VALUE OF MODELING UNCERTAINTY IN MULTI-OBJECTIVE PROGRAMMING

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ECE YİĞİT

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
INDUSTRIAL ENGINEERING

SEPTEMBER 2019

Approval of the thesis:

VALUE OF MODELING UNCERTAINTY IN MULTI-OBJECTIVE PROGRAMMING

submitted by **ECE YİĞİT** in partial fulfillment of the requirements for the degree of **Master of Science in Industrial Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** __________

Prof. Dr. Yasemin Serin
Head of Department, **Industrial Engineering** __________

Assist. Prof. Dr. Melih Çelik
Supervisor, **Industrial Engineering** __________

Assist. Prof. Dr. Sakine Batun __________
Co-supervisor, **Industrial Engineering**

Examining Committee Members:

Prof. Dr. Meral Azizoğlu
Industrial Engineering Department, METU __________

Assist. Prof. Dr. Melih Çelik
Industrial Engineering Department, METU __________

Assist. Prof. Dr. Sakine Batun
Industrial Engineering Department, METU __________

Prof. Dr. Esra Karasakal
Industrial Engineering Department, METU __________

Assist. Prof. Dr. Diclehan Tezcaner Öztürk
Industrial Engineering Department, Hacettepe University __________

Date:

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Ece Yiğit

Signature :

ABSTRACT

VALUE OF MODELING UNCERTAINTY IN MULTI-OBJECTIVE PROGRAMMING

Yiğit, Ece
M.S., Department of Industrial Engineering
Supervisor: Assist. Prof. Dr. Melih Çelik
Co-Supervisor : Assist. Prof. Dr. Sakine Batun

September 2019, 74 pages

In this thesis, the value of modeling uncertainty in multi-objective problems is investigated. First, a mathematical model for a general two-stage multi-objective stochastic problems is introduced. Then, a new approach is presented for calculating the value of the stochastic solution and the expected value of perfect information for such problems. Computational experiments are provided to test the validity and the performance of the proposed methodology considering a bi-objective problems that involve uncertainty.

Keywords: stochastic programming, multi-objective programming, value of stochastic solution, expected value of perfect information

ÖZ

ÇOK AMAÇLI PROGRAMLAMADA BELİRSİZLİĞİ MODELLEMENİN DEĞERİ

Yiğit, Ece
Yüksek Lisans, Endüstri Mühendisliği Bölümü
Tez Yöneticisi: Dr. Öğr. Üyesi. Melih Çelik
Ortak Tez Yöneticisi : Dr. Öğr. Üyesi. Sakine Batun

Eylül 2019 , 74 sayfa

Bu çalışma ile, çok amaçlı problemlerde belirsizliği modellemenin önemi araştırılmıştır. Öncelikle, iki aşamalı çok amaçlı genel bir problemin matematiksel modeli sunulmuştur. Daha sonra, çok amaçlı iki aşamalı problemlerde stokastik çözümün değeri ve mükemmel bilginin beklenen değeri hesaplamaları için yeni bir yaklaşım sunulmuştur. Önerilen yöntemin geçerliliğini test etmek için, belirsizlik içeren iki amaçlı bir sırt çantası problemi üzerinde hesaplamalar yapılmıştır.

Anahtar Kelimeler: stokastik programlama, çok amaçlı programlama, stokastik çözümün değeri, mükemmel bilginin beklenen değeri

To my family

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my supervisors Melih Çelik and Sakine Batun for their invaluable guidance and encouragement throughout the course of this work and also in my undergraduate studies. Their contributions to not only my academic career but also my personal decisions are priceless. I am thankful for their continuous support.

I would like to thank my beloved family and friends who are always there for me. Without their never-ending love, I would not be the person I am today.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

CHAPTER 1

INTRODUCTION

Most of the real-world problems involve more than one objective and have uncertainty in their nature. In order to model uncertainty in such problems, stochastic programming can be used. Two-stage stochastic programming is one of the most preferred structures in stochastic programming. There are many applications of stochastic programming in real life, such as in scheduling, routing, workforce planning, path planning, and so on.

Even though multi-objective stochastic programming is well studied in the literature, there does not exist much work regarding the value of modeling uncertainty in such problems. Our main motivation to conduct this study is to propose and test measures that can be used to assess this value. The proposed measures are applied to a Two-Stage Multi-Objective Stochastic Knapsack Problem with two objectives. We first provide a description of the problem and the corresponding stochastical model, along with the models for the Mean Value Problem and the Wait-and-See Problem. Following this, we present the result of our computational experiments, mainly focusing on the Value of Stochastic Solution (VSS) and Expected Value of Perfect Information (EVPI) measures. We also analyze the validity of our measures by testing them under cases where the decision maker's preferences follow an underlying utility function.

The remainder of the thesis is organized as follows; in Chapter 2, preliminaries on stochastic programming and multi-objective optimization are provided. Chapter 3 provides a literature review regarding multi-objective stochastic programming. In Chapter 4, the proposed measures are described. In Chapter 5, the details of our computational experiments and the results are presented. Finally, in Chapter 6, the conclusions of the study and the summary of the insights are pointed out.

CHAPTER 2

PRELIMINARIES

In this chapter, we briefly present the preliminaries on stochastic programming and multi-objective optimization that are most relevant to this thesis. For more comprehensive information on stochastic programming, we refer to Birge and Louveaux (2011) and Kall and Wallace (1994), whereas for a more detailed information on multi-objective optimization, we refer to Ehrgott (2005), Rardin (1998) and Köksalan and Wallenius (2012).

2.1 Preliminaries on Stochastic Programming

Most of the real life decision making situations involve uncertainty. In order to model such problems in an accurate way, uncertainty in problem parameters should be considered carefully. Stochastic programming, which could be briefly defined as mathematical programming with random problem parameters, provides a framework to capture uncertainty when only imperfect information (rather than perfect information) about parameters is available through a probability distribution. The interest in stochastic programming has increased significantly due to the importance of uncertainty in and accessibility of parameters as a result of technological developments particularly in recent years. There are many applications of stochastic programming in real life, such as scheduling problems, routing problems, assignment problems and shortest path problems.

The availability of information over time is explicitly captured through stages in stochastic programs (SPs). A *two-stage stochastic program* (Beale (1955), Dantzig (1955)) is the simplest and most commonly used structure in stochastic programming.

There are two decision stages in a two-stage SP and the objective is to optimize a function which is composed of a first-stage component and the expected value of a second-stage component. The *first stage decisions* ($\mathbf{x}$) are made before the resolution of the uncertainty and the availability of the complete information on random parameters. Given the first-stage decisions, the *second-stage decisions* ($\mathbf{y}$) are made after the random scenario ($\omega \in \Omega$) is realized (i.e., uncertainty is resolved) and the values of random parameters ($\xi(\omega)$) become known (Beale (1955), Dantzig (1955)).

A general two-stage stochastic problem can be formulated in the *deterministic equivalent form* as follows:

$$
\begin{aligned}
(SP) \quad & \text{maximize } z = c^{\intercal}\mathbf{x} + \mathcal{Q}(\mathbf{x}) \\
& \text{subject to } A\mathbf{x} = b \\
& \qquad\qquad \mathbf{x} \in X
\end{aligned} \tag{2.1}
$$

where $\mathcal{Q}(\mathbf{x}) = E_{\xi}\big[Q(\mathbf{x}, \xi(\omega))\big]$ and

$$
\begin{aligned}
Q(\mathbf{x}, \xi(\omega)) = \quad & \underset{\mathbf{y}(\omega)}{\text{maximize}} \; q(\omega)^{\intercal}\mathbf{y}(\omega) \\
& \text{subject to } W(\omega)\mathbf{y}(\omega) + T(\omega)\mathbf{x} = h(\omega), \\
& \qquad\qquad \mathbf{y}(\omega) \in Y.
\end{aligned}
$$

In this formulation, $\mathbf{x}$ and $\mathbf{y}(\omega)$ are the first-stage and second-stage decision vectors, respectively, where ω represents the uncertainty. $\xi(\omega)$ is the vector of all random parameters, so is composed of the components of the second-stage objective coefficient vector $q(\omega)$, constraint coefficients for the second- and first-stage variables in the second stage, denoted by $W(\omega)$, $T(\omega)$, respectively, and the right-hand side vector for the second-stage constraints, represented by $h(\omega)$. The objective is to find the solution that maximizes the sum of the first-stage objective and the expected second-stage objective. Here, $E_{\xi}\big[Q(\mathbf{x}, \xi(\omega))\big]$ is the expected optimal objective function value of the second-stage problem given the first-stage decisions $\mathbf{x}$.

Each possible realization of ω (and hence $\xi(\omega)$) is called a *scenario*. In the case where the set of possible scenarios is discrete, letting $k \in K$ be the scenario index and π_k be the probability of scenario k, the above SP can represented in its extensive form as follows:

$$
\begin{aligned}
\text{(SP)} \quad & \text{maximize } z = c^{\mathsf{T}}\mathbf{x} + \sum_{k \in K} \pi_k q_k \mathbf{y_k} \\
& \text{subject to } A\mathbf{x} = b \\
& \qquad W_k \mathbf{y_k} + T_k \mathbf{x} = h_k \qquad \forall k, \\
& \qquad \mathbf{x} \in X, \\
& \qquad \mathbf{y_k} \in Y \qquad \forall k.
\end{aligned}
\tag{2.2}
$$

Under the availability of perfect information on the scenario (i.e., ω) and hence the problem parameters (i.e., $\xi(\omega)$) to be realized, the optimal solution could be found by solving the following deterministic version of the two-stage stochastic program for the known scenario ω.

$$
\begin{aligned}
\text{(WSP)} \quad & \text{maximize } z(\mathbf{x}, \xi(\omega)) = c^{\mathsf{T}}\mathbf{x}(\omega) + q(\omega)^{\mathsf{T}}\mathbf{y}(\omega) \\
& \text{subject to } A\mathbf{x}(\omega) = b, \\
& \qquad W(\omega)\mathbf{y}(\omega) + T(\omega)\mathbf{x}(\omega) = h(\omega), \\
& \qquad \mathbf{x}(\omega) \in X, \\
& \qquad \mathbf{y}(\omega) \in Y.
\end{aligned}
\tag{2.3}
$$

This problem is known as the *wait-and-see problem* (WSP) and the expected value of the wait-and-see solution can be represented as $z_{WS} = E_{\xi}[\max z(\mathbf{x}, \xi(\omega))]$.

Since the WSP is a relaxation of the SP, we have $z_{WS} \geq z_{SP}$, where z_{SP} represents the optimal objective function value of the SP. *Expected value of perfect information* (EVPI) is defined as $EVPI = z_{WS} - z_{SP}$ and can be interpreted as the amount by which having perfect information on the uncertain parameters would improve the expected optimal objective value of the stochastic problem. In other words, it is the maximum value that the decision maker would agree to pay in order to obtain perfect information.

The *expected value problem* (EVP) (*mean value problem*) is the following deterministic version of the stochastic problem where the parameter values are set as their expected values $(\tilde{W}, \tilde{T}, \tilde{h}, \tilde{q})$. The corresponding scenario is known as the *expected* *value scenario* (mean value scenario).

$$
\begin{aligned}
\text{(EVP)} \quad & \text{maximize } z(\mathbf{x}, \tilde{\xi}) = c^\intercal x + \tilde{q}^\intercal \mathbf{y} \\
& \text{subject to } A\mathbf{x} = b \\
& \qquad \tilde{W}\mathbf{y} + \tilde{T}\mathbf{x} = \tilde{h} \\
& \qquad \mathbf{x} \in X \\
& \qquad \mathbf{y} \in Y
\end{aligned}
\tag{2.4}
$$

Because of the number of possible scenarios and the complexity of the model itself, solving the SP is often computationally demanding and the corresponding EVP is solved instead in practice. The optimal solution of the EVP is known as the *expected value solution*. Letting $\mathbf{x}_{EV}$ denote the optimal solution of the EVP, the expected value of using $\mathbf{x}_{EV}$ can be represented as $z_{EVP} = E_\xi[\max z(\mathbf{x}_{EV}, \xi(\omega))]$.

The solution of the EVP is a solution to the SP, but not necessarily the optimal solution. Therefore, we have $z_{SP} \geq z_{EVP}$. As indicated by this relation, the expected objective value could be improved by by formulating and solving the SP rather than using the expected value solution. The difference is known as the *value of the stochastic solution* (VSS) and represented as $VSS = z_{SP} - z_{EVP}$. The VSS provides an insight about the importance of explicitly modeling the uncertainty in mathematical programming framework and solving the corresponding SP. The VSS can also be interpreted as the price of ignoring uncertainty in problem parameters.

2.2 Preliminaries on Multi-Objective Optimization

Multi-objective optimization is used when there exists more than one objective that the decision maker (DM) wants to optimize simultaneously. A general multi-objective program (MOP) with p objectives can be formulated as follows:

$$
\begin{aligned}
\text{(MOP)} \quad & \text{"maximize" } z(\mathbf{x}) = \left\{z^1(\mathbf{x}), z^2(\mathbf{x}), \ldots, z^p(\mathbf{x})\right\} \\
& \text{subject to } \mathbf{x} \in X
\end{aligned}
\tag{2.5}
$$

In the above formulation, $\mathbf{x}$ is the *decision vector* and $z(\mathbf{x})$ is the corresponding *objective vector*. The aim in multi-objective optimization is to generate a set of *efficient* *solutions*, which corresponds to a set of *non-dominated points* in the objective space, so that the DM can choose one among them according to his/her preferences.

A point $z(\mathbf{x})$ said to dominate the point $z(\tilde{\mathbf{x}})$ if $z^i(\mathbf{x}) \geq z^i(\tilde{\mathbf{x}})$ for all $i = 1, 2, \ldots, p$ and $z^i(\mathbf{x}) > z^i(\tilde{\mathbf{x}})$ for at least one i. If there does not exist such $x \in X$, then $\tilde{x}$ is said to be an efficient solution and $z(\tilde{\mathbf{x}})$ is a non-dominated point. A point $z(\overline{\mathbf{x}})$ in objective space is *strictly dominated* by another point $z(\mathbf{x})$ if and only if $z^i(\mathbf{x}) > z^i(\overline{\mathbf{x}})$ for all $i = 1, 2, \ldots, p$. If there does not exist such $\mathbf{x}$, then $z(\overline{\mathbf{x}})$ is said to be *weakly non-dominated*.

In multi-objective optimization, the point whose components are the best values of each objective is called the *ideal point* and represented as $z_{IP} = \{z^1_{IP}, z^2_{IP}, \ldots, z^p_{IP}\}$ where $z^i_{IP} = \max\limits_{\mathbf{x} \in E}\{z^i(\mathbf{x})\}$, and E denotes the set of all efficient solutions. The *nadir point* is the point whose components are the worst objective value components of the nondominated vectors and is denoted as $z_{NP} = \{z^1_{NP}, z^2_{NP}, \ldots, z^p_{NP}\}$ where $z^i_{NP} = \min\limits_{x \in E}\{z^i(\mathbf{x})\}$. The ideal and nadir points determine the range of the objective values.

One of the most known and used methods for solving multi-objective problems is ϵ-constraint method (Haimes et al., 1971). In ϵ-constraint method, multi-objective problems are converted to single objective sub-problems, where all objectives except one is treated as constraints. Since the main focus on this thesis study is bi-objective problems, we provide an ϵ-constraint method for such problems in Table 2.1.

Consider sub-problem of SP, denoted by SP(ϵ):

$$\begin{aligned} (\text{SP}(\epsilon)) \quad & \text{"maximize" } z^1(\mathbf{x}) + 0.00001 z^2(\mathbf{x}) \\ & \text{subject to } z^2(\mathbf{x}) \geq \epsilon \\ & \mathbf{x} \in X. \end{aligned} \tag{2.6}$$

Here, sub-problems are provided for SP only. The algorithm is applied to EVP and WSP also. It is shown that a similar method provided by Bérubé et al. (2009) can be used to find exact efficient solution set of bi-objective problems.

The function that can be used as a representation of the preferences of the DM on the values of each objective is called the *utility function* (or *value function* if there is no uncertainty), represented by $f(\mathbf{z})$. When the utility function is unknown, the set (or a

Table 2.1: Epsilon-Constraint Algorithm for Bi-Objective Stochastic Problems

Algorithm	
Step 0. (Initialization)	
	Compute ideal (z_{IP}) and nadir (z_{NP}) points of SP.
	Let E be the set of efficient solutions and N be the
	corresponding non-dominated points.
	Note that the points (z^1_{IP}, z^2_{NP}) and $(z^1_{NP}, z^2_{IP}) \in N$.
	Set $\epsilon = z^2_{NP}$.
Step 1. (Find a new point)	
	While $\epsilon < z^2_{IP}$ solve SP(ϵ) and add optimal solution $\mathbf{x}^*$
	to E and corresponding point on objective space $z(\mathbf{x}^*)$ to N.
	Set $\epsilon = z^2(\mathbf{x}^*) - \Delta$.
Step 2. (Stop)	

subset) of all efficient solutions can be generated and presented to the DM for further evaluation.

CHAPTER 3

LITERATURE REVIEW

Both multi-objective optimization and stochastic programming have extensive application areas and theoretical foundations in operations research. Consequently, these are very well-studied areas and a comprehensive literature review of these areas is well outside the scope of this thesis work. Reviews on the foundations and solution methods for multi-objective optimization include Köksalan and Wallenius (2012), Ehrgott and Gandibleux (2000) , Deb (2001) ,whereas Birge and Louveaux (2011), Kall and Wallace (1994) provide an extensive overview of the theory and solution approaches for stochastic programming.

Many real-life problems that can be tackled by operations research methods involve multiple objectives and uncertainty in their nature. In this chapter, we present a review of the relevant literature on studies that incorporate both of these aspects.

R. Caballero and Rey (2004) discuss a two-step transformation procedure in order to find efficient solutions of a stochastic multi-objective problem. These steps include transformation of the objectives to a single objective and establishing deterministic equivalent of stochastic problem. A widely-used method is to transform the stochastic program with multiple objectives into a single-objective version by means of scalarization in the objective function (e.g., by means of using weights for each objective) to reduce the problem to a single-objective version and solving the resulting stochastic program. Such a process is called the *stochastic approach* or *scalarizing method*. We do not consider such methods in our review, as these are essentially for single-objective models. A review of this stream of literature is provided in Ben Abdelaziz (2012).

Despite the wide range potential applications of multi-objective stochastic programming in areas such as manufacturing and production planning, supply chain management, humanitarian logistics, health care management and finance, the literature on studies applying the *multi-objective approach* or *non-scalarizing method*, i.e., those retaining the multi-objective structure in the stochastic program, is quite limited. In the remainder of this chapter, we classify those studies into two groups, namely studies that focus on providing solution approaches for multi-objective stochastic programming in general and those that address specific problems in different application areas.

3.1 Studies on the modeling and solution methods for multi-objective stochastic programming

A recent and extensive survey of the non-scalarizing modeling and solution approaches for multi-objective stochastic programming is given by J. Gutjahr and Pichler (2013). The paper provides the generic modeling approach and discusses the exact, approximate, and metaheuristic-based approaches for both risk-neutral and risk-averse cases. In doing so, the authors point to a number of computational issues. Despite the extensive coverage of solution methods, no discussion on the performance measurement aspects is provided.

