A Variable Neighborhood Search Simheuristic for Project Portfolio Selection under Uncertainty

Javier Panadero[1] · **Jana Doering**[2] ·
Renatas Kizys[3] · **Angel A. Juan**[1] ·
Angels Fito[2]

Abstract With limited financial resources, decision-makers in firms and governments face the task of selecting the best portfolio of projects to invest in. As the pool of project proposals increases and more realistic constraints are considered, the problem becomes NP-hard. Thus, metaheuristics have been employed for solving large instances of the project portfolio selection problem (PPSP). However, most of the existing works do not account for uncertainty. This paper contributes to close this gap by analyzing a stochastic version of the PPSP: the goal is to maximize the expected net present value of the inversion, while considering random cash flows and discount rates in future periods, as well as a rich set of constraints including the maximum risk allowed. To solve this stochastic PPSP, a simulation-optimization algorithm is introduced. Our approach integrates a variable neighborhood search metaheuristic with Monte Carlo simulation. A series of computational experiments contribute to validate our approach and illustrate how the solutions vary as the level of uncertainty increases.

[1]
Open University of Catalonia
IN3 - Computer Science Dept.
Barcelona, Spain
E-mail: {jpanaderom, ajuanp}@uoc.edu

[2]
Universitat Oberta de Catalunya
Economics and Business Dept.
Barcelona, Spain
E-mail: {jdoering, afitob}@uoc.edu

[3]
Economics and Finance Subject Group
Portsmouth Business School
University of Portsmouth
Portsmouth, UK.
E-mail: renatas.kizys@port.ac.uk

Keywords project portfolio selection · stochastic optimization · net present value · variable neighborhood search · simheuristics

1 Introduction

Financial decisions are directly linked to wealth creation through capital accumulation, sustainable economic development, and an increase in welfare (Patrick, 1966). This thriving for improvement suggests a mentality of optimization in financial decision-making. Both firms and governments alike face investment decisions consisting in selecting, from an array of candidate projects, those that most successfully fulfill the organizations strategic objectives and ensure future profitable growth. This project portfolio selection problem (PPSP) is concerned with identifying efficient portfolios of projects instead of evaluating the suitability of solely individual projects (Urli and Terrien, 2010).

Traditional approaches to the PPSP aim at building a ranking of the projects and allocating the available budget according to this ranking. Among the most widely employed are the analytical hierarchy process (Suh et al., 1994) and the scoring method (Coldrick et al., 2005). However, these approaches suffer from two major shortcomings (Carazo et al., 2010): *(i)* they typically assume independence among projects –thus neglecting synergy and cannibalism effects as well as interdependences–; and *(ii)* they fail to provide optimal solutions when the decision-maker wishes to consider further constraints beyond budget restrictions.

When considering realistic instances, this problem usually becomes NP-hard due to its sheer complexity, since the budget-allocating entity usually pursues several conflicting objectives while taking into account a considerable number of restraining factors (Fernandez et al., 2015). As noticed by Urli and Terrien (2010), objectives can be of quantitative nature –such as net present value or market share–, or pertain to qualitative measures –such as personnel capabilities or environmental impact. While employing exact methods in solving NP-hard combinatorial optimization problems (COPs) tends to be computationally expensive, metaheuristics can provide a near-optimal solution to such problems in reasonable computing times (Soler et al., 2017).

In this paper, we analyze a stochastic version of the PPSP: the goal is to maximize the expected net present value of the inversion, while considering random cash flows and discount rates in future periods as well as a rich set of constraints. These constraints include the maximum risk allowed and other conditions defined by the decision maker. In order to solve this version of the problem, we propose a simheuristic algorithm. As described in Juan et al. (2015a), simheuristic approaches integrate metaheuristics with simulation techniques in order to deal with the random nature of stochastic COPs. In particular, we use an extension of the variable neighbourhood search (VNS) metaheuristic (Hansen and Mladenović, 2001) that integrates Monte Carlo simulation (MCS) techniques. In short, while the metaheuristic generates promising portfolios for a deterministic version of the problem, simulation techniques are applied to: *(i)* estimate the expected net present value and risk of these project portfolios under time-variant uncertainty conditions; *(ii)* complete a risk analysis on each project portfolio; and *(iii)* provide feedback to the metaheuristic in order to better guide the searching process. The VNS framework was used since, as discussed in Hansen and Mladenović (2014), it offers an excellent trade-off between simplicity and performance.

Thus, the main contributions of this paper are: *(i)* to propose a mathematical formulation for a rich version of the stochastic PPSP where the goal is to maximize the net present value of the investment; *(ii)* to develop a simheuristic algorithm able to solve this stochastic version of the PPSP; and (*iii)* to analyze, using the aforementioned algorithm, how the selected portfolio of projects vary as the uncertainty level increases.

We solve both deterministic and stochastic PPSPs, and compare the near-optimal solutions. The deterministic PPSP indicates that portfolios consisting of risky projects have a higher NPV than portfolios consisting of relatively safe projects. However, it is worth noting that such a relation is not necessarily linear due to the the presence of cardinality and quantity constraints. Also, the instances employed in our experiments vary in terms of the pairwise correlation between cashflows from any two projects. The ensuing interdependencies among the projects can be regarded as a constraint to the volume of projects that can be included in a portfolio. Turning to the the stochastic PPSP, we find that a portfolio of projects in a stochastic environment always yields a lower (expected) NPV than a portfolio in a deterministic environment. Furthermore, our research findings indicate that a near-optimal solution to the deterministic PPSP is generally sub-optimal under uncertainty. Lastly, a near-optimal solution to the stochastic PPSP leads to a higher (expected) NPV than a near-optimal solution to the deterministic PPSP evaluated under uncertainty.

The remainder of the paper is structured as follows. Section 2 presents a literature review. Section 3 contains the description of the problem as well as a mathematical formulation. We propose our solving methodology in Section 4. Following this, the computational experiments are presented in Section 5 and their results analyzed in Section 6. Lastly, we present our conclusions and future research lines in Section 7.

2 Literature Review on the PPSP

Early work on the PPSP (Ghasemzadeh and Archer, 2000) considers a single weighted objective function and constraints concerning budget and man-hours. However, their test instances were very limited because they aspired a comparison between manually computed portfolios and those constructed employing their decision support system. To solve a PPSP employing further constraints, a two-stage procedure is proposed by Doerner et al. (2004) and Doerner et al. (2006). During the first phase, the Pareto frontier of efficient project portfolios is constructed through optimization. Then, in the second phase it is interactively explored by the decision-makers to account for personal preferences. They further take into account floor and ceiling constraints for inclusion of projects from any given subset, as well as resource limitations and minimum benefit requirements for individual projects. As there are possible synergies between projects that should be evaluated in order to accurately estimate the benefits of a project portfolio, the authors make an attempt at incorporating these considerations into their methodology. The Pareto ant colony optimization approach is further enhanced by Stummer and Sun (2005), who suggest that their improved model performs better with many objective functions and a large set of efficient solutions and is thus specifically suitable for real-life problems.

These interdependences show that the portfolio optimization is not a trivial task as the number of possible portfolios increases exponentially with the number of possible projects. Thus, Urli and Terrien (2010) consider project interdependences modeled by an interaction matrix as proposed by Schmidt (1993) in addition to restrictions on monetary and human resources. Rabbani et al. (2010) further consider that some projects may be mandatory or mutually exclusive. Furthermore, project interaction leads to the consideration of timing of project implementation.

While previous research considers static optimization approaches, more recently, research has also drawn on findings from other areas, such as scheduling: Gutjahr et al. (2008) and Gutjahr et al. (2010) also take employee competencies and the evolution of their knowledge scores over time through learning or depreciation into account. Carazo et al. (2010) further investigate this research line and include scheduling as a continuative concept being implemented simultaneously to, but also following the project selection. As previous work, they also consider certain interrelations between different projects and allow for the transfer of unused monetary resources to the next period. Urli and Terrien (2010) included continuous project portfolio adjustment over the respective time horizon and solved small and medium instances in satisfactory computation time. However, the determination of all non-dominated project portfolios still remains difficult when considering large, but realistically relevant instances (100 projects or more). While this might not be relevant in most firm investment decisions, it is a significant drawback for governments or bodies awarding funding for projects and even financial institutions (Cruz et al., 2014).

