He held a P.K. Woolley Research Fellowship within the Department of Economics and Related Studies at University of York, in the UK. He has also held Visiting Academic positions in China, Germany, Lithuania, Indonesia and Spain. His institutional website address is https://www.southampton.ac.uk/people/5xmnbw/doctor-renatas-kizys and his email address is r.kizys@soton.ac.uk

ANGEL A. JUAN is a Full Professor in the Dept. of Applied Statistics & Operations Research at the Universitat Politcnica de Valncia (Spain). Dr. Juan holds a Ph.D. in Industrial Engineering and an M.Sc. in Mathematics. He completed a predoctoral internship at Harvard University and postdoctoral internships at the Massachusetts Institute of Technology and the Georgia Institute of Technology. His main research interests include applications of simheuristics and learnheuristics in computational logistics, transportation, and finance. He has published more than 130 articles in JCR-indexed journals and over 300 papers indexed in Scopus. His website address is http://ajuanp.wordpress.com and his email address is ajuanp@upv.es