A general-purpose algorithm called Adaptive Pareto-Sampling (APS) to determine the set of efficient solutions of two-objective optimization problems involving uncertainty is proposed by Gutjahr (2009). The algorithm is iterative and it combines the solution of corresponding deterministic two-objective problem with random sampling. Especially, the problems whose corresponding deterministic bi-objective problem can be formulated as a bi-objective integer linear problem are investigated; however, in theory, the algorithm can be applied to ordinary combinatorial problems. It is proven that, under mild conditions, the proposed solution set converges to the true set of Pareto-optimal solutions. Computational experiments are conducted on a stochastic bicriteria knapsack problem and a discussion on the runtime complexity of the approaches is also provided.

Norkin (2014) describes an approximation technique for solving multiobjective stochastic optimization problems of "input - random output" systems. An interactive multicriteria controlled random search method is analyzed, where the decision maker defines the region for new random search by considering visual analysis of previous iterations' non-dominated points set. In order to estimate performance indicators, parallel Monte Carlo simulations are used. On an insurance multicriteria optimization support system the technique is demonstrated. Convergence analysis is conducted and the conditions in order to assure the convergence with probability one under appropriate adjustment of sampling parameters are provided.

In his paper, Norkin (2017) extends the stochastic branch and bound method, which is used for solving scalar global and integer stochastic programming, to multiobjective stochastic problems, called stochastic vector branch and bound method. For sets of optimal values of subproblems, vector lower and upper bounds are established. Vector upper bounds are obtained by means of finding ideal point, and a feasible point's vector objective function is used as a lower bound. A general framework for both discrete and continuous optimization problems is provided and the convergence of those algorithms to set of approximate solutions are proved.

Whereas the focus in this thesis is not on the solution approaches on multi-objective stochastic programming, we make use of a part of these solution approaches (particularly the adaptive epsilon-constraint method) in the following chapters. Furthermore, none of the papers we discuss in this section propose any specific measures for evaluating the value of considering stochasticity or that of perfect information in their approaches, which is a gap we aim to bridge in this thesis.

3.2 Studies on addressing specific problems modeled by multi-objective stochastic programming

There exist a limited number of studies that apply the aforementioned solution approaches on specific problems in the operations research literature. In what follows, we provide a discussion of recent papers in this area.

Gutjahr and Reiter (2010) study a bi-objective two-stage stochastic portfolio selection problem. The uncertainty in such problems might arise from the risks related to a project and the amount of time needed to complete tasks of the project. The specific objectives considered in this study are the expected total economic and strategic gains and the expected total overtime cost. A procedure called adaptive Pareto sampling, previously developed by the authors, combined with the non-dominated sorting genetic algorithm II (NSGA-II) as an auxiliary procedure, is followed. For the estimation of expectations, importance sampling approach is used and a deterministic multi-objective problem is obtained. As a result of computational experiments, the authors conclude that the proposed methodology practically performs equally well compared to an approach including complete enumeration with extensive simulation. Furthermore, the run-time of the former is only 1% of the latter.

In their paper, Cardona-Valdes et al. (2011) study the design of a two-echelon distribution system where a number of candidate distribution centers, multiple customers, and production plants exist. In this supply chain, both the total cost and maximum service time is to be minimized simultaneously. The problem is formulated as a stochastic bi-objective mixed integer linear program where uncertainty in customer demand is modeled by scenarios. The solution approach uses the epsilon-constraint method, where a transformation of the stochastic bi-objective program into a single objective stochastic problem is made, followed by solving the deterministic equivalent of the stochastic program using the L-shaped method. The performance of the method is tested using randomly generated instances.

Fonseca et al. (2010) consider a bi-objective model for reverse logistics planning problem. They propose a two-stage stochastic bi-objective mixed-integer programming formulation where the strategic and operational decisions are made in the first and second stages, respectively. The objective is to minimize first-stage cost and expected second stage cost while minimizing the financial risk. The authors present the proposed stochastic model in its extensive form and test their model on a set of different scenarios using a case study based on data from the province of Cordoba in Spain. It is observed that non-dominated solutions to the problem on the case study can be obtained in a reasonable computational time by using an iterative method (Ross and Soland, 1980).

Tricoire et al. (2012) formulate a two-objective stochastic covering tour problem concerned with disaster management. The uncertainty of beneficiary demand in the population centers is modeled by random variables. The first objective in the model is to minimize the total cost of opening distribution centers and that of travel between centers. The second objective is to minimize the expected uncovered demand. In the first-stage, the decisions to open which distribution centers as well as the delivery tours are made. At the beginning of the second-stage, actual demands become known, and the actual supply values are determined. The authors use a branch-and-cut algorithm within an epsilon-constraint method in order to solve the resulting bi-objective two-stage stochastic program with recourse. While the deterministic single-objective Covering Tour Problem is NP-hard, the complexity of the problem is increased even further in bi-objective stochastic extension. The solution approach is tested on real-world data from rural communities in Senegal.

For solving stochastic multiobjective combinatorial optimization problems, a metaheuristics algorithm is proposed by Gannouni et al. (2017). The authors extend the main components of multi-objective evolutionary algorithms such as dominance criteria, elitism and diversification for the stochastic case. They establish their approach by using a hybridization of probabilistic programming with metaheuristic. The proposed algorithm is tested on a bi-objective stochastic vehicle routing problem and it is shown that the algorithm is able to generate a set of well-distributed probabilistic efficient solutions.

In the job-scheduling problem parameters such as the processing time of tasks, availability of resources, and deadlines of each stage are assumed to be known with certainty. However, these parameters might not be known precisely in real-life decision making problems. Therefore, stochastic programming is used in order to model such uncertainties.

Hao et al. (2015) study the bi-criteria stochastic job-shop scheduling problem where processing times are not known, and the objectives are to minimize expected average makespan and total tardiness. The authors propose a multi-objective estimation of distribution algorithm for this problem. First, the probability model of operation order is estimated. Using Monte-Carlo methods, sampling of processing time of operations are performed and then total tardiness of each sampling and the expected makespan are calculated. Using numerical experiments, the authors conclude that the algorithm obtains efficient solutions with acceptable schedule quality within a reasonable amount of computational time. Zehetner and J. Gutjahr (2018) assert that due to the complexity of the bi-objective stochastic covering tour problem, solving large instances using method of Tricoire et al. (2012) may not be possible.

Preserving the problem definition in Tricoire et al. (2012), Zehetner and J. Gutjahr (2018) use the NSGA-II algorithm to tackle the multi-objectivity of the model. Three different methods are considered to work around the stochastic aspect, namely a fixed random sample, a sample which is exchanged in each iteration, and APS. Experiments are conducted for the same test benchmarks as in Tricoire et al. (2012) and it is concluded that the NSGA-II-based solution and its computation time are always better than those of Tricoire et al. (2012).

To the best of our knowledge, the only study on multi-objective stochastic programming that involves performance measurement is by Rath et al. (2015), where the authors consider the location of humanitarian depots and the subsequent relief distribution under the uncertainty of the road network condition and with the objectives of maximizing expected demand coverage and minimizing budget use. The authors measure the VSS and the EVPI after transforming the multi-objective model to a single-objective one using the epsilon-constraint method. However, under such an approach, the measures are taken in terms of only one of the objectives. In this thesis work, we put forward alternative ways to measure the VSS and the EVPI, so that all objectives can be taken into account.The details of the VSS and EVPI measurement method by Rath et al. (2015) will be provided in the subsequent chapters of the thesis.

Rath et al. (2015) study a two-stage bi-objective stochastic programming for determining depot locations in disaster relief operations. In their study, they analyze several variants of the model. Using an adaptive Epsilon-constraint method, they obtain efficient solutions for the problem. When calculating the VSS and EVPI, they only consider improvement of objective function value in one objective, whereas, in our proposed method, both objectives are taken into account. They present average and maximum values of the VSS and EVPI. In our approach, we provide to the decision maker bounds of those values considering all of the objectives.

CHAPTER 4

VALUE OF THE STOCHASTIC SOLUTION AND THE EXPECTED VALUE OF PERFECT INFORMATION IN MULTI-OBJECTIVE STOCHASTIC PROGRAMMING

Many problems in practice involve both conflicting objectives and uncertain parameters. Our focus in this study is on proposing measures that can be used to assess the value of modeling uncertainty and having perfect information about how uncertainty will unfold in such problems. Using the same notation in Chapter 2, a multi-objective stochastic program (MOSP) with p objectives can be represented as follows:

$$\begin{aligned} \text{(MOSP)} \qquad & \text{"max"} \left\{ {c^1}^\top \mathbf{x} + Q^1(\mathbf{x}), ..., {c^p}^\top \mathbf{x} + Q^p(\mathbf{x}) \right\} \\ & \text{subject to } A\mathbf{x} = b \\ & \mathbf{x} \in X, \end{aligned} \tag{4.1}$$

where, for each objective $m = 1, 2, \ldots, p$, $Q^m(\mathbf{x}) = E_\xi \left[Q^m(\mathbf{x}, \xi(\omega)) \right]$ and

$$\begin{aligned} Q^m(\mathbf{x}, \xi(\omega)) = \qquad & \max_{\mathbf{y}(\omega)} q^m(\omega)^\top \mathbf{y}(\omega) \\ & \text{subject to } W^m(\omega)\mathbf{y}(\omega) + T^m(\omega)\mathbf{x} = h^m(\omega) \\ & \mathbf{y}(\omega) \ \in Y \end{aligned} \tag{4.2}$$

Although the value of modeling uncertainty and perfect information about uncertainty is well studied for single-objective problems (e.g., Birge and Louveaux (2011)), to the best of our knowledge, there is only a single study (Rath et al., 2015) regarding the calculation of VSS and EVPI of MOSPs in the literature. Rath et al. (2015) formulate depot location problem in disaster relief operations as a bi-objective stochastic program by incorporating the uncertainty in road accessibility through a discrete set of scenarios. To compute VSS for every solution on the non-dominated frontier, they generate a corresponding solution for the expected value problem by optimizing one of the objectives and treating the other one (which only involves a first-stage component) as a constraint. They compare the expected value of the objective (which is treated through the objective function) of these two solutions and refer to the difference as the VSS. In doing so, they define and compute a single VSS value for every point on the non-dominated frontier of the stochastic program. In other words, handling one of the objectives as a constraint, they use the same framework used for single-objective problems. They measure the EVPI value of the non-dominated points in a similar manner, using the wait-and-see problem instead of the expected value problem. For the problem instances considered in the study, they report the average and maximum values across the solutions on the non-dominated frontier. Using the results, they analyze areas of the objective space where it is advantageous to solve the stochastic problem, where the deterministic problem has solutions almost as good as stochastic model.

Using the VSS and EVPI calculation framework by Rath et al. (2015) ignores the multi-objective nature of the problems modeled by the MOSP, as the measurement is made from the perspective of only one of the objectives. Hence, there is need for a performance measurement framework that incorporates the existence of multiple objectives into the measurement process. Motivated by this, in this thesis, we aim to provide a new framework to estimate the VSS and EVPI in MOSPs where the impact of modeling uncertainty and having perfect information on uncertain parameters on each objective is explicitly considered. We assume that the relative importance of the objectives for the DM is unknown.

In the remainder of this chapter, three efficient frontiers are generated by solving the expected value problem, stochastic problem, and wait-and-see problem. Let E_{EEV}, E_{SP} and E_{WSP} represent the set of efficient solutions on these frontiers, respectively. The corresponding objective function values of these solutions generate three non-dominated frontiers, denoted by S^E_{EEV}, S^E_{SP} and S^E_{WSP}, respectively. In other words, the sets S^E_{EEV}, S^E_{SP} and S^E_{WSP} represent the non-dominated points of the EVP, SP and WSP, respectively.

Considering that the scales of objectives might be different, normalized objective values are used to calculate the VSS and EVPI. For an objective vector $z = (z^1, ..., z^p)$, the m^{th} (where $m = 1, ..., p$) component of the corresponding normalized objective vector $z_{norm} = (z^1_{norm}, ..., z^p_{norm})$ is found by:

$$z^m_{norm} = \frac{z^m - z^m_{worst}}{z^m_{best} - z^m_{worst}}$$

where z^m_{best} and z^m_{worst} are the best and worst values, respectively, of the m^{th} objective function values of the points in $E_{EEV} \cup E_{SP} \cup E_{WSP}$. Note that, the best value of the m^{th} objective is determined by the components of the ideal points of these three frontiers as:

$$z^m_{best} = \max \left\{ z^m_{EEV,IP}, z^m_{SP,IP}, z^m_{WSP,IP} \right\},$$

and the worst value is determined by the components of the nadir points as:

$$z^m_{worst} = \min \left\{ z^m_{EEV,NP}, z^m_{SP,NP}, z^m_{WSP,NP} \right\}.$$

4.1 VSS in Multi-Objective Stochastic Programming

For an optimization problem with a single objective, VSS is estimated by comparing the expected objective values of the solutions obtained by solving the EVP and SP. In the case of the MOSP, even if the DM indicates her most preferred non-dominated point of the EVP frontier, her preferred point on the SP frontier is still not certain. Consequently, calculation of the VSS is no longer straightforward. Considering the possible points that may be desirable for the DM on the SP frontier, we determine lower and upper bounds for the VSS, instead of a single scalar value. Furthermore, to limit the search, we only consider the points on the SP frontier that dominate the most preferred point on the EVP frontier. Using this idea, we extend the definition of the VSS of a non-dominated point on the EVP frontier to the VSS of the whole EVP frontier.

In what follows, we provide the mathematical basis for the calculation of these intervals.

4.1.1 VSS of an Efficient Solution

Assume that the DM's preferences are not available and she is interested in an efficient solution A on E_{EEV}. Let $D^E_{SP,A}$ be the set of efficient SP solutions that dominate A.

Without sacrificing from any of the objectives, by replacing A by a solution $B \in D^E_{SP,A}$, the DM can improve at least one of the objectives by

$$u(A,B) = \max_p \{z^p(B)_{norm} - z^p(A)_{norm}\},$$

which is the Tchebycheff distance between $z(A)$ and $z(B)$.

Which efficient solution the DM would pick among the ones in $D^E_{SP,A}$ to replace A by depends on her preferences. Therefore, the value of replacing A and hence the VSS for A, which we refer to as VSS_A hereafter, could actually be estimated as an interval rather than a single value. We calculate the lower and upper bounds (VSS^L_A and VSS^U_A) of this interval using the following equations:

$$VSS^L_A = \begin{cases} \min\limits_{x \in D^E_{SP,A}} \{u(A,x),\} & \text{if } D^E_{SP,A} \neq \emptyset, \\ 0, & \text{otherwise.} \end{cases}$$
$$= \begin{cases} \min\limits_{x \in D^E_{SP,A}} \{\max\limits_p \{z^p(x)_{norm} - z^p(A)_{norm}\}\}, & \text{if } D^E_{SP,A} \neq \emptyset, \\ 0, & \text{otherwise.} \end{cases}$$

$$VSS^U_A = \begin{cases} \max\limits_{x \in D^E_{SP,A}} \{u(A,x),\} & \text{if } D^E_{SP,A} \neq \emptyset, \\ 0, & \text{otherwise.} \end{cases}$$
$$= \begin{cases} \max\limits_{x \in D^E_{SP,A}} \{\max\limits_p \{z^p(x)_{norm} - z^p(A)_{norm}\}\}, & \text{if } D^E_{SP,A} \neq \emptyset, \\ 0, & \text{otherwise.} \end{cases}$$

Calculating those values for A, we are able to claim that the minimum and maximum improvement on either of the objectives (without sacrificing from the other ones) that can be obtained by formulating and solving the problem as a SP are VSS^L_A and VSS^U_A, respectively.

Note that, if $D^E_{SP,A} = \emptyset$, A cannot be improved in any of the objectives (without sacrificing from the other ones) by modeling the uncertainty, and hence is also an efficient solution to the SP. Accordingly, we have $VSS^L_A = VSS^U_A = 0$ by definition.

It should be noted here that the DM may prefer a point on the EVP frontier that may not dominate point A. Hence, the interval we provide for the VSS is an approximation, and the actual VSS value may be outside this interval. In the next chapter, we test the validity of our intervals by performing tests under various utility functions to observe how frequently the VSS is outside the determined interval, and the amount of under- or overestimation.

4.1.2 VSS of an Efficient Frontier

In the case where the DM's preference on the EVP efficient frontier is not available, one may be interested in calculating the VSS of the whole EVP efficient frontier as well. The lower (upper) bound of the VSS of an efficient frontier is determined by the smallest (largest) VSS value of the points on the frontier. Accordingly, we calculate lower and upper bounds on the VSS of an efficient frontier as follows:

$$VSS^L = \min_{x' \in E_{EEV}} \{VSS^L_{x'}\}$$

$$VSS^U = \max_{x' \in E_{EEV}} \{VSS^U_{x'}\}$$

Note that, the bounds we define on the VSS of an efficient frontier is preference-independent. In other words, by using this interval estimate, we are able to state the minimum and maximum gain that the DM can have by solving the SP (rather than EVP) regardless of her preferences.

Remark. If lower bound on the VSS of efficient frontier EEV is 0, that is $VSS^L = 0$, then there exists $A \in E_{EEV}$ and $B \in E_{SP}$, such that $z^p(B) = z^p(A)$ for all $m = 1, ..., p$.

4.2 EVPI in Multi-Objective Stochastic Programming

Our analysis for calculating the VSS of a point and the EVP efficient frontier can be extended to the case of the EVPI of a point on the SP frontier and the whole SP frontier.

4.2.1 EVPI of an Efficient Point

Let C be a point on the SP efficient frontier E_{SP} and $D^E_{WSP,C}$ be the set of efficient solutions of the WSP that dominate C. The DM's gain can be improved at least one of the objectives (while loss of any of the objectives is not allowed) by the Tchebycheff distance between $z(C)$ and $z(D)$, which is calculated by

$$u(C, D) = \max_p \big\{ z^p(D)_{norm} - z^p(C)_{norm} \big\}.$$

In accordance with her preferences, the DM chooses a solution from $D^E_{WSP,C}$ instead of efficient solution C. The value of replacing solution C should be provided as an interval, instead of a single value. The lower and upper bounds, denoted by $EVPI^L_C$ and $EVPI^U_C$, respectively, are calculated by equations below:

$$EVPI^L_C = \begin{cases} \min\limits_{x \in D^E_{WSP,C}} \big\{u(C, x)\big\}, & \text{if } D^E_{WSP,C} \neq \emptyset, \\ 0, & \text{otherwise.} \end{cases}$$
$$= \begin{cases} \min\limits_{x \in D^E_{WSP,C}} \big\{\max\limits_p \big\{z^p(x)_{norm} - z^p(C)_{norm}\big\}\big\}, & \text{if } D^E_{WSP,C} \neq \emptyset, \\ 0, & \text{otherwise.} \end{cases}$$

$$EVPI^U_C = \begin{cases} \max\limits_{x \in D^E_{WSP,C}} \big\{u(C, x)\big\}, & \text{if } D^E_{WSP,C} \neq \emptyset, \\ 0, & \text{otherwise.} \end{cases}$$
$$= \begin{cases} \max\limits_{x \in D^E_{WSP,C}} \big\{\max\limits_p \big\{z^p(x)_{norm} - z^p(C)_{norm}\big\}\big\}, & \text{if } D^E_{WSP,C} \neq \emptyset, \\ 0, & \text{otherwise.} \end{cases}$$

$EVPI^L_C$ and $EVPI^U_C$ are the least and the most improvement on any of the objectives (where sacrifice from the other objectives is forbidden) by modelling and solving the problem as a WSP, rather than the SP.