A further important factor when considering the particular case of project portfolio selection processes of financial institutions is project divisibility (Urli and Terrien, 2010). When the possible decision variables are no longer binary, the complexity is yet increased. While business projects are at least partially indivisible, research projects funded by governments can often also be executed with partial funding and it is thus a further question how much of the sought after funding is awarded, introducing further constraints to the budget allocation. Hence, more recent research increasingly focuses on large-scale instances and partial allocation. Cruz et al. (2014) solve a stationary project portfolio optimization problem, in which partial support of the requested budget is allowed. Unlike previous research, they assume that the preferences of the decision-maker are to some extent known. Outranking is employed in an a priori preference system in order to model that decision makers will have preferences towards different portfolios on the efficient frontier based on their personal goals concerning the achievement of objectives. Incorporating these preferences allows identifying those portfolios that lie on the efficient frontier and simultaneously are not outranked by another portfolio. They incorporate budgetary constraints in that they define upper and lower bounds for inclusion of projects from a particular group. Fernandez et al. (2015) further enhance this approach by including synergies in their optimization.

Due to the uncertainty present in different facets of project appraisal, simulation has been incorporated into the metaheuristics framework to address this. Gabriel et al. (2006) employ MCS in their methodology to simulate possible cost scenarios for the respective optimization constraints. They analyzed a government agency facing a project portfolio decision and showed that their approach significantly improved the decision-making process and led to more robust results due to the incorporation of uncertainty. Medaglia et al. (2007) combine a multi-objective evolutionary algorithm with MCS in order to solve a project portfolio problem that allows for partial funding of projects, project interdependences, constrained resources and uncertainty in the objective function regarding the preferences of the decision-maker. Huang (2007) treats the project parameters as uncertain and combines a genetic algorithm with random fuzzy simulation in order to account for this. An interesting application combining discrete-event simulation with a genetic algorithm to select security control portfolios is discussed in Kiesling et al. (2016). In the context of an IT infrastructure subject to a number of threats, the authors focus on selecting the best policy from efficient combinations of security controls. Another way to address uncertainty is by considering robust solutions that perform reasonably well across the full range of feasible parameter values. Thus, Liesiö et al. (2007) propose a multi-objective robust portfolio modeling methodology. Their approach relies on preference programming methods. They illustrate the effectiveness of their approach in a case study involving real data from a road pavement project in Finland. In Liesiö et al. (2008), the same authors extend their previous work by also considering project interdependencies, incomplete cost information and variable budget levels. Using a personal computer, they are able to solve, in reasonable computing times, instances up to 60 projects, 5 optimization criteria, and 10 constraints. For large-size instances (e.g., with 200 projects or more), the authors suggest the use of heuristic-based approaches.

As can be seen from the reviewed examples, uncertainty can be considered for the project parameters, the modeled constraints, and the objective function. In this paper, we develop a simheuristics algorithm to address a rich and stochastic version of the PPSP with the goal of maximizing the net present value of the investment. Our approach thus addresses several gaps in the literature and combines different research challenges. Firstly, while most authors consider a particular subset of projects, such as R&D projects (Doerner et al., 2004) or government projects (Cruz et al., 2014), we formulate a general approach that can easily be applied to a range of project types, such as R&D projects, investments, or financial projects. We further include constraints on the number of projects included in the portfolio, as well as on the divisibility of project funds requested. Likewise, we give the decision maker the possibility to pre-select portfolios based on personal or strategic preferences (independently of the projects risk-return characteristics). In addition, we create instances that can be employed for comparison in future analyses, as the previously studied examples are either very small in nature or not openly accessible.

3 Formal Description of the Stochastic and Rich PPSP

We consider a stochastic and rich variant of the PPSP, in which there is a large set of candidate projects that compete for a limited global budget. The budget allocation is limited in the following ways: *(i)* if funded, each project i has to receive a minimum amount, $\epsilon_i \in (0,1]$, expressed as a percentage of the available budget (this is an amount below which the project could not yield successful results and it would thus not make sense to include it in the portfolio); *(ii)* similarly, a project is maximally funded with the amount originally requested, $\delta_i \in [\epsilon_i, 1]$, also expressed as a percentage of the available budget; *(iii)* the problem is further constrained by the decision makers' option to include certain projects irrespective of their characteristics for strategic or political reasons (this will be modeled by the binary variable q_i, which will take the value 1 whenever the project i has to be necessarily included); *(iv)* there is a minimum (k_{min}) and a maximum (k_{max}) number of projects that can be funded (these threshold values are decided by the evaluation committee in advance to assure a certain diversification level); and *(v)* the risk of the portfolio of selected projects cannot exceed a given threshold, r_{max}, where the risk is calculated employing a traditional variance-covariance matrix, $\sigma_{ij} = \sigma_{ji}$, thus accounting for interdependences between projects i and j.

Under these conditions, the projects are evaluated based on their net present value (NPV), which is defined as the difference between the present value of future cash inflows and the present value of future cash outflows. Since predicting future cash flows might be subject to some degree of uncertainty –which will be higher as we move further in the future–, these cash flows will be modeled as random variables. When computing the NPV associated with a given cash flow, an interest rate is used to transform future cash values into present ones. Since this interest rate might also be subject to some uncertainty, we will model it as a random variable too (Figure 1). Notice that the use of random variables for modeling future cash flows as well as the interest rate transform our COP into a stochastic one. In summary, under the aforementioned constraints our main goal will be to maximize the expected NPV of the project portfolio.

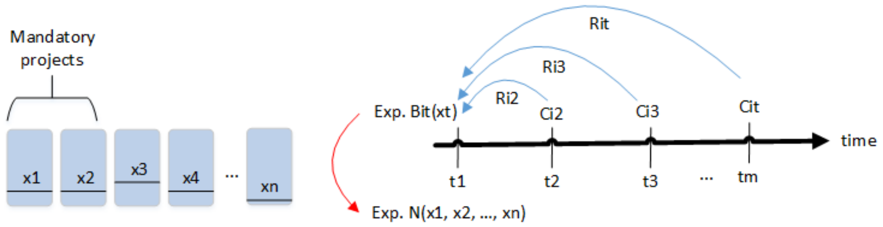

Fig. 1: Illustrative scheme of the Stochastic PPSP.

More formally, consider a set of n projects, $P = \{1, 2, ..., n\}$, and a set of m future times, $T = \{1, 2, ..., m\}$. Let us assume that the actual cash flow generated by each project is directly proportional to the quantity invested in it. Thus, for each $i \in P$ and for each $t \in T$, the actual cash flow of project i at time t will be given by $C_{it} \cdot x_i$, where C_{it} is a random variable representing the potential cash flow when project i receives the total requested funding and x_i represents the actual investment in project i (measured as a percentage of the total available budget). The periodical benefit of project i at period t, $B_{it}(x_i)$, is defined as the actual cash flow of project i at time t adjusted by a random discount rate R_{it}, i.e.:

$$B_{it}(x_i) = \frac{C_{it} \cdot x_i}{(1+R_{it})^t} \ \forall i \in P, \forall t \in T \tag{1}$$

The net present value associated with each project i, $N_i(x_i)$, is computed by adding all the periodical benefits provided by the project over time, i.e.:

$$N_i(x_i) = \sum_{t=1}^{m} B_{it}(x_i) \ \forall i \in P \tag{2}$$

Similarly, the net present value of an investment plan $x = (x_1, x_2, ..., x_n)$, $N(x)$, is obtained as the aggregation of individual net present values, i.e.:

$$N(x) = \sum_{i=1}^{n} N_i(x_i) \tag{3}$$

The goal is to find a project investment plan, x, that maximizes the expected net present value, i.e.:

$$\text{Max} \quad E[N(x)] = \sum_{i=1}^{n} \sum_{t=1}^{m} E[B_{it}(x_i)] \tag{4}$$