4.2.2 EVPI of an Efficient Frontier

After providing the DM all efficient solutions on the SP frontier, the DM chooses an alternative according to her preferences. In the case where the DM'spreference on the

SP frontier is not known, it is sensible to provide the DM bounds on EVPI which are calculated considering all efficient solutions of the SP.

An efficient frontier's lower and upper bounds on the EVPI can be calculated as follows:

$$EVPI^L = \min_{x' \in E_{SP}} \{EVPI^L_{x'}\}$$

$$EVPI^U = \max_{x' \in E_{SP}} \{EVPI^U_{x'}\}$$

Since the definitions of these bounds are independent from preferences of the DM, she can be informed about the minimum and maximum gain that can be obtained by solving the WSP, instead of the SP, without having information about her preferences.

4.3 Numerical Example

Suppose that the EVP, SP and WSP non-dominated frontiers of a bi-objective problem is obtained. Assume that the DM pays attention to points A, B, C, D and E. Let $A = (3365.0, 4072.0)$, $B = (3827.0, 3760.0)$ and $C = (3582.0, 3958.0)$ be points on S^E_{WSP} and $D = (3185.0, 3723.0)$ and $E = (3516.0, 3474.0)$ be points on S^E_{SP}, which are plotten in Figure 4.1. In calculation of bounds of EVPI of D, points A, B and

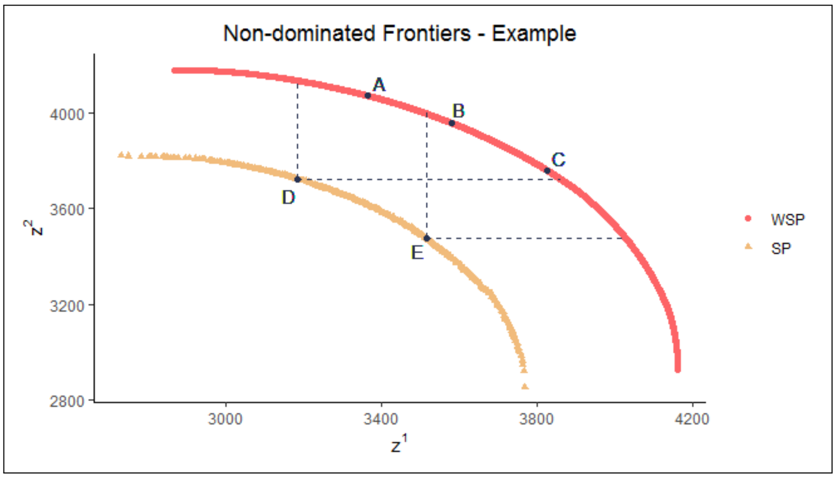

Figure 4.1: Calculation of EVPI - example instance.

C considered, where as, for point E, only points B and C are taking into account. Using below equations, lower and upper bounds on EVPI of points D and E can be found:

$$EVPI_D^L = \min_{x\in\{A,B,C\}} \{u(D,x)\} = \min_{x\in\{A,B,C\}} \{\max_{i=1,2}\{z^i(x)_{norm} - z^i(D)_{norm}\}\}$$
$$EVPI_D^U = \max_{x\in\{A,B,C\}} \{u(D,x)\} = \max_{x\in\{A,B,C\}} \{\max_{i=1,2}\{z^i(x)_{norm} - z^i(D)_{norm}\}\}$$

and

$$EVPI_E^L = \min_{x\in\{B,C\}} \{u(E,x)\} = \min_{x\in\{B,C\}} \{\max_{i=1,2}\{z^i(x)_{norm} - z^i(E)_{norm}\}\}$$
$$EVPI_E^U = \max_{x\in\{B,C\}} \{u(E,x)\} = \max_{x\in\{B,C\}} \{\max_{i=1,2}\{z^i(x)_{norm} - z^i(E)_{norm}\}\}$$

Let $F = (3271.6, 3676.4)$, $G = (3380.7, 3601.9)$ and $H = (3493.1, 3499.0)$ be points on S^E_{SP} and assume that the DM is interested in points $I = (3160.3, 3571.4)$, $J = (3333.4, 3446.8)$ and $K = (3390.7, 3393.7)$ on S^E_{EVP}, which are plotten in Figure 4.2. In order to calculate VSS of points I the points F and G are considered.

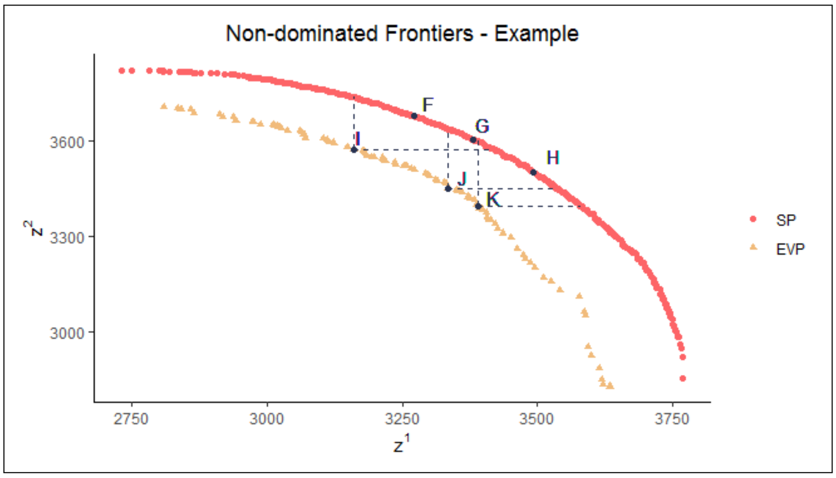

Figure 4.2: Calculation of VSS - example instance.

$$VSS_I^L = \min_{x\in\{F,G\}} \{u(I,x)\} = \min_{x\in\{F,G\}} \{\max_{i=1,2}\{z^i(x)_{norm} - z^i(I)_{norm}\}\}$$
$$VSS_I^U = \max_{x\in\{F,G\}} \{u(I,x)\} = \max_{x\in\{F,G\}} \{\max_{i=1,2}\{z^i(x)_{norm} - z^i(I)_{norm}\}\}$$

Likewise, the points G and H are taking into account to find VSS lower and upper bound of point J.

$$VSS_J^L = \min_{x\in\{G,H\}} \{u(J,x)\} = \min_{x\in\{G,H\}} \{\max_{i=1,2}\{z^i(x)_{norm} - z^i(J)_{norm}\}\}$$
$$VSS_J^U = \max_{x\in\{G,H\}} \{u(J,x)\} = \max_{x\in\{G,H\}} \{\max_{i=1,2}\{z^i(x)_{norm} - z^i(J)_{norm}\}\}$$

Since there is only one non-dominated point on S_{SP}^{E} (which is H) that the DM is interested in and dominates point K, VSS upper and lower bound of point K is found by below equation:

$$VSS_K^L = VSS_K^U = u(H, K)$$

CHAPTER 5

COMPUTATIONAL RESULTS

In this section, we apply the performance measures defined in the preceding section on a Two-Stage Multi-Objective Stochastic Knapsack Problem with two objectives (i.e., a bi-objective stochastic knapsack problem).

We first provide a description of the problem and the corresponding mathematical model, along with the Expected Value Problem and the Wait-and-See Problem. Following this, we present the results of our computational experiments, mainly focusing on an analysis of the VSS and EVPI measures.

In the problem we consider, the second-stage problem always has a feasible solution for any feasible first stage decision vector. In the literature, these problems are called two-stage stochastic problems with relatively complete recourse. By working with such problems, we can ensure that the corresponding Expected Value Problem is always feasible, and thus the VSS measures can be readily obtained. In cases where the Expected Value Problem is infeasible, we may assume the VSS to be equal to ∞.

5.1 The Two-Stage Multi-Objective Knapsack Problem

The general knapsack problem is a well-known combinatorial optimization problem to model transportation, scheduling, production and network optimization problems.

In the deterministic knapsack problem, given a set of items, each with a weight and a reward, the objective is to find a subset, such that total reward of the subset is maximized, while its total weight does not exceed certain knapsack capacity.

In many real-life applications of the Knapsack Problem, the "weights" of the "items" may not be known in advance with certainty. An example application may be in project selection, where the costs and/or durations of the projects to be undertaken in sequence are not known in advance, whereas there is a fixed budget and/or time limit. Such applications motivate the modeling of the multi-objective and stochastic version of the Knapsack Problem (the MOSKP).

The single-objective version of the Stochastic Knapsack Problem we consider is based on Kosuch (2014) and consists of deterministic rewards r_i and stochastic weights w_i for each item i in a set I of available items, subject to a deterministic knapsack capacity C. Once the item selection is made in the first stage, item weights become known, and a penalty is incurred for each unit of exceeded capacity (overage). Without loss of generality, we assume a single unit of penalty for each unit of overage.

The first-stage decisions for the two-stage stochastic programming formulation are denoted by vector $\mathbf{x} = \{x_1, x_2, ..., x_{|I|}\}$, where binary decision variable x_i, $i \in I$ takes value 1 if the item is selected in the knapsack, and takes value 0 otherwise. The objective is to maximize expected net profit, which is given by the difference between the expected total reward of selected items and expected overage. The sets, parameters, and decision variables we use in the two-stage stochastic programming formulation are as follows:

Sets
I: items
Ω: scenarios

Parameters
r_i: reward of item $i \in I$
w_{ik}: weight of item i in scenario $k \in \Omega$
C: capacity

Decision Variables

$$x_i = \begin{cases} 1, & \text{if item } i \in I \text{ is selected,} \\ 0, & \text{otherwise.} \end{cases}$$

y_k : capacity overage in scenario $k \in \Omega$

Based on these definitions, the single-objective version of the problem can be stated as below:

$$\begin{aligned} \max_{\mathbf{x}} \quad & \sum_{i \in I} r_i x_i - E[h(\mathbf{x}, k)] \\ \text{subject to} \quad & x_i \in \{0, 1\} \; \forall \, i \in I \end{aligned} \tag{5.1}$$

where

$$\begin{aligned} h(\mathbf{x}, k) = \quad & \min y_k \\ & \text{subject to} \sum_{i \in I} w_{ik} x_i \leq C + y_k \\ & y_k \geq 0. \end{aligned}$$

In the existence of multiple objectives, the the MOSKP can then be modeled using multiple rewards for each item and multiple capacities for each knapsack in each objective. For the project selection example, these corresponds to different project costs and durations, and separate financial budget and makespan deadline, respectively.

The additional parameters for the the MOSKP model are as follows:
r_i^m: reward of item $i \in I$ in objective $m = 1, 2$.
C^m: capacity of knapsack for objective $m = 1, 2$.

$$\begin{aligned} \text{"}\max_{\mathbf{x}}\text{"} \quad & \left\{ \sum_{i \in I} r_i^1 x_i - E[h^1(\mathbf{x}, k)], \sum_{i \in I} r_i^2 x_i - E[h^2(\mathbf{x}, k)] \right\} \\ \text{subject to} \quad & x_i \in \{0, 1\} \; \forall \, i \in I \end{aligned}$$

where for $m = 1, 2$,

$$\begin{aligned} h^m(\mathbf{x}, k) = \quad & \min_{y} y_k \\ & \text{subject to} \sum_{i \in I} w_{ik} x_i \leq C^m + y_k \\ & y_k \geq 0. \end{aligned}$$

The corresponding Expected Value Problem for the the MOSKP is given as follows:

$$\begin{aligned} \text{"}\max_{\mathbf{x}}\text{"} \quad & \left\{ \sum_{i \in I} r_i^1 x_i - E[g^1(\mathbf{x})], \sum_{i \in I} r_i^2 x_i - E[g^2(\mathbf{x})] \right\} \\ \text{subject to} \quad & x_i \in \{0,1\} \; \forall \, i \in I \end{aligned} \tag{5.2}$$

where for $m = 1, 2$,

$$\begin{aligned} g^m(\mathbf{x}) = \quad & \min \tilde{y} \\ & \text{subject to} \sum_{i \in I} \tilde{w}_i x_i \leq C^m + \tilde{y}, \\ & \tilde{y} \geq 0. \end{aligned}$$

where $\tilde{w}_i = E_k[w_{ik}]$ and let $\tilde{x}_i$ be the optimal first-stage decision variables of above problem. The EVP is found by solving:

$$\text{"}\max_{\mathbf{x}}\text{"} \quad \left\{ \sum_{i \in I} r_i^1 \tilde{x}_i - E[h^1(\mathbf{x}, k)], \sum_{i \in I} r_i^2 \tilde{x}_i - E[h^2(\mathbf{x}, k)] \right\} \tag{5.3}$$

where for $m = 1, 2$,

$$\begin{aligned} h^m(\mathbf{x}, k) = \quad & \min_{\mathbf{y}} y_k \\ & \text{subject to} \sum_{i \in I} w_{ik} \tilde{x}_i \leq C^m + y_k, \\ & y_k \geq 0. \end{aligned}$$

The corresponding WSP for the MOSKP can be constructed by defining a first-stage decision variable for each scenario and taking an expectation over the net profits for each scenario.

$\mathbf{x}_k$: the first-stage decision vector in scenario $k \in \Omega$.

$$\begin{aligned} \text{"}\max_{\mathbf{x}}\text{"} \quad & \left\{ E\left[\sum_{i \in I} r_i^1 x_{ik} - h^1(\mathbf{x}, k) \right], E\left[\sum_{i \in I} r_i^2 x_{ik} - h^2(\mathbf{x}, k) \right] \right\} \\ \text{subject to} \quad & x_{ik} \in \{0,1\} \; \forall \, i \in I \end{aligned} \tag{5.4}$$

where for $m = 1, 2.$,

$$\begin{aligned} h^m(\mathbf{x}, k) \quad & = \min_{\mathbf{y}} y_k \\ & \text{subject to} \sum_{i \in I} w_{ik} x_{ik} \leq C^m + y_k, \\ & y_k \geq 0. \end{aligned}$$

5.2 Analysis of the VSS and EVPI Values

5.2.1 Numerical Example

We consider a two-stage bi-objective knapsack problem where $|I| = 30$ and $|\Omega| = 2$. The weight and reward parameters for each item are generated as integer from a discrete uniform distribution in the interval [1,100]. The capacities are calculated as $\frac{\sum_{(i\in I)}\sum_{(k\in\Omega)} w_{ik}^m}{2|\Omega|}$ for each objective function $m = 1, 2$.

In order to solve corresponding EVP, the parameter values are set as their expected values. First, model 5.2 is solved and an efficient solution $(\mathbf{x}^*)$ is obtained. Next, fixing $(\mathbf{x})$ values to $(\mathbf{x}^*)$ in model 5.3, efficient solutions to SP are attained. Likewise, the model 5.4 is used in order to obtain the non-dominated points of WSP.

Using the epsilon-constraing method provided in Chapter 2, the non-dominated efficient frontiers are generated. The number of efficient solutions for the EVP, SP and WSP are 42, 75 and 193, respectively.

The EEV, SP, and WSP non-dominated frontiers for example problem are shown in Figure 5.1.

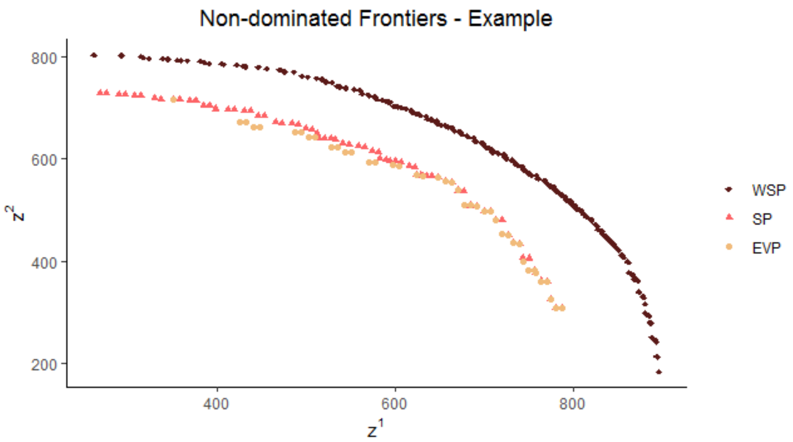

Figure 5.1: Non-dominated frontiers of example problem.

The upper and lower VSS bounds for the example instance are plotted in Figure 5.2. Please note that the points on Figure 5.2 correspond to each solution on S_{EEV}^E. Along the x-axis, z^1 values of efficient solutions decrease as z^2 values increase. The last point on the figure is (z^1_{NP}, z^2_{IP}).

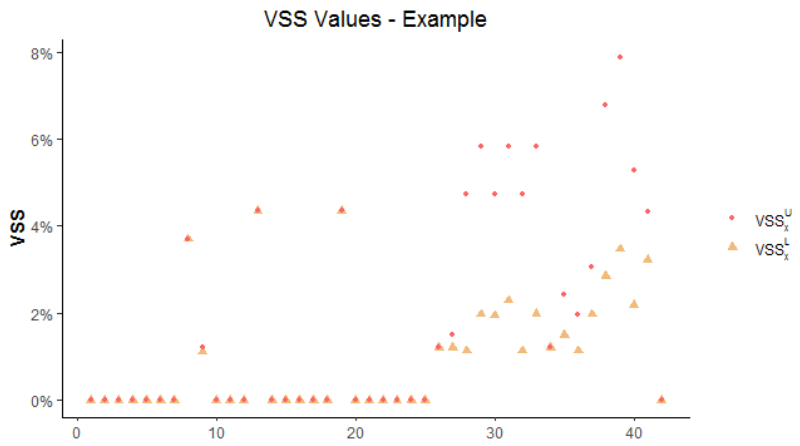

Figure 5.2: VSS upper and lower bounds of example problem.

For this example instance, it is observed that for most of the non-dominated points with high z^1 values, lower and upper bounds of VSS are 0. However, lower and upper bounds of VSS tend to take positive values when z^2 values are high.

Figure 5.3 shows the lower and upper bounds of EVPI values of the example instance. As in the case of VSS, the x-axis corresponds to solutions of S^E_{SP}.

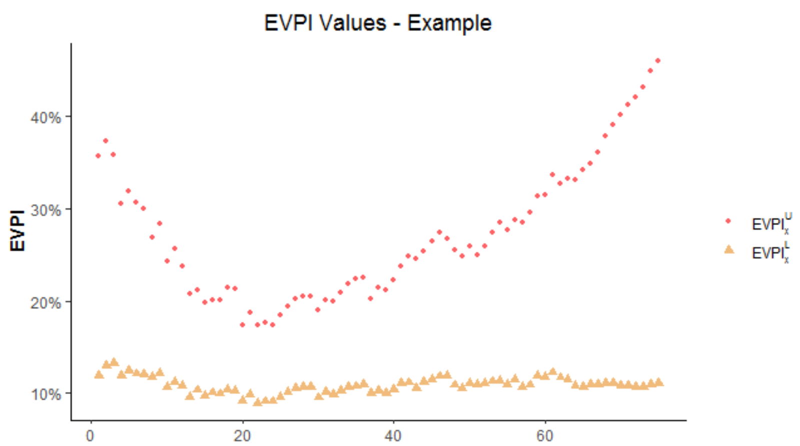

Figure 5.3: EVPI upper and lower bounds of example instance.

5.2.2 Results of Experiments

In this section, the analysis of the MOSKP is provided. Instance parameters are presented first, followed by a discussion of the results of the analysis.

Using the epsilon constraint method provided in Chapter 2, all non-dominated frontiers for EEV, SP and WSP are obtained for each instance. We coded the algorithm in GAMS using and using CPLEX 12.4.0.1 as the solver. We executed our experiments on a on 64-bit Microsoft 10 Enterprise installed Intel(R) Core (TM) i7-4770S CPU @ 3.10 GHz computer with 16.00 GB RAM.

The frontier graphs for each instance are plotted using the objective vector of every efficient solution obtained to the corresponding EVP, SP and WSP. In addition, the VSS and EVPI intervals of these solutions are also provided. The objective function values (z^1, z^2) are plotted in non-dominated frontiers. However, normalized values of the objective function values are used in the calculation of the VSS and EVPI. In each of these plots, the x-axis corresponds to each solution on S^E_{EEV} and S^E_{SP}, respectively. The first solution in each plot attains the maximum value of the first objective (z^1 value), whereas the last solution has maximum z^2 value. Please note that the number of solutions on the VSS and EVPI graphs are equal to number of efficient solutions obtained by solving the EVP and SP, respectively.

In the figures provided, some portions of the plots may appear as flat lines, which is due to the scaling of the graph and does not necessarily indicate that the value remains constant.

We conduct experiments on the MOSKP with 100 nodes and 10 scenarios. The rewards of items for 10 instances are retrieved from input files available at Köksalan (2016). From the parameters given for 100 nodes and three objectives, first two coefficients of objectives are used. The weight parameter for each item is generated as an integer value from a discrete uniform distribution in the interval [1,100]. The capacities are calculated as $\frac{\sum_{(i \in I)} \sum_{(k \in \Omega)} w^m_{ik}}{2|\Omega|}$ for each objective function $m = 1, 2$.

The number of efficient solutions generated for each instance is provided in Table 5.1.

Table 5.1: The number of efficient solutions for all instances for the MOSKP.

	EVP	**SP**	**WSP**
Instance 1	206	450	2357
Instance 2	103	471	2010
Instance 3	92	510	2433
Instance 4	72	356	2102
Instance 5	164	380	2323
Instance 6	429	598	2624
Instance 7	96	572	2518
Instance 8	308	495	2305
Instance 9	53	406	2128
Instance 10	446	465	2731

The number of efficient solutions of the WSP is higher than that of the SP. Likewise, the number of efficient solutions of the SP is higher than that of the EVP. Over all instances, the number of efficient solutions on any frontier ranges from 53 to 2731.

The computational time (in CPU seconds) to obtain the efficient solutions of the EVP, SP and WSP for the MOSKP are provided in Table 5.2. It is observed that SP frontier is generated faster than both EVP and WSP frontiers.. This may seem counterintuitive, as the EVP and WSP frontiers are the results of deterministic models, which are expected to run faster than the SP. The reason behind this discrepancy is due to the fact that obtaining the efficient solutions for the EVP requires the solutions of two different models, as opposed to a single model for the SP. The solutions are obtained by solving the EVP and their actual objective values are obtained by solving a modified version of the SP that includes the EVP solution as a constraint on the first-stage decision variables. Moreover, the computational time reported for generating the efficient solutions for the EVP also includes the time spent on eliminating the dominated solutions based on their actual objective values.

Although the efficient frontier of the WSP is generated by solving a deterministic model, the model needs to be solved for every scenario, which increases the required computational effort. Moreover, the number of efficient solutions obtained for the WSP is significantly higher since its feasible region is larger than that of the SP as the nonanticipativity constraints are relaxed in WSP (i.e., the first-stage variables can take different values in different scenarios). This also affects the computational time.

Table 5.2: The computational time (in CPU seconds) to obtain efficient solutions of the EVP, SP and WSP for the MOSKP.

	EVP	**SP**	**WSP**
Instance 1	316.68	106.70	1752.42
Instance 2	125.68	167.39	1442.41
Instance 3	187.78	163.00	2033.85
Instance 4	106.11	60.803	1400.34
Instance 5	100.88	59.29	1564.432
Instance 6	171.70	94.97	1846.97
Instance 7	161.68	89.39	1493.08
Instance 8	132.70	78.08	1468.58
Instance 9	97.86	68.538	1765.59
Instance 10	123.03	72.90	1524.32

It should be noted that, the solutions obtained by solving the WSP problem is not implementable since they do not satisfy the nonanticipativity constraint. They are obtained not for practical purposes but for estimating the EVPI.

The EVPI values for all instances for the MOSKP is provided in Table 5.3. As can be observed from Table 5.3, the EVPI values range between 15.78% (Instance 1 and Instance 6) and 72.46% (Instance 2), which indicates that the DM can benefit from a more accurate estimation of the random problem parameters. For a solution on the SP frontier, the width of the lower bound - upper bound interval (i.e., the range) for the EVPI is less than 24.88% (Instance 2) on average. Lower width values indicate narrower intervals and hence more accurate estimates for the EVPI.

The VSS values for all instances for the MOSKP is provided in Table 5.4. The VSS values range between 0% (indicated by the lower bound for many instances) and 19.32% (Instance 3). The low average range values indicate the high accuracy of our VSS estimates for most of the instances.

To detail our findings on the EVPI and VSS values, non-dominated frontiers, EVPI, and VSS plots for Instance 7 are given in Figures 5.4, 5.5 and 5.6, respectively. The graphs for the remaining instances are provided in Appendix A.

Unlike in other non-dominated frontier graphs provided in Appendix A, a remark-

Table 5.3: The EVPI values for all instances of the MOSKP.

	Lower Bound			**Upper Bound**			**Range**		
	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max
Instance 1	15.78%	16.96%	21.82%	28.82%	37.85%	64.16%	12.82%	20.89%	42.45%
Instance 2	21.07%	22.04%	25.57%	37.40%	46.92%	72.46%	16.04%	24.88%	46.89%
Instance 3	17.38%	18.61%	19.43%	34.00%	42.35%	60.20%	15.18%	23.74%	41.17%
Instance 4	17.66%	19.05%	24.71%	33.27%	43.00%	70.45%	14.27%	23.95%	45.74%
Instance 5	15.99%	18.14%	22.31%	32.10%	39.35%	62.45%	14.51%	21.21%	40.17%
Instance 6	15.78%	16.96%	21.82%	28.82%	37.85%	64.16%	12.82%	20.89%	42.45%
Instance 7	18.73%	19.94%	20.56%	35.99%	43.30%	62.15%	15.76%	23.36%	42.00%
Instance 8	17.27%	17.65%	19.37%	30.97%	40.34%	65.58%	13.52%	22.69%	46.21%
Instance 9	18.97%	19.53%	22.93%	33.69%	43.11%	70.89%	14.47%	23.58%	47.96%
Instance 10	15.85%	19.42%	23.32%	35.48%	42.11%	65.95%	15.60%	22.69%	42.63%

Table 5.4: The VSS values for all instances of the MOSKP.

	Lower Bound			**Upper Bound**			**Range**		
	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max
Instance 1	0.00%	0.03%	0.62%	0.00%	0.07%	1.59%	0.00%	0.04%	1.15%
Instance 2	0.00%	0.80%	2.98%	0.00%	2.54%	14.95%	0.00%	1.75%	12.01%
Instance 3	0.00%	0.62%	4.09%	0.00%	1.72%	19.32%	0.00%	1.10%	15.23%
Instance 4	0.00%	0.67%	1.89%	0.00%	1.55%	4.49%	0.00%	0.88%	3.24%
Instance 5	0.00%	0.09%	1.58%	0.00%	0.29%	5.67%	0.00%	0.20%	4.09%
Instance 6	0.00%	0.01%	0.72%	0.00%	0.03%	2.80%	0.00%	0.02%	2.34%
Instance 7	1.20%	4.04%	5.43%	4.62%	9.36%	16.05%	3.24%	5.32%	10.94%
Instance 8	0.00%	0.04%	1.14%	0.00%	0.15%	5.98%	0.00%	0.10%	4.98%
Instance 9	0.41%	1.60%	2.51%	0.69%	4.29%	9.74%	0.28%	2.69%	8.01%
Instance 10	0.00%	0.00%	1.00%	0.00%	0.01%	3.53%	0.00%	0.01%	2.53%

able distance from EVP curve to SP curve is observed in Instance 7. The lower and upper EVPI bounds of non-dominated SP frontier are $EVPI^L = 18.73\%$ and $EVPI^U = 62.15\%$. The solution with maximum EVPI upper bound value is the first solution on the x-axis. EVPI upper bound reaches its maximum where the difference between the first and the second objective function values of the efficient solutions on SP are high (i.e., in the extreme portions of the non-dominated frontier), whereas,

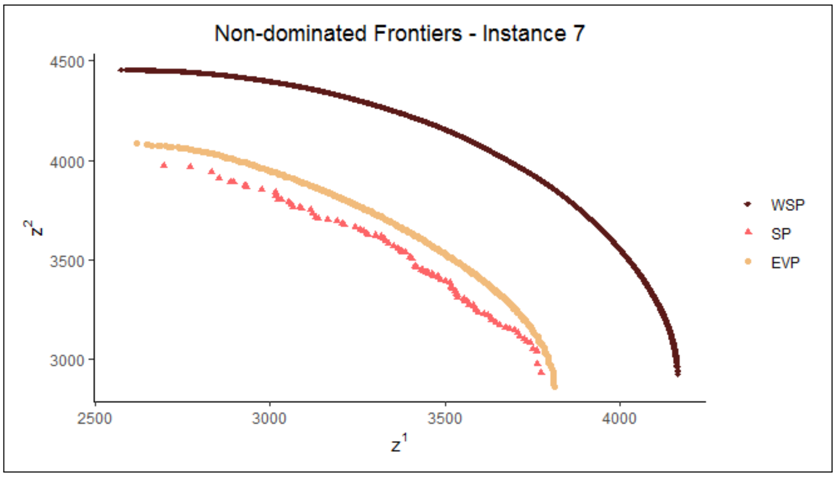

Figure 5.4: Non-dominated frontiers of the MOSKP - Instance 7.

EVPI upper bound reaches its minimum where the difference is low (i.e., in the middle portion of the non-dominated frontier). In the extreme portions of the efficient frontier, the emphasis on one of the objectives is significantly higher. Therefore, the EVPI at the extreme portions is mostly due to the uncertainty in one of the objectives. More explicitly, in the extreme portion of the non-dominated frontier with lower z^1 values, EVPI is mostly attributable to the uncertainty related to z^2. EVPI is generated mainly due to the uncertainty in z^1 in the other extreme portion of the frontier. In the middle portion, we observe lower EVPI values due to relatively close importance levels of the objectives. When the problem is solved for each scenario, the solution that optimizes z^1 does not necessarily optimizes z^2 as well. Therefore, availability of perfect information does not improve the solution in this middle portion as much as it improves the solutions in the extreme portions.

Figure 5.6 represents lower and upper bounds on VSS of non-dominated EEV frontier of Instance 7.

As can be seen in Figure 5.6, the lower and upper bounds on VSS are 1.20% and 16.05% for Instance 7. Since $VSS^L \neq 0$, it is guaranteed that solving SP instead of EVP would yield higher objective function values.

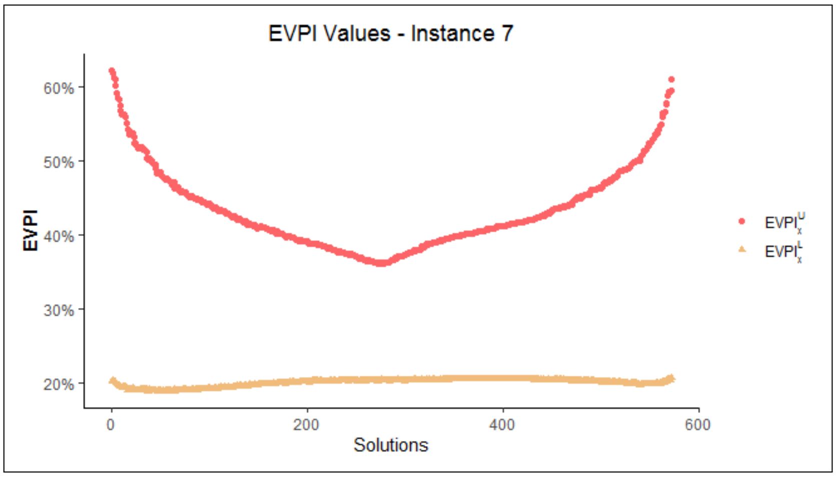

Figure 5.5: EVPI values of the MOSKP - Instance 7.

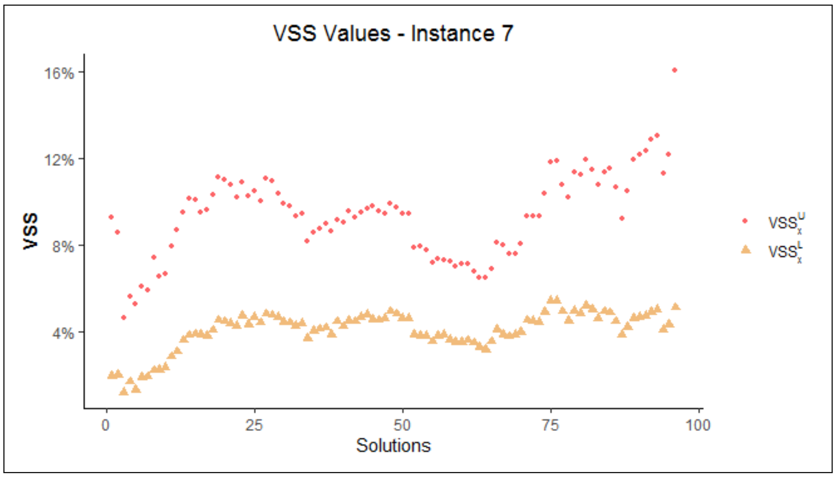

Figure 5.6: VSS values of the MOSKP - Instance 7.

5.3 Utility Function Analysis

In the preceding section, our definition of the VSS and EVPI lower and upper bounds depends on the condition that no sacrifice is made from any objectives while calculating these. Hence, for the VSS (EVPI), we only check points on the SP (WSP) efficient frontier that dominate a given point on the EEV (SP) efficient frontier. On the other hand, it is possible that a decision maker that prefers a given point on the EEV (SP) efficient frontier may prefer one on the SP (WSP) efficient frontier that does not dominate the former point. In other words, the decision maker may prefer to balance the trade-off between the two objectives in a way that she improves some of

Table 5.5: The weights used in simulating DM's value function.

w1	0.98	0.02
w2	0.90	0.10
w3	0.80	0.20
w4	0.70	0.30
w5	0.60	0.40
w6	0.50	0.50
w7	0.40	0.60
w8	0.30	0.70
w9	0.20	0.80
w10	0.10	0.90
w11	0.02	0.98

the objectives while sacrificing from the other ones. In such a case, the Tchebycheff distance between the solutions chosen by the decision maker on the EEV frontier and SP frontier may be larger than the upper bound defined on the VSS, which implies that the upper bound for the actual VSS may not be valid for this decision maker.

When the utility function of the decision maker is known, it is possible to identify whether the preferred points of the decision maker on each frontier. In this part, for both VSS and EVPI, we compare the utilities of the points we use in our VSS and EVPI calculations to those that yield the highest utility. If our points coincide with those that maximize the utility on each frontier, the differences will be zero, which is the most desirable case. However, when the points are different, we would like the difference to be as small as possible.

We use the following set of utility functions from Lokman et al. (2016). The weight combinations are given in Table 5.5.

$$\text{Linear: max} \sum_{i=1}^{p} w_i z_i$$

$$\text{Quadratic: max} \sum_{i=1}^{p} -w_i^2 \big(z_i - z_i^{IP}\big)^2$$

$$\text{Tchebycheff: max } \big\{\min_{i=1,\dots,p}\big(w_i(z_i - z_i^{IP})\big)\big\}$$

5.3.1 VSS when the Utility Function of the Decision Maker is Known

Suppose that it is possible to represent the DM's underlying preferences by a utility function, $f(\mathbf{z})$. The values of $f(\mathbf{z})$ for all $z \in S^E_{EEV}$ and $z \in S^E_{SP}$ are calculated. Let EEV_* and SP_* be the most preferred solution of the DM from the EEV and SP frontiers, respectively. More formally,

$$f(EEV_*) = \underset{z \in S^E_{EEV}}{max}\ f(\mathbf{z})$$

and

$$f(SP_*) = \underset{z \in S^E_{SP}}{max} f(\mathbf{z}).$$

In the first approach, if the point SP_* dominates the point EEV_*, the VSS is computed as below:

$$VSS = \max\{SP^1_* - EEV^1_*, SP^2_* - EEV^2_*\}$$

The VSS cannot be computed if EEV_* is not dominated by the point SP_* because it violates the dominance restriction in the definition of VSS.

In the second approach, in finding the point maximizing utility function on the efficient frontier of SP, we only consider the points that dominate the point EEV_* . Let D_{EEV_*} be all the points on SP efficient frontier that dominate EEV_*. For every point $z \in D_{EEV_*}$, $f(\mathbf{z})$ is calculated. Let $SP_{EEV_*} \in D_{EEV_*}$ be the most preferred solution of the DM in this approach, where

$$f(SP_{EEV_*}) = \underset{z \in D_{EEV*}}{max}\ f(\mathbf{z}).$$

Then, we calculate the VSS using the following equation:

$$VSS = \max\{SP^1_{EEV_*} - EEV^1_*, SP^2_{EEV_*} - EEV^2_*\}$$

5.4 EVPI when the Utility Function of the Decision Maker is Known

Assuming DM's value function $f(\mathbf{z})$ is known or can be simulated. First, the values of $f(\mathbf{z})$ for all $z \in S^E_{SP}$ and $z \in S^E_{WSP}$ are calculated and the most preferred solutions that maximizes f are denoted by SP_*, WSP_*, respectively.

$$f(SP_*) = \underset{z \in S^E_{SP}}{max} f(\mathbf{z})$$

and

$$f(WSP_*) = \underset{z \in S^E_{WSP}}{max} f(\mathbf{z}).$$

In the first approach, EVPI is calculated as below if the point SP_* is dominated by the point WSP_*.

$$EVPI = \max \{WSP^1_{SP_*} - SP^1_*, WSP^2_{SP_*} - SP^2_*\}.$$

If SP_* is not dominated by the point WSP_*, the EVPI is cannot be calculated.