Also, according to our previous discussion the following constraints apply:

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \sigma_{ij} x_i x_j \leq r_{max} \tag{5}$$

$$\sum_{i=1}^{n} x_i = 1 \tag{6}$$

$$0 \leq x_i \leq 1 \ \forall i \in P \tag{7}$$

$$0 \leq \epsilon_i \leq \delta_i \leq 1 \ \forall i \in P \tag{8}$$

$$z_i \in \{0, 1\} \ \forall i \in P \tag{9}$$

$$\epsilon_i z_i \leq x_i \leq \delta_i z_i \ \forall i \in P \tag{10}$$

$$z_i \leq w \cdot x_i \ \forall i \in P \tag{11}$$

$$q_i \in \{0, 1\} \ \forall i \in P \tag{12}$$

$$q_i \leq z_i \ \forall i \in P \tag{13}$$

$$k_{min} \leq \sum_{i=1}^{n} z_i \leq k_{max} \tag{14}$$

Constraint (5) quantifies and limits the risk exposure of the decision-maker. Equations (6) and (7) restrain the total investment to the available resources. Equations (8) guarantee that the lower and upper bounds for each project are within the valid range. The auxiliary variables z_i are introduced in equations (9). The value of z_i is 1 if project i is actually included in the portfolio, and 0 otherwise. These binary variables are used in equations (10) to guarantee that the investment in each project is within its bounds. In the auxiliary equations (11), w is a very large positive value such that: if $x_i > 0$ then $wx_i \geq 1$. Equations (12) and (13) define and impose the pre-assignment constraints: if the project i is pre-selected (i.e., $q_i = 1$), it must be included in the solution (i.e., $z_i = 1$) irrespective of its risk-return characteristics. Finally, the cardinality constraint is provided in equation (14) to guarantee that the number of selected projects fits within the allowed bounds.

4 Our Simheuristic Approach

In order to solve the stochastic and rich PPSP described in the previous section, a simheuristic approach is proposed. It combines simulation techniques with an adaptive variable neighborhood search (VNS) metaheuristic –which also integrates an acceptance criterion based on simulated annealing (SA). The metaheuristic component itself relies on a constructive heuristic which employs biased randomization (BR) techniques. The main ideas behind each of these components are briefly explained next. After that, the way these components are integrated in our approach is described.

4.1 Overview of the Main Components

As described in Juan et al. (2015a), simheuristics extend metaheuristics by introducing simulation techniques that assess the performance of promising solutions in a stochastic environment: a selected subset of promising solutions generated by the metaheuristic component are simulated in order to estimate its performance under a stochastic environment. Usually, this performance is measured in terms of expected value of some key indicator, but other statistics could be analyzed as well.

The main advantage of the VNS metaheuristic (Mladenović and Hansen, 1997) lies in the search systematically employing different neighborhood structures, rendering it increasingly flexible within the solution space of the problem. This, in return, potentially leads to better solutions compared to single-neighborhood-based local search algorithms. Many extensions of VNS have been proposed, most of them oriented to the solving of large-scale instances (Melián, 2006; Moreno-Vega and Melián, 2008; Höller et al., 2008; Hansen et al., 2008).

The SA searching procedure is inspired by the process of physical annealing with solids in which a crystalline solid is heated and then allowed to cool slowly until it achieves its most regular possible crystal lattice configuration (Nikolaev and Jacobson, 2010). In a classical SA, the search starts with a high temperature and higher chance of transition to a worse solution that decreases as the search continues, thus reducing the chance of transition (Azizi and Zolfaghari, 2004). To avoid being trapped in a local minimum, our algorithm makes use of an adaptive cooling schedule, which includes the possibility of reheating.

Biased randomization (Juan et al., 2013; Grasas et al., 2017) is a mechanism to randomize deterministic heuristics. Heuristics include at least one step in which a choice has to be made, for instance, selecting an element from a sorted list. Employing BR, instead of selecting the ‘most promising’ option, a probability is assigned to each candidate option in such a way that the logic is maintained (i.e., the most promising options receive higher probabilities). Hence, different solutions may be built at each execution, some of which are expected to outperform the one provided by the deterministic version.

Algorithm 1 VNS-based Simheuristic

```
initSol ← genInitSol(Inputs) % Initial solution stage (biased-randomized heuristic)
baseSol ← initSol
fastSimulation(baseSol) % Monte Carlo simulation
bestSol ← baseSol
nIter ← 0; temperature ← 0
while (nIter $\leq$ maxIter) do % VNS stage
    k ← 1
    while (k $\leq K_{max}$) do
        newSol ← shaking(baseSol, k) % biased-randomized heuristic
        newSol ← localSearch(newSol)
        if (existInCache(newSol)) then
            newSol ← retrieveFromCache(newSol)
        else
            newSol ← localSolver(newSol)
            addToCache(newSol)
        end if
        if (detNPV(newSol) - detNPV(baseSol) $>$ 0) then
            fastSimulation(newSol) % Monte Carlo simulation
            if (stochNPV(newSol) - stochNPV(baseSol) $>$ 0) then
                baseSol ← newSol
                if (stochNPV(newSol) - stochNPV(bestSol) $>$ 0) then
                    bestSol ← newSol
                end if
                k ← 1
            end if
        else % SA-based acceptance criterion
            temperature ← calculateTemperature
            if (temperature $\geq$ Rand) then
                baseSol ← newSol
                k ← 1
            else
                k ← k + 1
            end if
        end if
    end while
    nIter ← nIter + 1
end while
deepSimulation(bestSol) % Refinement stage - Monte Carlo simulation
return bestSol
```

4.2 Algorithm Description

Our approach is depicted in Algorithm 1, which is composed of three stages. In the first stage, a feasible initial solution is constructed. Then, during the second stage, an adaptive VNS metaheuristic enhances the initial feasible solution by iteratively exploring the search space and conducting a short number of simulation runs in which both cash flows and discount rates are randomly generated to estimate the expected net present value. From this stage, a reduced set of promising solutions is obtained. In the third stage, an extended simulation experiment yields a more accurate estimate of the expected NPV, as well as other statistics whenever required. The initial-solution stage as well as the VNS stage are explained next in more detail.

Given the stochastic instance, we consider its deterministic counterpart obtained after replacing the random variables by their expected values. In order to generate an initial solution, *initSol*, we randomly choose a valid size s for the project portfolio (i.e., $k_{min} \leq s \leq k_{max}$ and $s \geq \sum q_i$). First, the pre-selected projects are included (Algorithm 2). Then, we randomly select projects until a portfolio of size s is generated. In order to set the weights of each project in this portfolio, we apply *LocalSolver*, a powerful optimization software (http://www.localsolver.com). This software was selected due to its ability to consider quadratic expressions as constraints, which is the case of equation (5). The entire process is repeated until the randomly generated solution satisfies all the constraints (i.e., until a feasible solution is obtained). Notice that, being a project-investment plan, *initSol* will be a feasible solution for both the deterministic and the stochastic versions of the problem.