In the second approach, after finding SP_* by using above definition, only points considered in calculation of EVPI are the ones that dominate SP_*. Let D_{SP_*} to be the set of all points on S^E_{WSP} dominating SP_*. Then, we find $WSP_{SP_*} \in D_{SP_*}$ that maximizes f.

$$f(WSP_*) = \underset{z \in D_{SP_*}}{max} f(\mathbf{z})$$

and in the second approach EVPI is found by

$$EVPI = \max \{WSP^1_{SP_*} - SP^1_*, WSP^2_{SP_*} - SP^2_*\}.$$

5.4.1 Numerical Example

Let us consider the same example given in Section 4.3. Assume that the DM's preferences are consistent with an underlying utility function which maximizes a weighted linear combination of objective function values using the weight vector $\mathbf{w_5} = (0.6, 0.4)$, that is, $f(z) = \max 0.6z^1 + 0.4z^2$.

First, the utility function values of solutions on the EVP, SP and WSP frontiers are calculated. The EVP solution that maximizes $f(z)$ is $\mathbf{z}_{EVP*} = (707.0, 497.0)$. In the first approach, the dominance relation is relaxed. Therefore, a solution on the SP frontier that maximizes her utility function is of concern. The point on the SP frontier with the maximum $f(z)$ value is $\mathbf{z}_{SP_1} = (720.0, 479.5)$.

In the second approach, we consider the points on the SP frontier that dominate $\mathbf{z}_{EVP*}$. There does not exist any solution on SP frontier that dominates $\mathbf{z}_{EVP*}$, that is, $D^E_{SP,z_{EVP*}} = \emptyset$, and $\mathbf{z}_{EVP*}$ is also an efficient solution of SP. Therefore, the non-dominated point on SP frontier that maximizes the utility function is $\mathbf{z}_{SP_2} = (707.0, 497.0)$.

The utility loss from the dominance restriction when calculating the VSS is evaluated by $f(\mathbf{z}_{SP_1}) - f(\mathbf{z}_{SP_2})$.

For this utility function, the utility loss in percentage when calculating the VSS is $\frac{(623.8-623.0)}{623.8} = 0.13\%$. The utility loss in percentage when calculating the VSS for all weight vectors and functions are provided in the Table 5.6. The utility loss is very low (less than 1.60%) for the linear utility function with different weight values. For the quadratic and the Tchebycheff utility function, the loss can be significant for some weight values.

Table 5.6: The utility loss in percentage when calculating the VSS for the example instance

	First Approach			**Second Approach**			**Difference**		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	777.39	-71.40	508.13	777.39	-71.40	427.28	0.00%	0.00%	15.91%
w2	738.95	-1585.83	466.65	738.95	-1585.83	392.40	0.00%	0.00%	15.91%
w3	690.90	-4894.76	414.80	690.90	-4894.76	348.80	0.00%	0.00%	15.91%
w4	648.20	-7802.13	362.95	648.20	-7980.16	305.20	0.00%	2.28%	15.91%
w5	623.80	-10254.24	311.10	623.00	-10318.48	261.60	0.13%	0.63%	15.91%
w6	609.25	-11394.81	259.25	609.25	-11394.81	218.00	0.00%	0.00%	15.91%
w7	600.50	-11592.25	253.50	598.30	-12810.85	253.50	0.37%	10.51%	0.00%
w8	617.05	-9551.31	295.75	607.20	-9903.17	295.75	1.60%	3.68%	0.00%
w9	646.00	-5670.01	338.00	643.80	-6373.45	338.00	0.34%	12.41%	0.00%
w10	683.70	-1883.81	380.25	680.40	-2017.60	380.25	0.48%	7.10%	0.00%
w11	719.79	-101.16	414.05	709.68	-214.34	414.05	1.40%	111.87%	0.00%

Please note that since $\mathbf{z}_{SP}$ does not dominate the point $\mathbf{z}_{EVP}$, bounds of VSS cannot be calculated using the definitions given in Chapter 4 in the first approach. In the second approach, however, the VSS is calculated as $u(\mathbf{z}_{EVP*}, \mathbf{z}_{SP_2}) = \max\,(707.0 - 707.0, 497.0 - 497.0) = 0$.

The VSS values calculated applying the first and the second approach are listed in Table 5.7.

Table 5.7: The VSS values for example instance using first and second approaches.

	First Approach			**Second Approach**		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	0.00%	0.00%	NDN	0.00%	0.00%	0.00%
w2	0.00%	0.00%	NDN	0.00%	0.00%	0.00%
w3	0.00%	0.00%	NDN	0.00%	0.00%	0.00%
w4	0.00%	NDN	NDN	0.00%	0.00%	0.00%
w5	NDN	NDN	NDN	0.00%	0.00%	0.00%
w6	0.00%	0.00%	NDN	0.00%	0.00%	0.00%
w7	NDN	NDN	0.00%	0.00%	0.00%	0.00%
w8	NDN	NDN	0.00%	0.00%	1.21%	0.00%
w9	NDN	NDN	0.00%	0.00%	1.97%	0.00%
w10	NDN	NDN	0.00%	0.00%	0.00%	0.00%
w11	NDN	NDN	0.00%	0.00%	0.00%	0.00%

The efficient solution of SP that maximizes $f(z)$ is $\mathbf{z}_{SP*} = (707.0, 497.0)$. Without considering dominance relation, the efficient solution of WSP that maximizes $f(z)$ is $\mathbf{z}_{WSP_1} = (797.0, 517.0)$. Please note that the point $\mathbf{z}_{WSP_1}$ dominates the point $\mathbf{z}_{SP*} = (707.0, 497.)$. Hence, the point obtained on the WSP frontier using both approaches is $\mathbf{z}_{WSP_1}$. As a result, there is no utility loss in the example when calculating the EVPI.

The utility loss in percentage when calculating the EVPI for this utility function is $\frac{(623.8-623.0)}{623.8} = 0.13\%$. The utility loss in percentage when calculating the EVPI for all weight vectors and functions are provided in the Table 5.8. As summarized in this table, the utility loss is less than 0.47% and 0.71% for the linear and the quadratic utility function with all weight values except for one. However, the utility loss is significant for the Tchebycheff utility function.

Since the point $\mathbf{z}_{SP*} = (707.0, 497.)$ is dominated by the point $\mathbf{z}_{WSP_1} = (797.0, 517.0)$, the EVPI can be calculated by definitions provided in Chapter 4. Note that the EVPI values calculated by applying both approaches are the same. The EVPI values calculated using both approaches are listed in Table 5.9.

Table 5.8: The utility loss in percentage when calculating the EVPI for the example instance.

	First Approach			Second Approach			Difference of Approaches		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	881.73	-132.57	621.32	869.21	-325.80	590.94	1.42%	145.77%	4.89%
w2	828.25	-2390.98	570.60	824.35	-2390.98	542.70	0.47%	0.00%	4.89%
w3	770.50	-7130.24	507.20	770.50	-7180.96	482.40	0.00%	0.71%	4.89%
w4	722.55	-12066.03	443.80	721.10	-12066.03	422.10	0.20%	0.00%	4.89%
w5	685.00	-15548.41	380.40	685.00	-15548.41	361.80	0.00%	0.00%	4.89%
w6	665.25	-17080.63	317.00	665.25	-17080.63	301.50	0.00%	0.00%	4.89%
w7	665.10	-15819.85	372.60	665.10	-15819.85	292.50	0.00%	0.00%	21.50%
w8	685.30	-12392.10	434.70	685.30	-12392.10	341.25	0.00%	0.00%	21.50%
w9	714.10	-7159.36	496.80	714.10	-7159.36	390.00	0.00%	0.00%	21.50%
w10	752.35	-2468.43	558.90	752.35	-2468.43	438.75	0.00%	0.00%	21.50%
w11	792.80	-141.26	608.58	792.80	-141.26	477.75	0.00%	0.00%	21.50%

Table 5.9: The EVPI values for the example instance using the first and the second approaches.

	First Approach			**Second Approach**		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	NDN	NDN	NDN	14.75%	14.75%	11.92%
w2	NDN	16.17%	NDN	14.59%	16.17%	11.92%
w3	13.56%	NDN	NDN	13.56%	16.32%	11.92%
w4	NDN	14.12%	NDN	16.32%	14.12%	11.92%
w5	12.15%	14.27%	NDN	12.15%	14.27%	11.92%
w6	9.23%	9.23%	NDN	9.23%	9.23%	11.92%
w7	13.37%	13.09%	NDN	13.37%	13.09%	14.75%
w8	12.32%	13.96%	NDN	12.32%	13.96%	14.75%
w9	14.91%	12.70%	NDN	14.91%	12.70%	14.75%
w10	11.84%	14.91%	NDN	11.84%	14.91%	14.75%
w11	11.92%	11.84%	NDN	11.92%	11.84%	14.75%

5.4.2 Results of Experiments

The loss on utility function value is ignorable when underlying function is linear or quadratic. That is, using second approach instead of first one does not give rise to a significant loss in utility function value. If Tchebycheff function is used in order to represent the DM's utility function, an average loss from 0.903% to 2.671% is observed.

The average loss of all 10 instances on utility function value by applying the second approach is provided in Table 5.10.

Table 5.10: The average utility loss when calculating the EVPI.

	Linear	**Quadratic**	**Tchebycheff**
w1	0.002%	0.014%	0.903%
w2	0.005%	0.000%	0.903%
w3	0.009%	0.000%	0.903%
w4	0.000%	0.000%	0.903%
w5	0.000%	0.000%	0.903%
w6	0.000%	0.000%	2.671%
w7	0.000%	0.000%	0.619%
w8	0.000%	0.000%	0.619%
w9	0.000%	0.000%	0.619%
w10	0.000%	0.000%	0.619%
w11	0.004%	0.000%	0.619%

The average loss on 10 instances on utility function value by applying the second approach when calculating the VSS are given in Table 5.11.

Observe that, if the underlying function representing the DM's preferences is linear, utility function value loss due to applying the second approach is ignorable. On average, a loss from 0.344% to 83.841% is presence when the utility function is assumed to be quadratic. Please note that, using $\mathbf{w1}$ ($\mathbf{w11}$) means that the DM gives much importance on the first (the second) objective function. In such cases, the problem is almost like a single-objective problem. Therefore, high percentage of utility function value loss observed on such extreme points. In the case where the utility function is simulated by Tchebycheff distance function, an average loss from 7.563% to 11.204% is calculated.

Table 5.11: The average utility loss when calculating the VSS.

	Linear	**Quadratic**	**Tchebycheff**
w1	0.263%	83.841%	7.563%
w2	0.073%	5.067%	7.563%
w3	0.076%	1.681%	7.563%
w4	0.012%	0.948%	7.563%
w5	0.077%	1.218%	7.563%
w6	0.016%	0.456%	9.116%
w7	0.025%	0.344%	11.204%
w8	0.055%	0.527%	11.204%
w9	0.040%	1.430%	11.204%
w10	0.044%	1.439%	11.204%
w11	0.110%	14.785%	11.204%

CHAPTER 6

CONCLUSION

In this thesis, we study the value of capturing uncertainty in two-stage multi-objective stochastic programs. We propose measures that can be used to estimate the lower and upper bounds on the VSS and EVPI when the preferences of the decision maker is unknown. VSS bounds are proposed for every solution on the efficient frontier generated by solving the expected value problem and for the frontier itself. Similarly, EVPI bounds are proposed for every efficient solution and for the efficient frontier of the stochastic program.

We test the performance of our bounds for a knapsack problem, which is one of the well studied problems in multi-objective programming literature. Using our results, we are able to identify the regions along the efficient frontier where the maximum value of modeling uncertainty is attained. We observe from our results that, maximum VSS values occur in one extreme portion of the frontier whereas high EVPI values are attained in both extreme portions of the frontier and hence the maximum EVPI value may be in either of these regions.

By using our results, we also discuss the validity of our bounds in the presence the decision maker's preferences by considering various utility functions with different objective weights. We measure the validity of our bounds by using the difference between the utility values of the solutions considered in our approach and the solutions selected based on the utility function of the decision maker. We observe from our results that this gap is negligibly small for linear utility functions but could be large for other utility functions with certain objective weights.

REFERENCES

Beale, E. (1955). On minimizing a convex function subject to linear inequalities. *Journal of the Royal Statistical Society*, Series B (Methodological):173–184.

Ben Abdelaziz, F. (2012). Solution approaches for the multiobjective stochastic programming. *European Journal of Operational Research*, 216:1–16.

Birge, J. and Louveaux, F. (2011). *Introduction to Stochastic Programming*. Springer, New York, NY, second edition.

Bérubé, J.-F., Gendreau, M., and Potvin, J.-Y. (2009). An exact ϵ-constraint method for bi-objective combinatorial optimization problems: Application to the Traveling Salesman Problem with Profits. *European Journal of Operational Research*, 194(1):39–50.

Cardona-Valdes, Y., Alvarez, A., and Özdemir, D. (2011). A bi-objective supply chain design problem with uncertainty. *Transportation Research Part C-emerging Technologies - TRANSPORT RES C-EMERG TECHNOL*, 19:821–832.

Dantzig, G. (1955). Linear programming under uncertainty. *Management Science*, 1(3-4):197–206.

Deb, K. (2001). *Multiobjective Optimization Using Evolutionary Algorithms. Wiley, New York*. Wiley, New York.

Ehrgott, M. (2005). *Multicriteria Optimization*. Springer, Berlin, Heidelberg, second edition.

Ehrgott, M. and Gandibleux, X. (2000). An annotated bibliography of multiobjective combinatorial optimization. *OR Spektrum*, 22:425–460.

Fonseca, M., García-Sánchez, l., Ortega-Mier, M., and Saldanha-da Gama, F. (2010). A stochastic bi-objective location model for strategic reverse logistics. *TOP*, 18:158–184.

Gannouni, A., Ellaia, R., and Talbi, E.-G. (2017). Solving stochastic multiobjective vehicle routing problem using probabilistic metaheuristic. MATEC Web of Conferences.

Gutjahr, W. (2009). A provably convergent heuristic for stochastic bicriteria integer programming. *J. Heuristics*, 15:227–258.

Gutjahr, W. J. and Reiter, P. (2010). Bi-objective project portfolio selection and staff assignment under uncertainty. *Optimization*, 59(3):417–445.

Haimes, Y. Y., Lasdon, L. S., and Wismer, D. A. (1971). On a bicriterion formulation of the problems of integrated system identification and system optimization. *IEEE Transactions on Systems, Man, and Cybernetics*, (1):296–297.

Hao, X., Gen, M., Lin, L., and Süer, G. (2015). Effective multiobjective eda for bi-criteria stochastic job-shop scheduling problem. *Journal of Intelligent Manufacturing*, 28.

J. Gutjahr, W. and Pichler, A. (2013). Stochastic multi-objective optimization: A survey on non-scalarizing methods. *Annals of Operations Research*, 236.

Kall, P. and Wallace, S. (1994). *Stochastic Programming*. Wiley.

Köksalan, M. (2016). *Files*. Retrieved from: http://users.metu.edu.tr/koksalan/files.html [Accessed: November 4, 2018].

Köksalan, M. and Wallenius, J. (2012). *Multiple Criteria Decision Making: Foundations and Some Approaches*, pages 171–183.

Kosuch, S. (2014). Approximability of the two-stage stochastic knapsack problem with discretely distributed weights. *Discrete Applied Mathematics*, 165:192 – 204.

Lokman, B., Köksalan, M., Korhonen, P. J., and Wallenius, J. (2016). An interactive algorithm to find the most preferred solution of multi-objective integer programs. *Annals of Operations Research*, 245(1):67–95.

Norkin, B. (2014). Sample approximations of multiobjective stochastic optimization problems. Electronic preprint. Available via: http://www.optimization-online.org/DB_HTML/2014/11/4655.html.

Norkin, V. I. (2017). *B&B Solution Technique for Multicriteria Stochastic Optimization Problems*, pages 345–378.

R. Caballero, E. Cerda, M. d. M. M. and Rey, L. (2004). Stochastic approach versus multiobjective approach for obtaining efficient solutions in stochastic multiobjective programming problems. *European Journal of Operational Research*.

Rardin, R. L. (1998). *Optimization in Operations Research*. Prentice Hall.

Rath, S., Gendreau, M., and J. Gutjahr, W. (2015). Bi-objective stochastic programming models for determining depot locations in disaster relief operations. *International Transactions in Operational Research*, 23.

Ross, G. and Soland, R. (1980). A multicriteria approach to the location of public facilities. *European Journal of Operational Research*, 4:307–321.

Tricoire, F., Graf, A., and J Gutjahr, W. (2012). The bi-objective stochastic covering tour problem. *Computers & operations research*, 39:1582–1592.

Zehetner, M. and J. Gutjahr, W. (2018). *Sampling-Based Genetic Algorithms for the Bi-Objective Stochastic Covering Tour Problem*, pages 253–284. Springer.

APPENDIX A

RESULTS OF INSTANCES ON MOSKP

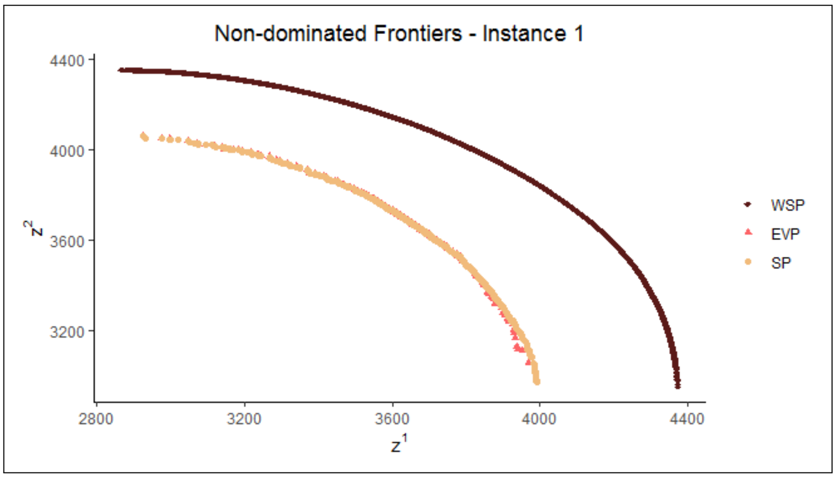

Figure A.1: Non-dominated frontiers of the MOSKP - Instance 1.