Algorithm 2 genInitSol(projectList, maxRisk, K_{min}, K_{max})

```
feasibleSol ← false
initSol ← null
auxProjectList ← copyProjectList(projectList)
while (feasibleSol is false) do
    s ← defineRandPortfolioSize($K_{min}$,$K_{max}$)
    selectedProjects ← selectPreselectedProjects(projectList)
    projectList ← removeProject(projectList, selectedProjects)
    portfolio ← addProjectToPortfolio(portfolio, selectedProjects)
    currentPortfolioSize ← getPortfolioSize(portfolio)
    while ( currentPortfolioSize < s) do
        project ← selectProjectRandomly(projectList)
        projectList ← removeProject(projectList, project)
        portfolio ← addProjectToPortfolio(project)
        currentPortfolioSize ← currentPortfolioSize + 1
    end while
    initSol ← localSolver(portfolio)
    if (satisfyConstraints(initSol, maxRisk) then
        feasibleSol ← true
    else
        projectList ← copyProjectList(auxProjectList)
        portfolio ← deletePortfolio(portfolio)
    end if
end while
return initSol
```

In the second stage, the *initSol* is improved using a VNS procedure. First, the feasible *initSol* is copied into *baseSol* and *bestSol*. Moreover, the size of the neighborhood, k, is set to one and the SA-related temperature is set to zero. Then, a new solution, *newSol*, is created by shaking the current one. This procedure consists of randomly deleting a number of non pre-selected projects in the current portfolio and then randomly introducing new projects. Each *newSol* is send to *LocalSolver* to determine the appropriate investment weights. In order to avoid calling the solver more than strictly necessary, a cache memory is implemented –it stores, in a hash map data structure, the weights assigned to previously analyzed project portfolios. Next, *newSol* undergoes a local search phase in order to find the local minimum within the defined neighborhood structure. In this local search phase, we randomly substitute projects from the current portfolio by projects outside the portfolio. This replacement is performed taking into account the risk-affinity between projects (i.e., the covariance matrix), and a BR technique relying on a geometric distribution is employed (Juan et al., 2015b). If *newSol* is promising in terms of deterministic NPV, then it is send through a fast simulation process, consisting of 200 runs, to estimate its associated expected NPV under uncertainty. Whenever this expected NPV outperforms the one of the *baseSol* and/or the one of the *bestSol*, these solutions are updated to *newSol* and the process continues. Also, to reduce the odds of getting trapped in a local minimum, an SA-like acceptance criterion is used to update *baseSol* with *newSol* in some occasions even if *newSol* does not outperform *baseSol*.

Once the VNS stage ends, the algorithm returns a selected list with of top 5 solutions. For each of these solutions, we perform a more intensive simulation experiment, consisting of $15,000$ runs, which provides a more accurate estimate of the expected NPV. Notice that the outcomes of this simulation experiment can also be used to complete a risk analysis on each proposed solution as well as to obtain other relevant statistics –e.g., NPV quartiles associated with each investment plan, etc.

5 Computational Experiments

The algorithm was implemented as a Java application and all experiments were performed on a standard personal computer equipped with Intel Core i7 CPU at 2.9 GHz with 8GB of RAM memory. As operating system we have used Windows 8. In order to test it, we created benchmark indexes with a set of 10 projects and the corresponding required inputs. In a preliminary analysis, we identified a range of acceptable risk levels, which, as similarly suggested for traditional portfolio optimization, we then divided into 1000 equidistant points as risk constraint. The so-created benchmarks differ in terms of their interdependence (correlation) between projects and are deterministic. In this regard, we distinguish among 6 different instances. Instance 1 assumes 0 correlation among each pair of projects. Instance 2 assumes that each pair of projects is correlated with the coefficient of 0.9. This coefficient is 0.5 for instance 3, -0.5 for instance 4, and -0.9 for instance 5. Finally, Instance 6 randomly generates the coefficients of correlation for each pair of projects. To evaluate the effects of uncertainty on project portfolio selection, we make several reasonable distributional assumptions about our stochastic variables C_{it} and R_{it}.

It is assumed that the cash flow of project i at time t, C_{it}, follows a normal distribution $N(\mu_{it}, \sigma_{it})$, where $\mu_{it} = E[C_{it}]$ is the deterministic value for the cash flow of project i at time t (given as an input), and $\sigma_{it} = \gamma \cdot |\mu_{it}| \cdot t$. In the previous expression, $\gamma > 0$ is an auxiliary parameter that is used to consider different variability levels in our experiments (intra-period layer of uncertainty), while t accounts for the fact that uncertainty grows as we move forward in the future (inter-period layer of uncertainty). In our experiments, we assume three different levels of intra-period uncertainty: low ($\gamma = 1.05$), medium ($\gamma = 1.10$), and high ($\gamma = 1.15$). For the purpose of our computational experiments we have used a total of five periods, i.e., $t \in \{1, 2, 3, 4, 5\}$. Similarly, it is assumed that the discount rate of project i at time t, R_{it}, follows a normal distribution $N(\mu'_{it}, \sigma'_{it})$, where $\mu'_{it} = E[R_{it}]$ is the deterministic value for the discount rate of project i at time t (given as an input), and $\sigma'_{it} = \gamma \cdot |\mu'_{it}| \cdot t$. As in the case of uncertain cash flows, the discount rate features two uncertainty layers, the intra-period layer of uncertainty and the inter-period one. It is worth noting that our methodology is robust to modification of the aforementioned distributional properties, either theoretical or empirically based on real-life scenarios.

The algorithm is executed ten times with different seeds, storing only the best solutions in each run. A maximum time of 150 seconds was allowed for each execution. In order to gauge the effect of uncertainty on project portfolio selection, we carried out two computational experiments. The first experiment considers deterministic instances, while the second assumes stochastic cash flows and discount rates.

6 Analysis of Results

In this section we present and discuss the results obtained in two computational experiments: Subsection 6.1 analyzes the results obtained for the deterministic version of the PPSP (i.e., assuming deterministic cash flows and discount rates), while Subsection 6.2 discusses the stochastic PPSP and shows that an optimal (or near-optimal) investment plan for the deterministic scenario might be a suboptimal plan for the stochastic one.

6.1 Analyzing the Deterministic PPSP

Figure 2 presents these results by instance. This figure sheds light on the relation between the maximum risk constraint and the associated expected NPV for the six instances.

Notice that relaxing the risk constraint leads to a higher average NPV. However, several departures from this general finding stand out. First, the nature of the relation between the expected NPV is also driven by the coefficient of correlation between each pair of projects. Portfolios of projects that are highly correlated (instances 2 and 3) typically generate a lower expected NPV than portfolios of less correlated projects (Instances 1, 4, and 5). Finally, Instance 6 also features positively correlated projects, although based on randomly drawn coefficients of correlation. Specifically, if the coefficient of correlation is high and the maximum allowed risk is low, then the portfolio

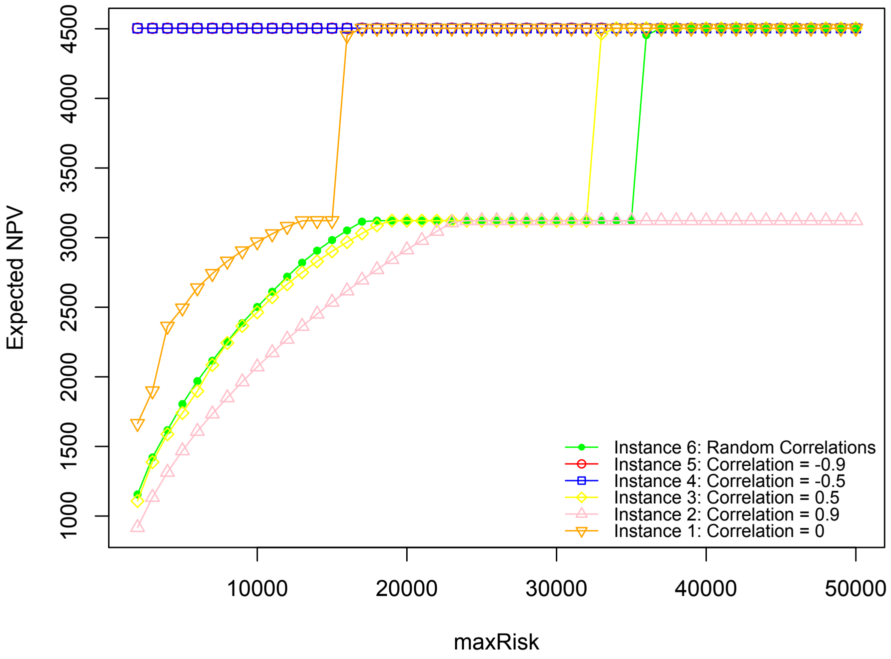

Fig. 2: Results for the Deterministic PPSP.

will be dominated by smaller projects that generate lower expected NPV. Intuitively, the higher the correlation among the projects the more limiting is the maximum risk constraint. To meet the maximum risk constraint, larger projects are crowded out –or cannibalized– by smaller projects. As the coefficient of correlation decreases, the expected NPV increases for the same value of the maximum allowed risk (due to a lower cannibalism effect). As anticipated, the NPV of the best-found portfolio increases with an increased allowance of risk exposure for each of the instances. Considering instances 4 and 5 (with negative correlations), it becomes clear that the risk constraint –which is always defined as a positive monetary value– does not affect the solution, and that a near-optimal solution is found in most cases. Similarly, it is intuitive that the higher the correlation between projects, the more limiting the risk constraint acts. This becomes visible when considering instance 2, which achieves a lower or equal NPV for all levels of maximum risk compared to the remaining instances. The jumps in the graphs of instances 1, 3, and 6 are due to the limited number of potential projects, which causes projects with higher benefits to be included only after a certain risk threshold.

6.2 Analyzing the Stochastic PPSP

For instance 6 (random correlations), Table 1 presents the obtained results. The first column shows the risk threshold considered (i.e., the maximum risk allowed by the user). Column [DD] represents the NPV associated with the best deterministic solution. Columns [Dy] (with $y \in \{L, M, H\}$) show the expected NPV obtained when the best deterministic solution is evaluated in the stochastic PPSP with the corresponding level of uncertainty (L = low, M = medium, and H = high). Similarly, columns [Sy] show the expected NPV obtained for the stochastic PPSP using the solution provided by our simheuristic approach. Table 1 indicates that, in general, the (expected) NPV increases with the maximum risk allowed, regardless of the solution type (deterministic or stochastic). However, it is worth noting that the NPV-curve becomes flat after the project portfolio reaches a certain risk treshold. For instance, when the risk threshold is 37,000 units, taking an additional unit of risk is not followed by a commensurate rise in the (expected) NPV. Furthermore, a deterministic solution evaluated in the stochastic PPSP yields a lower expected NPV than the corresponding stochastic solution obtained with our simheuristic aproach. This finding suggests that using deterministic solutions in a stochastic environment can lead to significant biases in project appraisal. Table 2 contains the gaps (measured as percentage difference) between [DD] and [Dy], and between [DD] and [Sy]. The gaps are always positive, which indicates that a higher degree of uncertainty erodes the expected NPV of a project and hence drives wedge between a deterministic and a stochastic solution. It is also worth noting than the gaps between [DD] and [Dy] are larger than the gaps between [DD] and [Sy].

Figure 3 shows the box-plots of the aforementioned gaps between DD and Dy or between DD and Sy (where $y \in \{L, M, H\}$). It is important to remark that these gaps are always positive, meaning that the NPV in the deterministic PPSP can be seen as an upper bound for the expected NPV in the stochastic PPSP –i.e., *ceteris paribus*, the existence of uncertainty in the project selection problem reduces the average quality of the NPV that can be attained. Indeed, the NPV is inversely related to the degree of uncertainty. For a given expected NPV, a higher degree of uncertainty translates into a larger standard deviation of a cashflow, and consequently the maximum risk allowed goes up. However, such as a reduction larger if the existence of uncertainty is ignored in the model. Also, note that these gaps grow with the uncertainty level: eH box-plots are above eM box-plots, which in turn are above eL box-plots, (where $e \in \{D, S\}$). Finally, it can be noticed that the expected NPV associated with the use of the best-deterministic solution in a stochastic environment acts as a lower-bound for the optimal value in the stochastic environment, i.e.: our solutions for the stochastic PPSP always outperform the expected NPV generated by using the best-deterministic solution in a stochastic environment. This observation not only adds credibility to the quality of our algorithm but also illustrates that using the best deterministic solution in a stochastic environment might be a bad decision since it usually provides sub-optimal values.

Max. Risk	Best Deterministic Solution				Best Stochastic Solution		
(in	Det.	Stoch. Low	Stoch. Med.	Stoch. High	Stoch. Low	Stoch. Med.	Stoch. High
thousands)	[DD]	[DL]	[DM]	[DH]	[SL]	[SM]	[SH]
2	1154.84	1103.00	1058.95	1043.74	1110.34	1093.22	1086.08
3	1420.82	1311.37	1306.22	1301.07	1332.20	1323.64	1319.05
4	1642.84	1535.16	1530.10	1525.05	1622.32	1604.95	1581.12
5	1836.75	1725.84	1720.63	1715.43	1816.08	1805.12	1766.02
6	1992.29	1872.40	1866.77	1861.16	1978.24	1958.11	1913.42
7	2115.22	1988.82	1982.87	1976.92	2102.21	2069.82	1998.97
8	2252.14	2127.32	2121.44	2115.57	2250.85	2201.07	2129.89
9	2386.90	2259.19	2253.18	2247.17	2320.56	2318.39	2261.88
10	2512.24	2383.79	2377.74	2371.71	2436.29	2428.58	2377.44
11	2632.33	2505.42	2499.46	2493.50	2545.21	2544.76	2499.42
12	2744.81	2616.56	2610.54	2604.52	2671.54	2651.98	2620.32
13	2843.14	2714.25	2708.19	2702.14	2784.26	2754.22	2740.02
14	2935.72	2805.69	2799.57	2793.47	2891.70	2853.29	2847.41
15	3025.78	2895.41	2889.28	2883.16	2984.07	2941.11	2936.00
16	3111.63	2980.59	2974.42	2968.27	2991.27	2986.12	2975.07
17	3194.71	3064.06	3057.91	3051.77	3122.54	3114.57	3105.33
18	3275.17	3144.15	3137.98	3131.82	3198.29	3177.38	3158.96
19	3351.05	3218.31	3212.06	3205.81	3291.92	3280.03	3271.88
20	3426.18	3294.59	3288.39	3282.20	3375.23	3364.32	3345.43
21	3497.78	3363.88	3357.58	3351.29	3413.32	3404.49	3390.69
22	3554.55	3422.14	3415.91	3409.68	3430.27	3418.54	3413.78
23	3604.21	3472.34	3466.13	3459.92	3546.32	3506.41	3494.02
24	3666.31	3535.52	3529.36	3523.20	3624.75	3618.37	3606.41
25	3666.31	3535.52	3529.36	3523.20	3624.75	3618.37	3606.41
26	3666.31	3535.52	3529.36	3523.20	3624.75	3618.37	3606.41
27	3666.31	3535.52	3529.36	3523.20	3624.75	3618.37	3606.41
28	3666.31	3535.52	3529.36	3523.20	3624.75	3618.37	3606.41
29	3666.31	3535.52	3529.36	3523.20	3624.75	3618.37	3606.41
30	3666.31	3535.52	3529.36	3523.20	3624.75	3618.37	3606.41
31	3666.31	3535.52	3529.36	3523.20	3624.75	3618.37	3606.41
32	3666.31	3535.52	3529.36	3523.20	3624.75	3618.37	3606.41
33	3666.31	3535.52	3529.36	3523.20	3624.75	3618.37	3606.41
34	3666.31	3535.52	3529.36	3523.20	3624.75	3618.37	3606.41
35	3666.31	3535.52	3529.36	3523.20	3624.75	3618.75	3606.41
36	4454.73	4349.38	4344.43	4339.50	4412.34	4407.93	4399.93
37	4503.66	4397.67	4392.70	4387.73	4455.08	4452.91	4450.07
38	4503.66	4397.67	4392.70	4387.73	4455.08	4452.91	4450.07
39	4503.66	4397.67	4392.70	4387.73	4455.08	4452.91	4450.07
40	4503.66	4397.67	4392.70	4387.73	4455.08	4452.91	4450.07
41	4503.66	4397.67	4392.70	4387.73	4455.08	4452.91	4450.07
42	4503.66	4397.67	4392.70	4387.73	4455.08	4452.91	4450.07
43	4503.66	4397.67	4392.70	4387.73	4455.08	4452.91	4450.07
44	4503.66	4397.67	4392.70	4387.73	4455.08	4452.91	4450.07
45	4503.66	4397.67	4392.70	4387.73	4455.08	4452.91	4450.07
46	4503.66	4397.67	4392.70	4387.73	4455.08	4452.91	4450.07
47	4503.66	4397.67	4392.70	4387.73	4455.08	4452.91	4450.07
48	4503.66	4397.67	4392.70	4387.73	4455.08	4452.91	4450.07
49	4503.66	4397.67	4392.70	4387.73	4455.08	4452.91	4450.07
50	4503.66	4397.67	4392.70	4387.73	4455.08	4452.91	4450.07