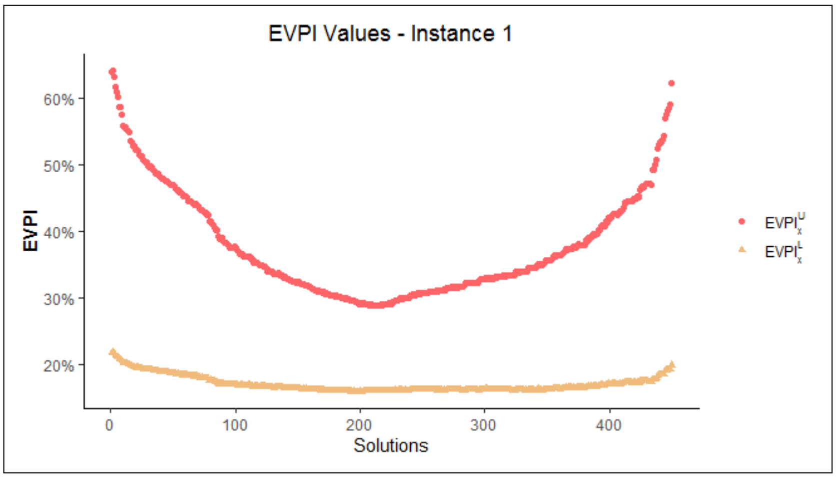

Figure A.2: The EVPI bounds of the MOSKP - Instance 1.

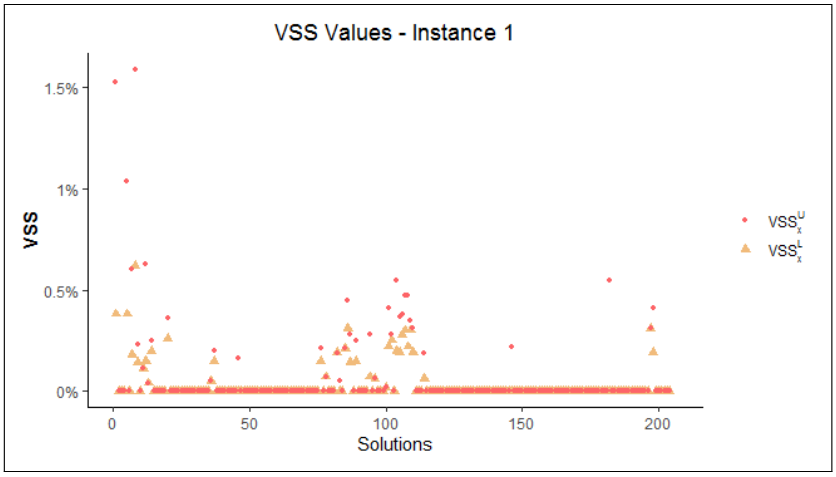

Figure A.3: The VSS bounds of the MOSKP - Instance 1.

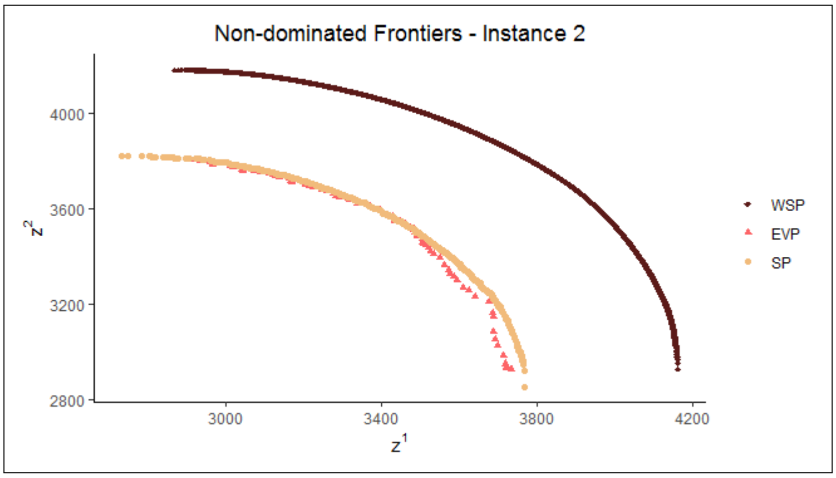

Figure A.4: Non-dominated frontiers of the MOSKP - Instance 2.

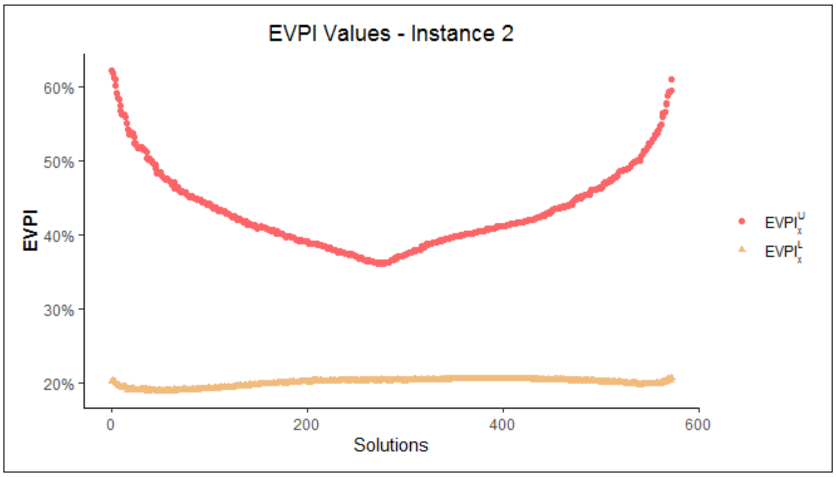

Figure A.5: The EVPI bounds of the MOSKP - Instance 2.

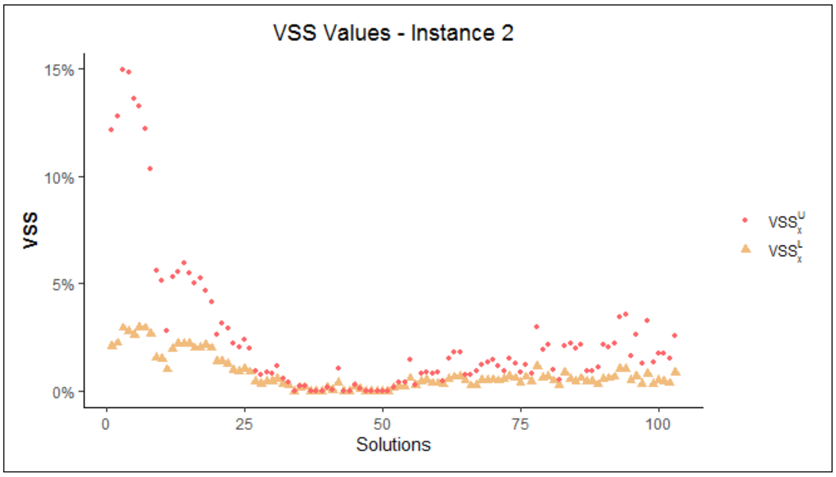

Figure A.6: The VSS bounds of the MOSKP - Instance 2.

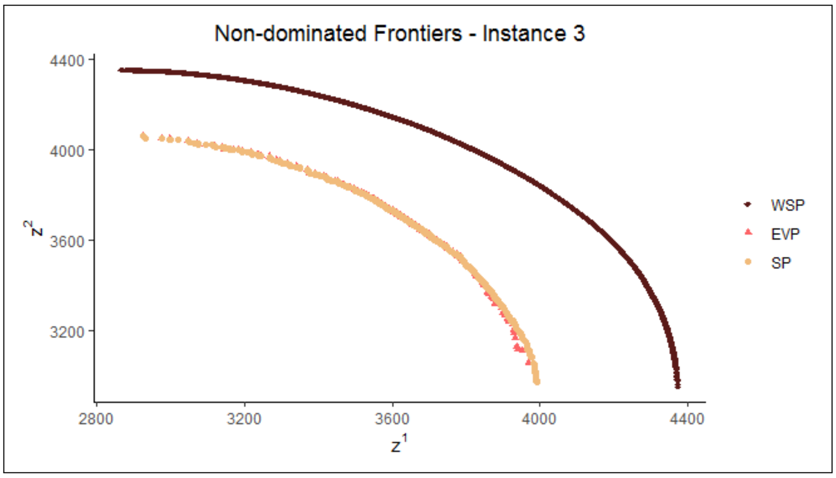

Figure A.7: Non-dominated frontiers of the MOSKP - Instance 3.

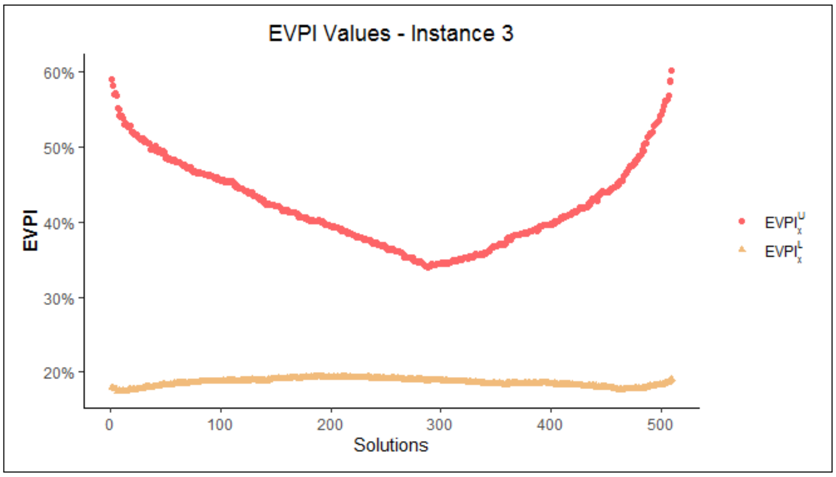

Figure A.8: The EVPI bounds of the MOSKP - Instance 3.

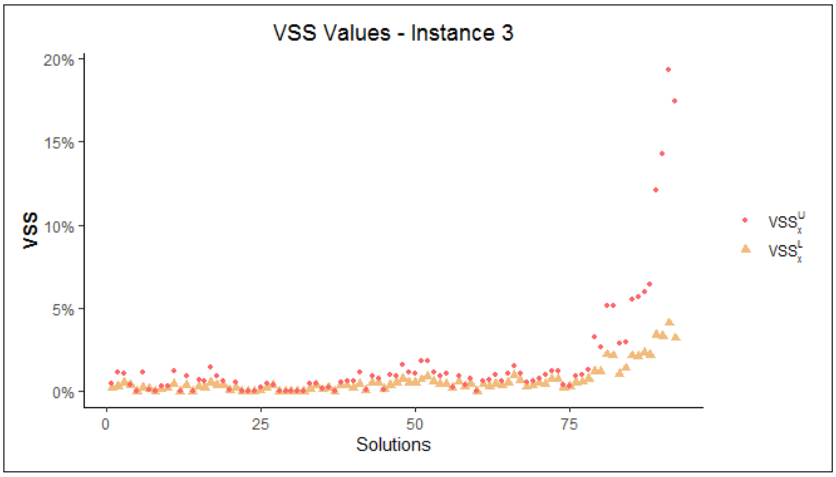

Figure A.9: The VSS bounds of the MOSKP - Instance 3.

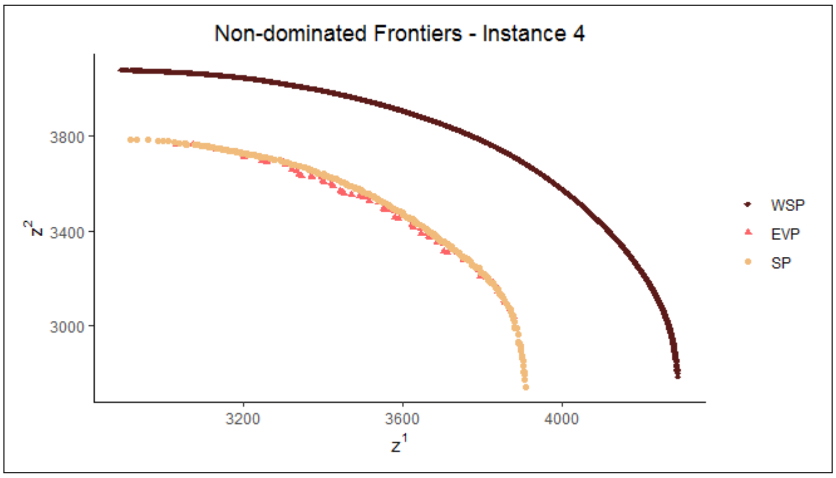

Figure A.10: Non-dominated frontiers of the MOSKP - Instance 4.

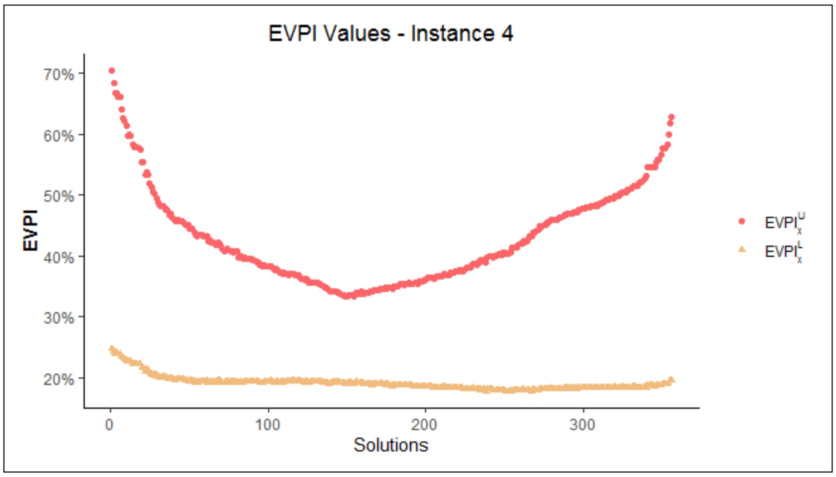

Figure A.11: The EVPI bounds of the MOSKP - Instance 4.

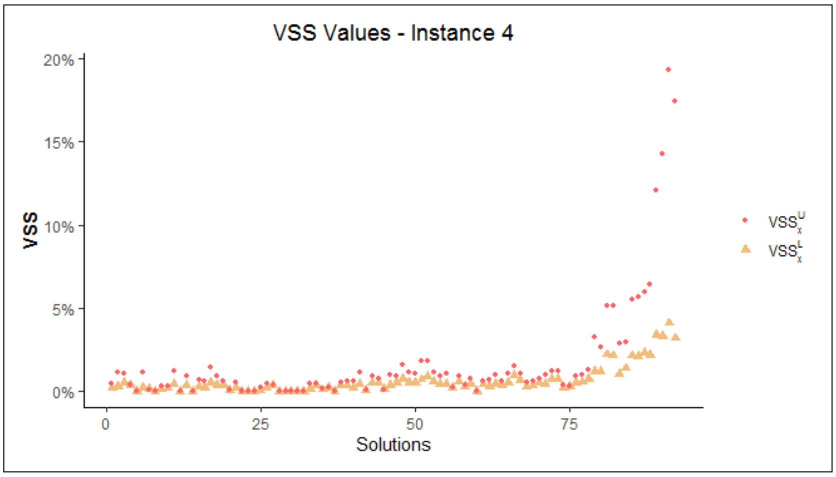

Figure A.12: The VSS bounds of the MOSKP - Instance 4.

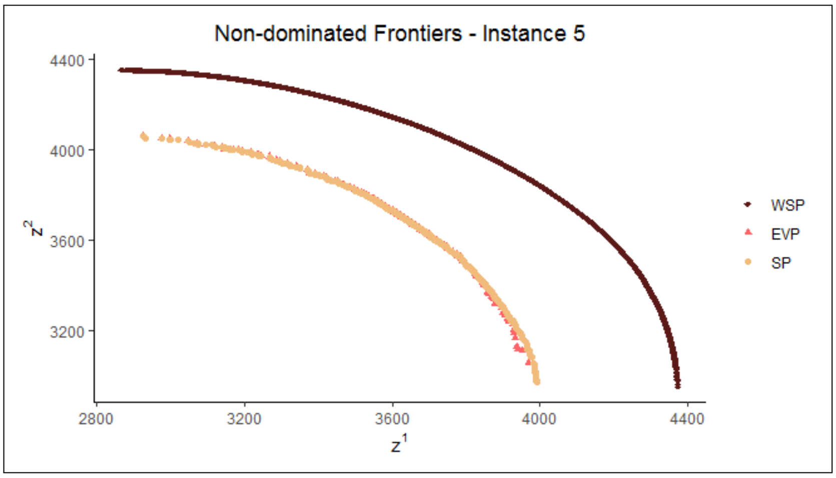

Figure A.13: Non-dominated frontiers of the MOSKP - Instance 5.

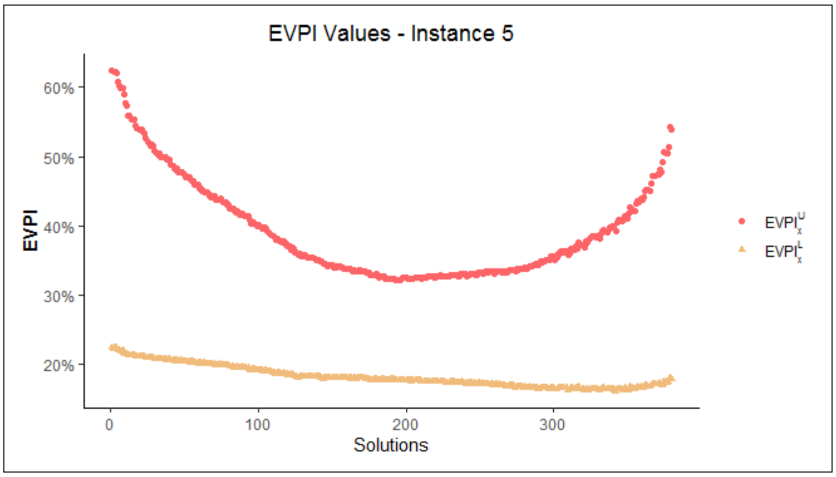

Figure A.14: The EVPI bounds of the MOSKP - Instance 5.

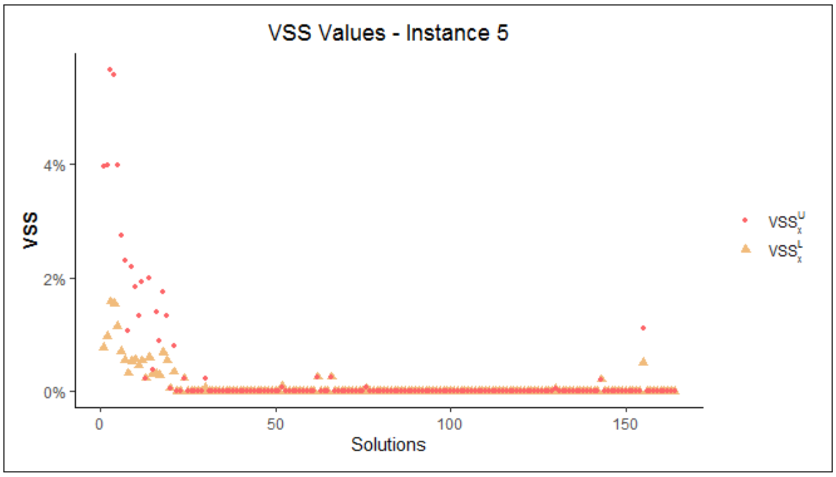

Figure A.15: The VSS bounds of the MOSKP- Instance 5.

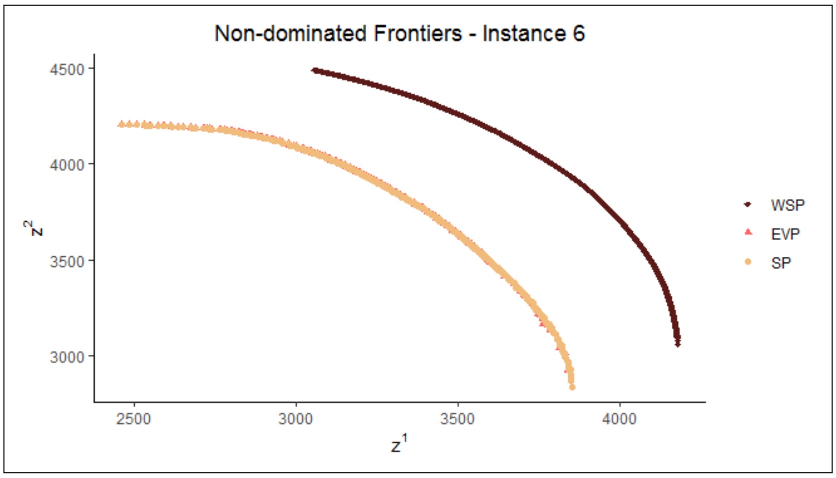

Figure A.16: Non-dominated frontiers of the MOSKP - Instance 6.

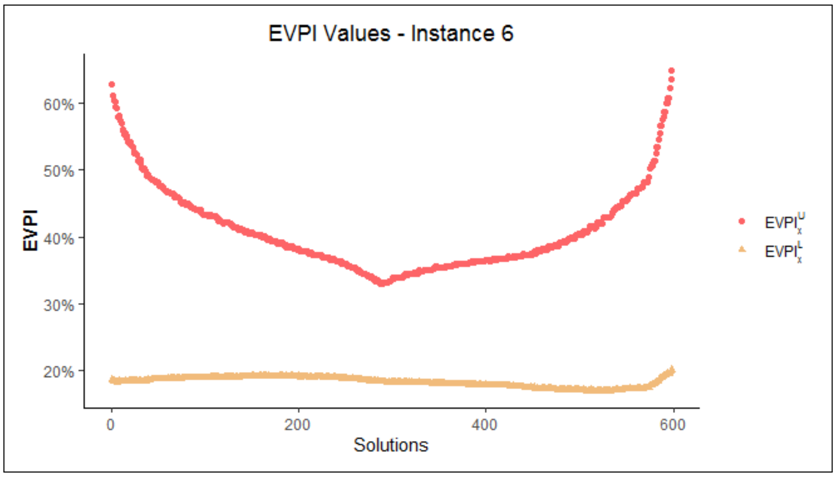

Figure A.17: The EVPI bounds of the MOSKP - Instance 6.

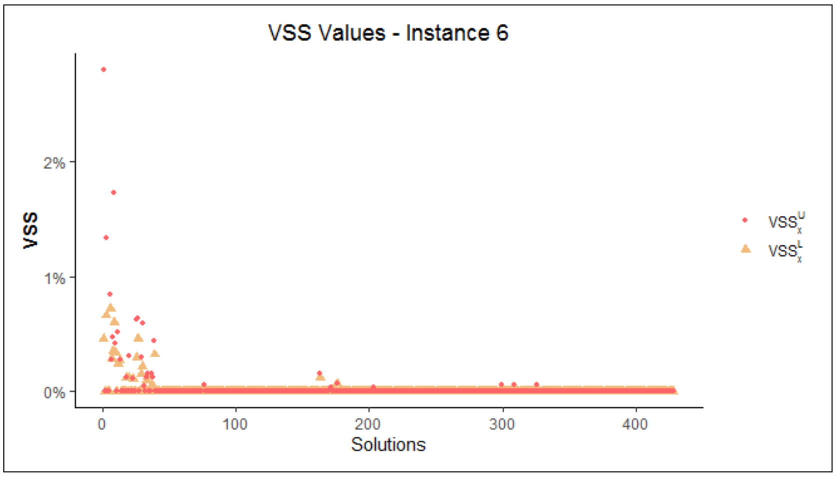

Figure A.18: The VSS bounds of the MOSKP - Instance 6.

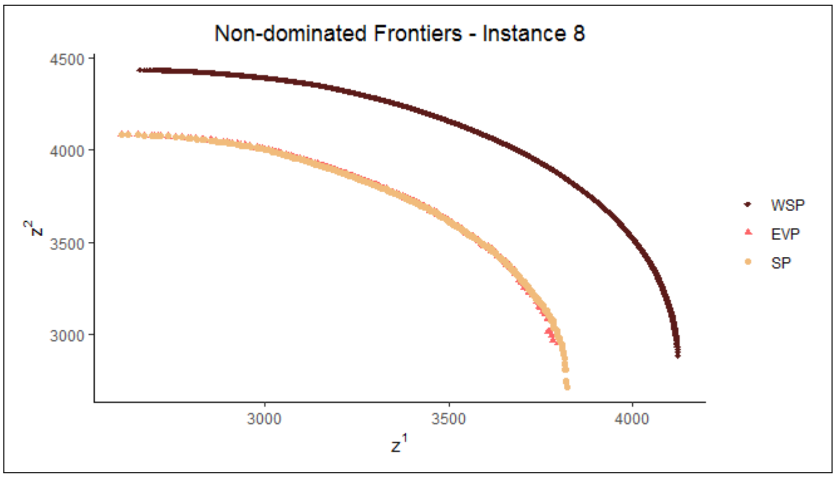

Figure A.19: Non-dominated frontiers of the MOSKP - Instance 8.

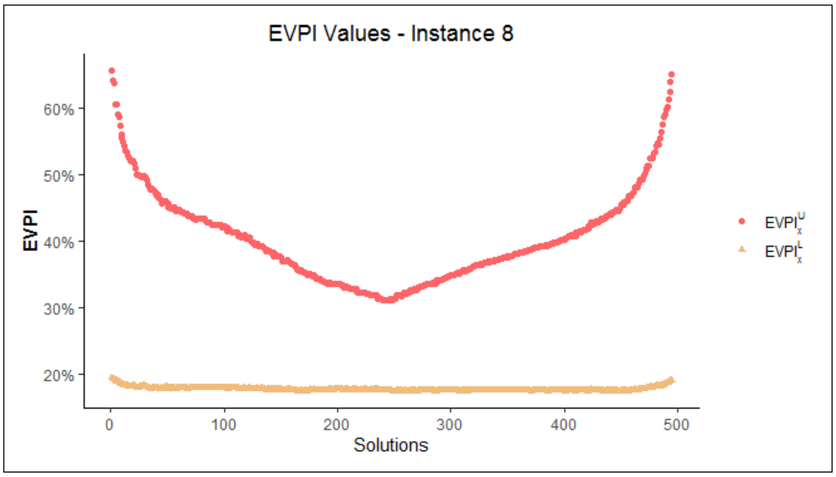

Figure A.20: The EVPI bounds of the MOSKP - Instance 8.

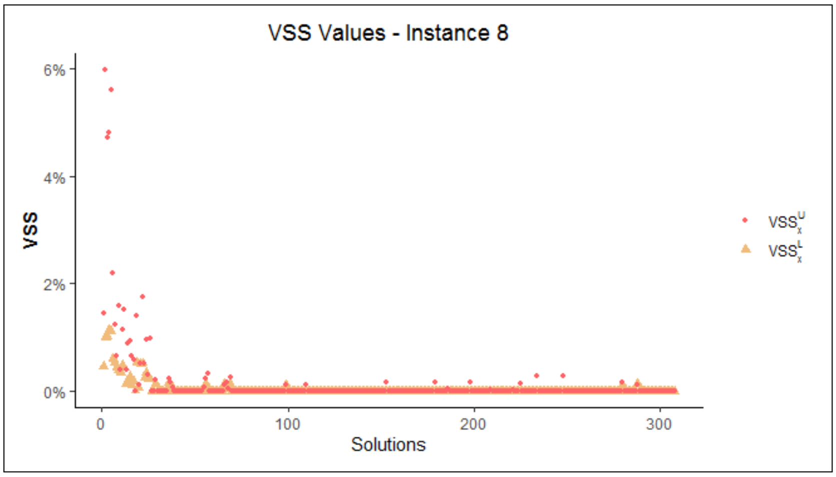

Figure A.21: The VSS bounds of the MOSKP - Instance 8.

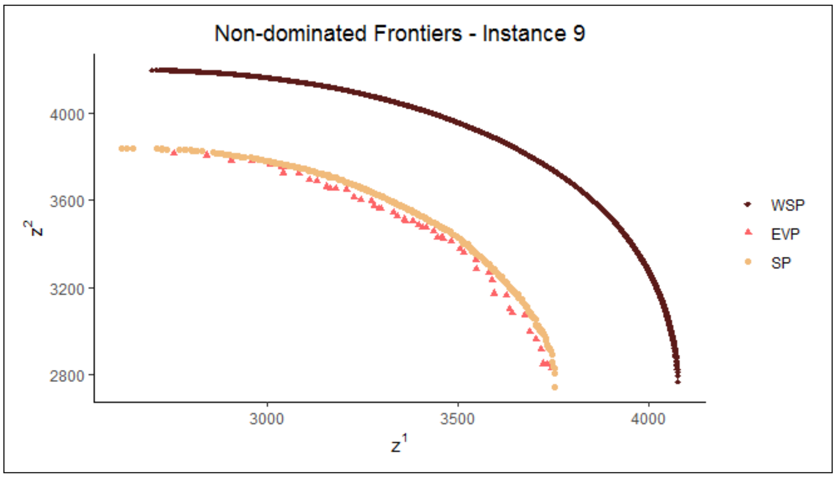

Figure A.22: Non-dominated frontiers of the MOSKP - Instance 9.

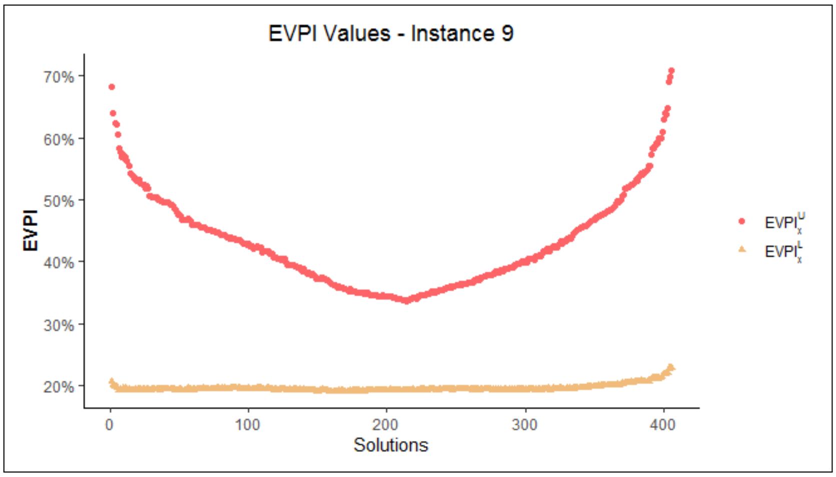

Figure A.23: The EVPI bounds of the MOSKP - Instance 9.

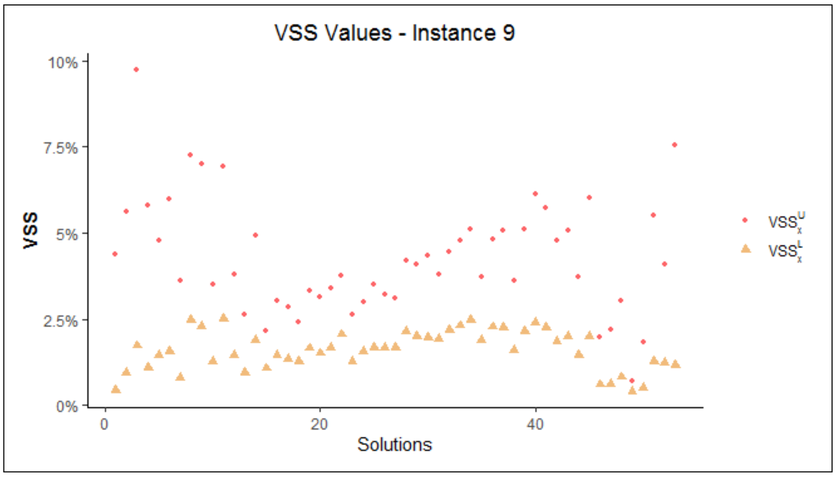

Figure A.24: The VSS bounds of the MOSKP - Instance 9.

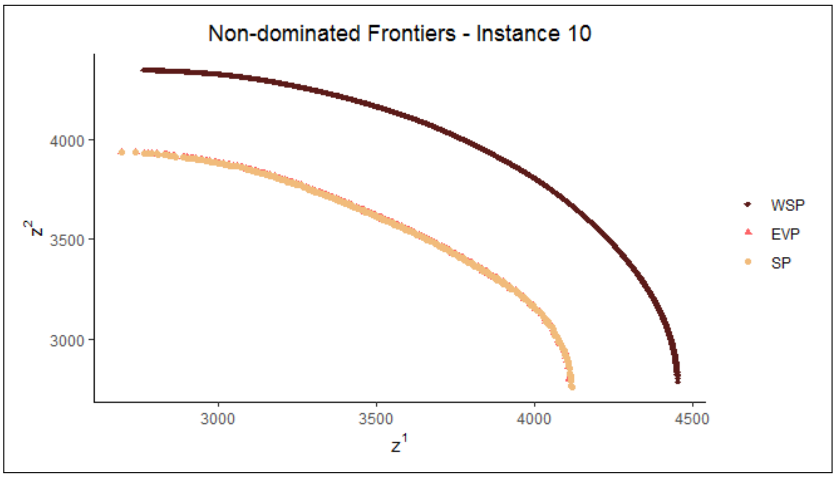

Figure A.25: Non-dominated frontiers of the MOSKP - Instance 10.

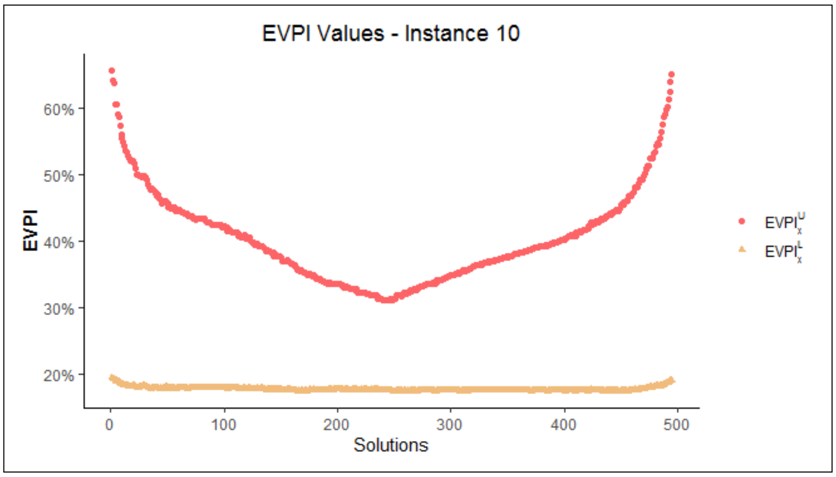

Figure A.26: The EVPI bounds of the MOSKP - Instance 10.

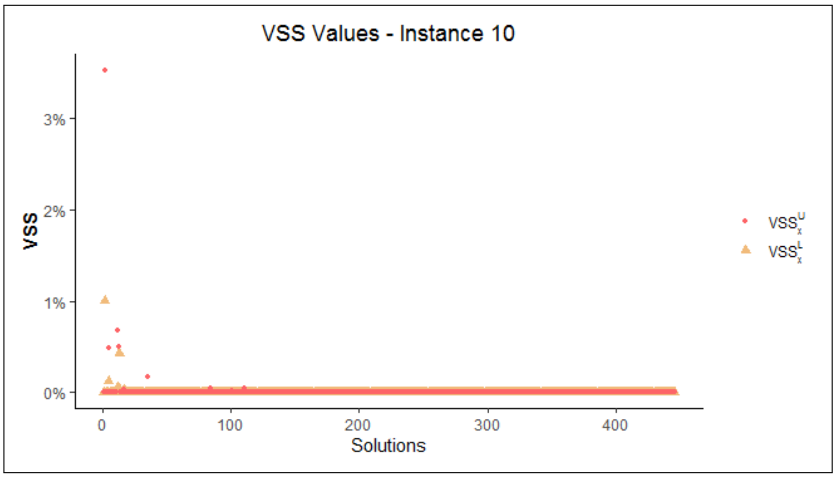

Figure A.27: The VSS bounds of the MOSKP - Instance 10.