Table 1: This table summarizes the net present value (NPV) of deterministic (Dy) and stochastic (Sy) solutions in a deterministic (eD) or stochastic (eL, eM, eH) environment (Instance 6).

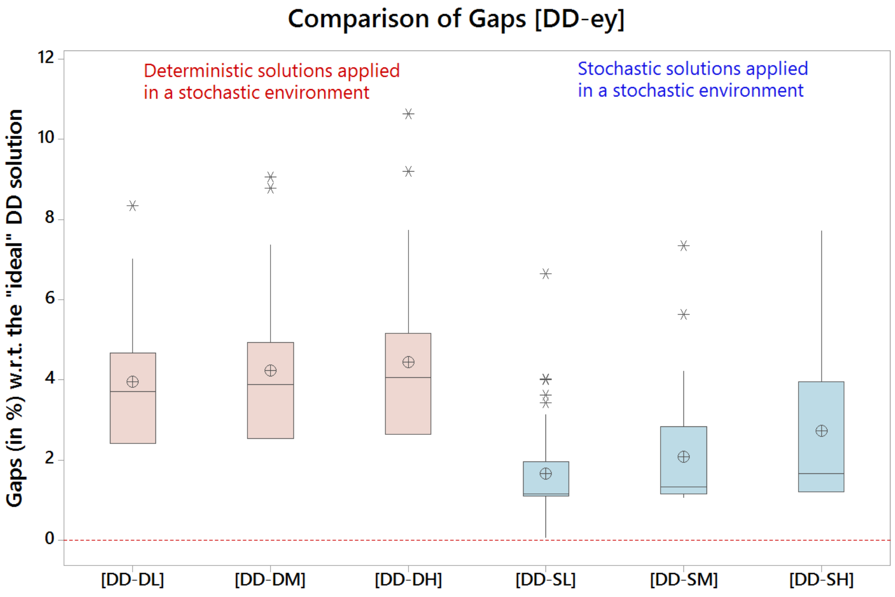

Fig. 3: Boxplots of Gaps w.r.t. the Best Solution for the Deterministic PPSP (DD).

Max. Risk	Deterministic	Gaps [%]					
(in thousands)	[DD]	[DD-DL]	[DD-DM]	[DD-DH]	[DD-SL]	[DD-SM]	[DD-SH]
2	1154.84	4.70%	9.06%	10.64%	4.01%	5.64%	6.33%
3	1420.82	8.35%	8.77%	9.20%	6.65%	7.34%	7.72%
4	1642.84	7.01%	7.37%	7.72%	1.26%	2.36%	3.90%
5	1836.75	6.43%	6.75%	7.07%	1.14%	1.75%	4.01%
6	1992.29	6.40%	6.72%	7.05%	0.71%	1.75%	4.12%
7	2115.22	6.36%	6.67%	7.00%	0.62%	2.19%	5.82%
8	2252.14	5.87%	6.16%	6.46%	0.06%	2.32%	5.74%
9	2386.90	5.65%	5.93%	6.22%	2.86%	2.96%	5.53%
10	2512.24	5.39%	5.66%	5.93%	3.12%	3.44%	5.67%
11	2632.33	5.07%	5.32%	5.57%	3.42%	3.44%	5.32%
12	2744.81	4.90%	5.14%	5.39%	2.74%	3.50%	4.75%
13	2843.14	4.75%	4.98%	5.22%	2.11%	3.23%	3.76%
14	2935.72	4.63%	4.86%	5.09%	1.52%	2.89%	3.10%
15	3025.78	4.50%	4.72%	4.95%	1.40%	2.88%	3.06%
16	3111.63	4.40%	4.61%	4.83%	4.02%	4.20%	4.59%
17	3194.71	4.26%	4.47%	4.68%	2.31%	2.57%	2.88%
18	3275.17	4.17%	4.37%	4.58%	2.40%	3.08%	3.68%
19	3351.05	4.12%	4.33%	4.53%	1.80%	2.17%	2.42%
20	3426.18	3.99%	4.19%	4.39%	1.51%	1.84%	2.41%
21	3497.78	3.98%	4.18%	4.37%	2.47%	2.74%	3.16%
22	3554.55	3.87%	4.06%	4.25%	3.62%	3.98%	4.12%
23	3604.21	3.80%	3.98%	4.17%	1.63%	2.79%	3.15%
24	3666.31	3.70%	3.88%	4.06%	1.15%	1.32%	1.66%
25	3666.31	3.70%	3.88%	4.06%	1.15%	1.32%	1.66%
26	3666.31	3.70%	3.88%	4.06%	1.15%	1.32%	1.66%
27	3666.31	3.70%	3.88%	4.06%	1.15%	1.32%	1.66%
28	3666.31	3.70%	3.88%	4.06%	1.15%	1.32%	1.66%
29	3666.31	3.70%	3.88%	4.06%	1.15%	1.32%	1.66%
30	3666.31	3.70%	3.88%	4.06%	1.15%	1.32%	1.66%
31	3666.31	3.70%	3.88%	4.06%	1.15%	1.32%	1.66%
32	3666.31	3.70%	3.88%	4.06%	1.15%	1.32%	1.66%
33	3666.31	3.70%	3.88%	4.06%	1.15%	1.32%	1.66%
34	3666.31	3.70%	3.88%	4.06%	1.15%	1.32%	1.66%
35	3666.31	3.70%	3.88%	4.06%	1.15%	1.31%	1.66%
36	4454.73	2.42%	2.54%	2.66%	0.96%	1.06%	1.25%
37	4503.66	2.41%	2.53%	2.64%	1.09%	1.14%	1.20%
38	4503.66	2.41%	2.53%	2.64%	1.09%	1.14%	1.20%
39	4503.66	2.41%	2.53%	2.64%	1.09%	1.14%	1.20%
40	4503.66	2.41%	2.53%	2.64%	1.09%	1.14%	1.20%
41	4503.66	2.41%	2.53%	2.64%	1.09%	1.14%	1.20%
42	4503.66	2.41%	2.53%	2.64%	1.09%	1.14%	1.20%
43	4503.66	2.41%	2.53%	2.64%	1.09%	1.14%	1.20%
44	4503.66	2.41%	2.53%	2.64%	1.09%	1.14%	1.20%
45	4503.66	2.41%	2.53%	2.64%	1.09%	1.14%	1.20%
46	4503.66	2.41%	2.53%	2.64%	1.09%	1.14%	1.20%
47	4503.66	2.41%	2.53%	2.64%	1.09%	1.14%	1.20%
48	4503.66	2.41%	2.53%	2.64%	1.09%	1.14%	1.20%
49	4503.66	2.41%	2.53%	2.64%	1.09%	1.14%	1.20%
50	4503.66	2.41%	2.53%	2.64%	1.09%	1.14%	1.20%
	Average:	3.94%	4.22%	4.44%	1.66%	2.08%	2.72%

Table 2: The tables summarizes the gap between the DD solution and each of the other solutions (Instance 6).

Finally, Figure 4 shows that: *(i)* as the maximum risk allowed increases, the problem becomes less constrained and, therefore, the stochastic solutions get closer to the deterministic ones; and *(ii)* for each risk level, gap obtained from our stochastic solutions [DD-Sy] are lower than those obtained from our deterministic solutions in a stochastic environment [DD-Dy].

6.3 Exploring a Multi-objective Scenario

In multi-objective optimization two or more conflicting goals are considered. This subsection explores the optimization of a bi-objective PPSP model, where the NPV associated with an investment plan is maximized (first goal) while, at the same time, its risk is minimized (second goal). Such a setting resembles Rooderkerk and van Heerde (2016), who have sought to balance risk and return, albeit in a specific retail-assortment related optimization problem. Importantly, a two-objective optimization problem can now be tailored to capture a varying degree of risk aversion of the decision maker.

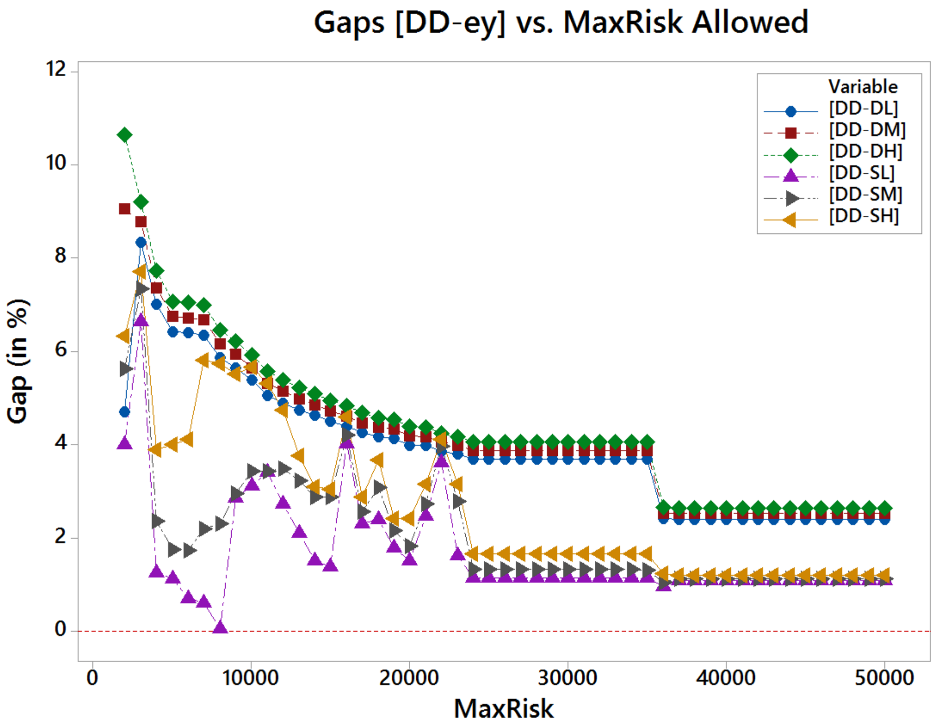

Fig. 4: Gaps [DD-*ey*] vs. MaxRisk Allowed (where '*ey*' represents different solutions).

In order to address this multi-objective scenario, we make use of the Multi-Directional Local Search (MDLS) method, which was introduced by Tricoire (2012). This method relies on the concept of Pareto dominance. A neighbor solution *x'* of x is efficient if *x'* is better than x for at least one objective. Hence, to find efficient neighbor solutions of x, it is sufficient with searching one direction at a time using single-objective local search methods.

The MDLS requires an initial set F of non-dominated solutions to start an iterative procedure. As initial set, we have used the set F containing the top five stochastic non-dominated solutions generated by our VNS-based Simheuristic. At each iteration, a solution x from F is randomly selected and then, for each objective, a corresponding local search method is employed to generate a neighbor solution *x'*.

The single-objective local searches employed in this stage are the ones already described in the previous section. First, we use biased randomization to exchange projects from the current portfolio by projects outside the portfolio. This replacement is performed taking into account the risk-affinity between projects. Then, the newly generated solution is sent to the commercial *LocalSolver* software, which determines the appropriate weights of the investment. We execute *LocalSolver* twice, once for each goal. Thus, during the first itera-

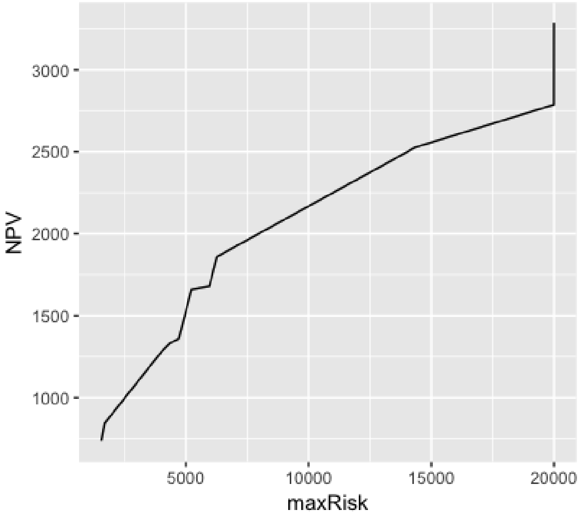

Fig. 5: Pareto frontier for instance 6.

tion the *LocalSolver* tries to maximize the NPV, while during the second one it is focused on minimizing the investment risk. Finally, at the end of each local search, a simulation process is carried out in order to estimate the NPV under uncertainty.

After that, the non-dominated set F is updated by merging solutions in F with the new neighbor solutions that have been provided by the local searches. To merge solutions the dominance rule is used, i.e.: all dominated solutions are deleted from F. The whole procedure is repeated until a maximum-allowed computing time is reached. At this point, the MDLS returns the set F of mutually non-dominated solutions.

Figure 5 shows a Pareto chart with the results obtained after executing the MDLS method for instance 6. Here, a maximum risk of 21,000 was allowed, and a medium ($\gamma = 1.10$) level of uncertainty was considered. This chart shows the Pareto frontier of non-dominated solutions for this particular scenario. Notice that there is a strong relationship between the two objective functions. As the NPV increases –thus obtaining a higher return of the investment plan– the risk also increases.

7 Conclusions and Future Work

As far as we know, this is the first work addressing the relevant problem of maximizing net present value in project portfolio selection problem (PPSP) under uncertainty and rich conditions. In the stochastic version of the PPSP discussed here, both the periodical cash flows as well as the discount rates are modeled as random variables, which represents a step forward with regards traditional approaches in which they are assumed to be deterministic and known in advance. Since real-life financial activities underlie plenty of uncertainty, adding randomness to these elements contributes to diminish the gap between theory and practice.

After completing a literature review on related work and providing a formal model for the stochastic PPSP, we propose a simheuristic approach to obtain efficient solutions to it. Our algorithm integrates Monte Carlo simulation inside a variable neighborhood search framework. The algorithm also uses other components inspired in simulated annealing and biased-randomization techniques. A series of computational experiments allow us to validate the solving methodology. Notice that, despite we have assumed specific probability distribution and covariance matrices during the numerical experiments, our methodology is general and it could be used with any other probability distribution and covariances (in practice, data should be collected for each case-study and then modeled using the right probability distribution, correlation values among projects, etc.).

Our research findings are as follows. First, we find that a relation between the expected NPV and risk is not necessarily linear. Indeed, the presence of cardinality and quantity constraints generate non-linearities in the risk-return relation. Second, project interdependencies –as measured by the correlation between cash-flows from two projects– can be regarded as a limit to the volume of projects that can be included in a portfolio. Third, a near-optimal solution to the deterministic PPSP is generally sub-optimal in a stochastic environment. Fourth, a near-optimal solution to the stochastic PPSP gives rise to a higher (expected) NPV than a near-optimal solution to the deterministic PPSP evaluated in a stochastic environment.

As research lines for future work, it would be worthy to explore the following topics, which represent limitations of our current work: *(i)* interdependencies can be defined as effects that dynamically influence risk and/or benefit figures of the project portfolio based on the constituent projects; *(ii)* although this paper has already explored a bi-objective extension of our single-objective approach, developing a complete multi-objective model and solving methodology can be of great interest for both researchers and practitioners; and *(iii)* other metaheuristic frameworks (e.g.: tabu search, simulated annealing, genetic algorithms, etc.) could be used as a base for our simheuristic approach, thus comparing the performance of different approaches.

Acknowledgements This work has been partially supported by the Spanish Ministry of Economy and Competitiveness (TRA2013-48180-C3-P, TRA2015-71883-REDT), FEDER. and the Erasmus+ programme (20161ES01KA108023465).

References

Azizi, N. and Zolfaghari, S. (2004). Adaptive temperature control for simulated annealing: A comparative study. *Computers and Operations Research*, 31(14):2439–2451.

Carazo, A. F., Gómez, T., Molina, J., Hernández-Díaz, A. G., Guerrero, F. M., and Caballero, R. (2010). Solving a comprehensive model for multiobjective project portfolio selection. *Computers & Operations Research*, 37(4):630–639.

Coldrick, S., Longhurst, P., Ivey, P., and Hannis, J. (2005). An R&D options selection model for investment decisions. *Technovation*, 25(3):185–193.

Cruz, L., Fernandez, E., Gomez, C., Rivera, G., and Perez, F. (2014). Many-objective portfolio optimization of interdependent projects with 'a priori' incorporation of decision-maker preferences. *Applied Mathematics and Information Sciences*, 8(4):1517–1531.

Doerner, K., Gutjahr, W. J., Hartl, R. F., Strauss, C., and Stummer, C. (2004). Pareto ant colony optimization: A metaheuristic approach to multiobjective portfolio selection. *Annals of Operations Research*, 131(1-4):79–99.

Doerner, K. F., Gutjahr, W. J., Hartl, R. F., Strauss, C., and Stummer, C. (2006). Pareto ant colony optimization with ILP preprocessing in multiobjective project portfolio selection. *European Journal of Operational Research*, 171(3):830–841.

Fernandez, E., Gomez, C., Rivera, G., and Cruz-Reyes, L. (2015). Hybrid metaheuristic approach for handling many objectives and decisions on partial support in project portfolio optimisation. *Information Sciences*, 315:102–122.

Gabriel, S. A., Kumar, S., Ordòñez, J., and Nasserian, A. (2006). A multiobjective optimization model for project selection with probabilistic considerations. *Socio-Economic Planning Sciences*, 40(4):297–313.

Ghasemzadeh, F. and Archer, N. (2000). Project portfolio selection through decision support. *Decision Support Systems*, 29(1):73–88.

Grasas, A., Juan, A. A., Faulin, J., de Armas, J., and Ramalhinho, H. (2017). Biased randomization of heuristics using skewed probability distributions: a survey and some applications. *Computers & Industrial Engineering*.

Gutjahr, W. J., Katzensteiner, S., Reiter, P., Stummer, C., and Denk, M. (2008). Competence-driven project portfolio selection, scheduling and staff assignment. *Central European Journal of Operations Research*, 16(3):281–306.

Gutjahr, W. J., Katzensteiner, S., Reiter, P., Stummer, C., and Denk, M. (2010). Multi-objective decision analysis for competence-oriented project portfolio selection. *European Journal of Operational Research*, 205(3):670–679.

Hansen, P. and Mladenović, N. (2001). Variable neighborhood search: Principles and applications. *European Journal of Operational Research*, 130(3):449–467.

Hansen, P. and Mladenović, N. (2014). Variable neighborhood search. In *Search methodologies*, pages 313–337. Springer.

Hansen, P., Mladenović, N., et al. (2008). Variable neighborhood search. *European Journal of Operational Research*, 191(3):593–595.

Höller, H., Melián, B., and Voß, S. (2008). Applying the pilot method to improve VNS and GRASP metaheuristics for the design of SDH/WDM networks. *European Journal of Operational Research*, 191(3):691–704.

Huang, X. (2007). Optimal project selection with random fuzzy parameters. *International Journal of Production Economics*, 106(2):513–522.

Juan, A. A., Faulin, J., Ferrer, A., Lourenço, H. R., and Barrios, B. (2013). Mirha: multi-start biased randomization of heuristics with adaptive local search for solving non-smooth routing problems. *Top*, 21(1):109–132.

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

Juan, A. A., Pascual, I., Guimarans, D., and Barrios, B. (2015b). Combining biased randomization with iterated local search for solving the multidepot vehicle routing problem. *International Transactions in Operational Research*, 22(4):647–667.

Kiesling, E., Ekelhart, A., Grill, B., Strauss, C., and Stummer, C. (2016). Selecting security control portfolios: a multi-objective simulation-optimization approach. *EURO Journal on Decision Processes*, 4(1-2):85–117.

Liesiö, J., Mild, P., and Salo, A. (2007). Preference programming for robust portfolio modeling and project selection. *European Journal of Operational Research*, 181(3):1488–1505.

Liesiö, J., Mild, P., and Salo, A. (2008). Robust portfolio modeling with incomplete cost information and project interdependencies. *European Journal of Operational Research*, 190(3):679–695.

Medaglia, A. L., Graves, S. B., and Ringuest, J. L. (2007). A multiobjective evolutionary approach for linearly constrained project selection under uncertainty. *European Journal of Operational Research*, 179(3):869–894.

Melián, B. (2006). Using memory to improve the vns metaheuristic for the design of sdh/wdm networks. In *Proceedings of the Third International Conference on Hybrid Metaheuristics*, HM'06, pages 82–93, Berlin, Heidelberg. Springer-Verlag.

Mladenović, N. and Hansen, P. (1997). Variable neighborhood search. *Computers & Operations Research*, 24(11):1097–1100.

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

Nikolaev, A. G. and Jacobson, S. H. (2010). Simulated annealing. In Gendreau, M. and Potvin, J.-Y., editors, *Handbook of Metaheuristics*, volume 146 of *International Series in Operations Research & Management Science*, pages 1–39. Springer US, 2nd edition.

Patrick, H. T. (1966). Financial Development and Economic Growth in Underdeveloped Countries. *Economic Development and Cultural Change*, 14(2):174–189.

Rabbani, M., Aramoon Bajestani, M., and Baharian Khoshkhou, G. (2010). A multi-objective particle swarm optimization for project selection problem. *Expert Systems with Applications*, 37(1):315–321.

Rooderkerk, R. P. and van Heerde, H. J. (2016). Robust optimization of the 01 knapsack problem: Balancing risk and return in assortment optimization. *European Journal of Operational Research*, 250(3):842 – 854.

Schmidt, R. L. (1993). A Model for R & D Project Selection with Combined Benefit, Outcome and Resource Interactions. *IEEE Transactions on Engineering Management*, 40(4):403–410.

Soler, A., Juan, A., and Kizys, R. (2017). A survey on financial applications of metaheuristics. *ACM Computing Surveys*.

Stummer, C. and Sun, M. (2005). New Multiobjective Metaheuristic Solution Procedures for Capital Investment Planning. *Journal of Heuristics*, 11(3):183–199.

Suh, C.-K., Suh, E.-H., and Baek, K.-C. (1994). Prioritizing telecommunications technologies for long-range r&d planning to the year 2006. *IEEE Transactions on Engineering Management*, 41(3):264–275.

Tricoire, F. (2012). Multi-directional local search. *Computers & Operations Research*, 39(12):3089 – 3101.

Urli, B. and Terrien, F. (2010). Project portfolio selection model, a realistic approach. *International Transactions in Operational Research*, 17(6):809–826.