APPENDIX B

EVALUATING THE UTILITY LOSS FROM THE DOMINANCE RESTRICTION WHEN CALCULATING THE VSS

Table B.1: The utility function values for calculating the VSS of the MOSKP Instance 1.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	3685.66	-445.37	1102.40	3635.74	-3473.81	892.19	1.35%	679.99%	19.07%
w2	3611.19	-7858.36	1012.41	3590.34	-9889.14	819.36	0.58%	25.84%	19.07%
w3	3534.68	-21157.34	899.92	3532.34	-22902.75	728.32	0.07%	8.25%	19.07%
w4	3486.67	-35208.18	787.43	3485.99	-36215.95	637.28	0.02%	2.86%	19.07%
w5	3452.62	-45429.84	674.94	3452.62	-46119.36	546.24	0.00%	1.52%	19.07%
w6	3447.00	-50283.81	562.45	3447.00	-50283.81	455.20	0.00%	0.00%	19.07%
w7	3465.94	-48057.82	658.98	3465.94	-48248.01	445.68	0.00%	0.40%	32.37%
w8	3520.92	-37588.49	768.81	3520.92	-37588.49	519.96	0.00%	0.00%	32.37%
w9	3606.98	-22946.29	878.64	3606.98	-23053.06	594.24	0.00%	0.47%	32.37%
w10	3704.30	-7648.30	988.47	3703.24	-7652.65	668.52	0.03%	0.06%	32.37%
w11	3792.62	-408.35	1076.33	3784.70	-592.49	727.94	0.21%	45.09%	32.37%

Table B.2: The utility function values for calculating the VSS of the MOSKP Instance 2.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	3751.26	-315.08	1016.75	3748.71	-315.08	829.86	0.07%	0.00%	18.38%
w2	3683.72	-6209.68	933.75	3683.72	-6649.59	762.12	0.00%	7.08%	18.38%
w3	3607.52	-17969.24	830.00	3599.10	-17969.24	677.44	0.23%	0.00%	18.38%
w4	3551.24	-31655.02	726.25	3551.24	-33886.42	592.76	0.00%	7.05%	18.38%
w5	3509.62	-41005.08	622.50	3494.98	-44443.18	508.08	0.42%	8.38%	18.38%
w6	3499.30	-43517.00	518.75	3498.10	-43933.35	437.10	0.03%	0.96%	15.74%
w7	3517.60	-40516.59	580.08	3515.04	-40516.59	524.52	0.07%	0.00%	9.58%
w8	3563.87	-32236.97	676.76	3562.66	-32236.97	611.94	0.03%	0.00%	9.58%
w9	3634.48	-19267.29	773.44	3632.96	-19407.84	699.36	0.04%	0.73%	9.58%
w10	3721.03	-6463.06	870.12	3720.16	-6626.91	786.78	0.02%	2.54%	9.58%
w11	3798.38	-364.46	947.46	3791.26	-435.98	856.72	0.19%	19.62%	9.58%

Table B.3: The utility function values for calculating the VSS of the MOSKP Instance 3.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	4065.74	-574.27	1266.26	4063.39	-574.27	1266.26	0.06%	0.00%	0.00%
w2	3951.73	-12722.91	1162.89	3951.73	-12781.95	1162.89	0.00%	0.46%	0.00%
w3	3813.42	-42067.56	1033.68	3812.16	-42099.51	1033.68	0.03%	0.08%	0.00%
w4	3682.48	-72680.78	904.47	3680.84	-72680.78	904.47	0.04%	0.00%	0.00%
w5	3581.68	-97170.84	775.26	3581.68	-99405.06	775.26	0.00%	2.30%	0.00%
w6	3539.15	-101988.22	646.05	3538.00	-103263.49	646.05	0.03%	1.25%	0.00%
w7	3551.52	-90079.55	748.68	3545.70	-90127.92	710.64	0.16%	0.05%	5.08%
w8	3606.89	-68380.34	873.46	3600.05	-71779.7	829.08	0.19%	4.97%	5.08%
w9	3683.32	-38945.6	998.24	3679.04	-40676.53	947.52	0.12%	4.44%	5.08%
w10	3775.3	-12236.29	1123.02	3775.30	-12429.97	1065.96	0.00%	1.58%	5.08%
w11	3858.46	-618.73	1222.84	3858.46	-618.73	1160.71	0.00%	0.00%	5.08%

Table B.4: The utility function values for calculating the VSS of the MOSKP Instance 4.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	3885.74	-381.40	969.91	3862.22	-1092.74	852.40	0.61%	186.51%	12.11%
w2	3798.48	-6299.03	890.73	3795.12	-6743.61	782.82	0.09%	7.06%	12.11%
w3	3711.24	-18811.01	791.76	3711.24	-19083.41	695.84	0.00%	1.45%	12.11%
w4	3634.72	-32975.56	692.79	3634.72	-32975.56	608.86	0.00%	0.00%	12.11%
w5	3572.34	-44425.96	593.82	3572.12	-44941.02	521.88	0.01%	1.16%	12.11%
w6	3539.50	-47725.05	522.60	3535.50	-48973.06	434.90	0.11%	2.61%	16.78%
w7	3546.72	-43454.03	627.12	3546.72	-43744.93	447.60	0.00%	0.67%	28.63%
w8	3577.45	-33168.41	731.64	3576.07	-33168.41	522.20	0.04%	0.00%	28.63%
w9	3629.08	-20012.67	836.16	3629.08	-20241.56	596.80	0.00%	1.14%	28.63%
w10	3703.66	-6994.91	940.68	3697.89	-7040.17	671.40	0.16%	0.65%	28.63%
w11	3768.86	-348.38	1024.30	3752.94	-641.08	731.08	0.42%	84.02%	28.63%

Table B.5: The utility function values for calculating the VSS of the MOSKP Instance 5.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	3970.80	-453.44	1043.41	3962.07	-509.43	1043.41	0.22%	12.35%	0.00%
w2	3890.40	-9079.01	958.23	3888.35	-9861.95	958.23	0.05%	8.62%	0.00%
w3	3798.82	-28047.02	851.76	3790.60	-29525	851.76	0.22%	5.27%	0.00%
w4	3721.60	-45978.17	745.29	3721.60	-45978.17	745.29	0.00%	0.00%	0.00%
w5	3681.20	-59495.85	638.82	3681.20	-59495.85	638.82	0.00%	0.00%	0.00%
w6	3667.10	-64772.36	542.50	3667.10	-64772.36	532.35	0.00%	0.00%	1.87%
w7	3697.36	-58333.90	651.00	3697.36	-58364.69	601.14	0.00%	0.05%	7.66%
w8	3759.87	-45095.37	759.50	3759.87	-45095.37	701.33	0.00%	0.00%	7.66%
w9	3842.68	-26133.82	868.00	3842.68	-26133.82	801.52	0.00%	0.00%	7.66%
w10	3947.45	-8915.71	976.5	3947.45	-8962.08	901.71	0.00%	0.52%	7.66%
w11	4038.21	-453.43	1063.3	4038.21	-453.43	981.86	0.00%	0.00%	7.66%

Table B.6: The utility function values for calculating the VSS of the MOSKP Instance 6.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	3734.66	-394.80	1115.53	3733.88	-394.80	967.36	0.02%	0.00%	13.28%
w2	3662.25	-7919.27	1024.47	3662.25	-7919.27	888.39	0.00%	0.00%	13.28%
w3	3578.90	-23993.13	910.64	3577.00	-24245.75	789.68	0.05%	1.05%	13.28%
w4	3510.99	-40006.64	796.81	3510.79	-40006.64	690.97	0.01%	0.00%	13.28%
w5	3474.94	-49730.61	682.98	3473.76	-49739.93	592.26	0.03%	0.02%	13.28%
w6	3468.30	-54256.29	569.15	3468.30	-54256.29	493.55	0.00%	0.00%	13.28%
w7	3495.68	-50375.31	655.92	3494.96	-50375.31	604.86	0.02%	0.00%	7.78%
w8	3551.47	-38137.29	765.24	3545.39	-38294.54	705.67	0.17%	0.41%	7.78%
w9	3629.58	-22799.46	874.56	3625.28	-23241.10	806.48	0.12%	1.94%	7.78%
w10	3728.02	-7919.93	983.88	3728.02	-8266.83	907.29	0.00%	4.38%	7.78%
w11	3814.82	-413.96	1071.34	3812.16	-413.96	987.94	0.07%	0.00%	7.78%

Table B.7: The utility function values for calculating the VSS of the MOSKP Instance 7.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	3796.31	-575.65	1172.86	3791.11	-575.65	1090.05	0.14%	0.00%	7.06%
w2	3721.35	-11364.42	1077.12	3721.35	-11461.73	1001.07	0.00%	0.86%	7.06%
w3	3639.14	-35792.51	957.44	3637.66	-35792.51	889.84	0.04%	0.00%	7.06%
w4	3571.06	-63180.75	837.76	3569.13	-63281.64	778.61	0.05%	0.16%	7.06%
w5	3524.96	-84268.65	718.08	3512.58	-84268.65	667.38	0.35%	0.00%	7.06%
w6	3518.65	-92912.4	611.35	3518.65	-92912.4	556.15	0.00%	0.00%	9.03%
w7	3571.52	-85885.66	733.62	3571.52	-88067.72	692.46	0.00%	2.54%	5.61%
w8	3675.45	-65238.9	855.89	3675.45	-65238.9	807.87	0.00%	0.00%	5.61%
w9	3800.22	-37226.22	978.16	3798.44	-39113.95	923.28	0.05%	5.07%	5.61%
w10	3937.01	-11460.85	1100.43	3925.97	-11658.68	1038.69	0.28%	1.73%	5.61%
w11	4054.2	-555.90	1198.25	4044.13	-633.18	1131.02	0.25%	13.9%	5.61%

Table B.8: The utility function values for calculating the VSS of the MOSKP Instance 8.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	3799.62	-630.87	1188.94	3786.74	-819.08	1188.94	0.34%	29.83%	0.00%
w2	3721.39	-11324.49	1091.88	3720.15	-11913.32	1091.88	0.03%	5.20%	0.00%
w3	3642.28	-34228.74	970.56	3637.3	-34686.70	970.56	0.14%	1.34%	0.00%
w4	3580.06	-55236.55	849.24	3580.06	-55431.84	849.24	0.00%	0.35%	0.00%
w5	3555.02	-71070.20	727.92	3555.02	-71070.2	727.92	0.00%	0.00%	0.00%
w6	3564.00	-75758.23	683.3	3564.00	-75905.45	606.60	0.00%	0.19%	11.22%
w7	3611.72	-70254.72	819.96	3611.72	-70301.42	671.46	0.00%	0.07%	18.11%
w8	3702.96	-53860.85	956.62	3702.96	-53860.85	783.37	0.00%	0.00%	18.11%
w9	3813.44	-30939.16	1093.28	3813.44	-30939.16	895.28	0.00%	0.00%	18.11%
w10	3940.18	-9993.19	1229.94	3940.18	-9993.19	1007.19	0.00%	0.00%	18.11%
w11	4054.98	-540.90	1339.27	4054.98	-540.90	1096.72	0.00%	0.00%	18.11%

Table B.9: The utility function values for calculating the VSS of the MOSKP Instance 9.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	3734.66	-394.80	1115.53	3733.88	-394.80	967.36	0.02%	0.00%	13.28%
w2	3662.25	-7919.27	1024.47	3662.25	-7919.27	888.39	0.00%	0.00%	13.28%
w3	3578.90	-23993.13	910.64	3577.00	-24245.75	789.68	0.05%	1.05%	13.28%
w4	3510.99	-40006.64	796.81	3510.79	-40006.64	690.97	0.01%	0.00%	13.28%
w5	3474.94	-49730.61	682.98	3473.76	-49739.93	592.26	0.03%	0.02%	13.28%
w6	3468.30	-54256.29	569.15	3468.30	-54256.29	493.55	0.00%	0.00%	13.28%
w7	3495.68	-50375.31	655.92	3494.96	-50375.31	604.86	0.02%	0.00%	7.78%
w8	3551.47	-38137.29	765.24	3545.39	-38294.54	705.67	0.17%	0.41%	7.78%
w9	3629.58	-22799.46	874.56	3625.28	-23241.10	806.48	0.12%	1.94%	7.78%
w10	3728.02	-7919.93	983.88	3728.02	-8266.83	907.29	0.00%	4.38%	7.78%
w11	3814.82	-413.96	1071.34	3812.16	-413.96	987.94	0.07%	0.00%	7.78%

Table B.10: The utility function values for calculating the VSS of the MOSKP Instance 10.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	4092.50	-524.64	1396.11	4089.61	-595.85	1396.11	0.07%	14.00%	0.00%
w2	3985.38	-10184.62	1282.14	3983.25	-10246.40	1282.14	0.05%	1.00%	0.00%
w3	3864.18	-32212.02	1139.68	3864.18	-32212.02	1139.68	0.00%	0.00%	0.00%
w4	3757.19	-58494.67	997.22	3757.19	-58494.67	997.22	0.00%	0.00%	0.00%
w5	3665.10	-83534.15	854.76	3665.10	-83534.15	854.76	0.00%	0.00%	0.00%
w6	3593.45	-98127.65	712.30	3593.45	-98127.65	712.30	0.00%	0.00%	0.00%
w7	3570.08	-96468.10	704.88	3570.08	-96468.10	700.38	0.00%	0.00%	0.64%
w8	3622.99	-76912.94	822.36	3622.99	-76912.94	817.11	0.00%	0.00%	0.64%
w9	3707.50	-45006.38	939.84	3707.50	-450006.38	933.84	0.00%	0.00%	0.64%
w10	3814.49	-14791.59	1057.32	3814.49	-14791.59	1050.57	0.00%	0.00%	0.64%
w11	3909.94	-743.43	1151.30	3909.94	-743.43	1143.95	0.00%	0.00%	0.64%

APPENDIX C

EVALUATING THE UTILITY LOSS FROM THE DOMINANCE RESTRICTION WHEN CALCULATING THE EVPI

Table C.1: The utility function values for calculating the EVPI of the MOSKP Instance 1.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	4060.78	-746.97	1494.40	4060.74	-748.12	1476.76	0.00%	0.15%	1.18%
w2	3955.05	-14298.98	1372.41	3953.36	-14298.98	1356.21	0.04%	0.00%	1.18%
w3	3837.54	-41398.89	1219.92	3833.58	-41398.89	1205.52	0.10%	0.00%	1.18%
w4	3750.03	-66998.16	1067.43	3750.03	-66998.16	1054.83	0.00%	0.00%	1.18%
w5	3700.50	-84897.80	914.94	3700.50	-84897.80	904.14	0.00%	0.00%	1.18%
w6	3683.35	-92276.76	762.45	3683.35	-92276.76	753.45	0.00%	0.00%	1.18%
w7	3705.12	-86637.49	882.48	3705.12	-86637.49	847.02	0.00%	0.00%	4.02%
w8	3770.39	-68196.50	1029.56	3770.39	-68196.50	988.19	0.00%	0.00%	4.02%
w9	3866.14	-41106.56	1176.64	3866.14	-41106.56	1129.36	0.00%	0.00%	4.02%
w10	3990.08	-14234.22	1323.72	3990.08	-14234.22	1270.53	0.00%	0.00%	4.02%
w11	4105.54	-801.12	1441.38	4104.97	-801.12	1383.47	0.01%	0.00%	4.02%

Table C.2: The utility function values for calculating the EVPI of the MOSKP Instance 2.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	4138.97	-550.35	1269.20	4138.97	-550.35	1269.20	0.00%	0.00%	0.00%
w2	4046.64	-10278.61	1165.59	4046.64	-10278.61	1165.59	0.00%	0.00%	0.00%
w3	3946.10	-30597.31	1036.08	3946.10	-30597.31	1036.08	0.00%	0.00%	0.00%
w4	3864.93	-50632.89	906.57	3864.93	-50632.89	906.57	0.00%	0.00%	0.00%
w5	3814.74	-64989.69	777.06	3814.74	-64989.69	777.06	0.00%	0.00%	0.00%
w6	3793.40	-71840.16	647.55	3793.40	-71840.16	647.55	0.00%	0.00%	0.00%
w7	3807.68	-68087.81	751.80	3807.68	-68087.81	751.80	0.00%	0.00%	0.00%
w8	3860.36	-53947.51	877.10	3860.36	-53947.51	877.10	0.00%	0.00%	0.00%
w9	3946.20	-32842.52	1002.40	3946.20	-32842.52	1002.40	0.00%	0.00%	0.00%
w10	4055.97	-11157.20	1127.70	4055.97	-11157.20	1127.70	0.00%	0.00%	0.00%
w11	4153.99	-579.05	1227.94	4153.99	-579.05	1227.94	0.00%	0.00%	0.00%

Table C.3: The utility function values for calculating the EVPI of the MOSKP Instance 3.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	4380.89	-889.99	1503.52	4380.73	-889.99	1503.52	0.00%	0.00%	0.00%
w2	4243.73	-16365.34	1380.78	4243.73	-16365.34	1380.78	0.00%	0.00%	0.00%
w3	4093.38	-47445.67	1227.36	4093.38	-47445.67	1227.36	0.00%	0.00%	0.00%
w4	3973.84	-79747.06	1073.94	3973.84	-79747.06	1073.94	0.00%	0.00%	0.00%
w5	3883.04	-102998.38	920.52	3883.04	-102998.38	920.52	0.00%	0.00%	0.00%
w6	3836.95	-111138.85	784.55	3836.95	-111138.85	767.10	0.00%	0.00%	2.22%
w7	3842.92	-101582.28	941.46	3842.92	-101582.28	931.56	0.00%	0.00%	1.05%
w8	3886.49	-78681.86	1098.37	3886.49	-78681.86	1086.82	0.00%	0.00%	1.05%
w9	3966.74	-47530.51	1255.28	3966.74	-47530.51	1242.08	0.00%	0.00%	1.05%
w10	4076.00	-16046.34	1412.19	4076.00	-16046.34	1397.34	0.00%	0.00%	1.05%
w11	4174.18	-823.06	1537.72	4174.18	-823.06				

Table C.4: The utility function values for calculating the EVPI of the MOSKP Instance 4.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	4262.41	-617.13	1369.55	4262.41	-617.13	1343.48	0.00%	0.00%	1.90%
w2	4145.76	-11326.68	1257.75	4145.24	-11326.68	1233.81	0.01%	0.00%	1.90%
w3	4017.64	-33794.81	1118.00	4017.64	-33794.81	1096.72	0.00%	0.00%	1.90%
w4	3909.79	-55985.34	978.25	3909.79	-55985.34	959.63	0.00%	0.00%	1.90%
w5	3834.20	-71098.77	838.50	3834.20	-71098.77	822.54	0.00%	0.00%	1.90%
w6	3797.30	-75694.26	698.75	3797.30	-75694.26	648.45	0.00%	0.00%	7.20%
w7	3792.06	-70498.21	778.14	3792.06	-70498.21	778.14	0.00%	0.00%	0.00%
w8	3818.84	-56438.21	907.83	3818.84	-56438.21	907.83	0.00%	0.00%	0.00%
w9	3879.44	-35059.78	1037.52	3879.44	-35059.78	1037.52	0.00%	0.00%	0.00%
w10	3968.03	-12360.48	1167.21	3968.03	-12360.48	1167.21	0.00%	0.00%	0.00%
w11	4056.40	-699.21	1270.96	4056.25	-699.21	1270.96	0.00%	0.00%	0.00%

Table C.5: The utility function values for calculating the EVPI of the MOSKP Instance 5.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	4345.07	-712.94	1476.08	4344.58	-712.94	1417.67	0.00%	0.00%	3.96%
w2	4236.21	-13148.29	1355.58	4236.21	-13148.29	1301.94	0.00%	0.00%	3.96%
w3	4119.00	-39075.58	1204.96	4119.00	-39075.58	1157.28	0.00%	0.00%	3.96%
w4	4024.72	-66638.88	1054.34	4024.72	-66638.88	1012.62	0.00%	0.00%	3.96%
w5	3956.32	-88654.24	903.72	3956.32	-88654.24	867.96	0.00%	0.00%	3.96%
w6	3921.35	-98308.77	753.10	3921.35	-98308.77	688.50	0.00%	0.00%	8.58%
w7	3932.78	-93361.49	840.84	3932.78	-93361.49	826.20	0.00%	0.00%	1.74%
w8	3990.13	-74077.96	980.98	3990.13	-74077.96	963.90	0.00%	0.00%	1.74%
w9	4086.78	-45242.67	1121.12	4086.78	-45242.67	1101.60	0.00%	0.00%	1.74%
w10	4211.40	-15376.17	1261.26	4211.40	-15376.17	1239.30	0.00%	0.00%	1.74%
w11	4323.78	-806.19	1373.37	4322.71	-806.19	1349.46	0.00%	0.00%	1.74%

Table C.6: The utility function values for calculating the EVPI of the MOSKP Instance 6.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	4051.94	-702.71	1354.36	4051.82	-702.71	1354.36	0.00%	0.00%	0.00%
w2	3955.78	-12376.00	1243.80	3955.78	-12376.00	1243.80	0.00%	0.00%	0.00%
w3	3858.46	-34643.38	1105.60	3858.46	-34643.38	1105.60	0.00%	0.00%	0.00%
w4	3790.77	-55754.13	967.40	3790.77	-55754.13	967.40	0.00%	0.00%	0.00%
w5	3752.94	-70848.45	829.20	3752.94	-70848.45	829.20	0.00%	0.00%	0.00%
w6	3746.30	-76555.39	713.90	3746.30	-76555.39	691.00	0.00%	0.00%	3.21%
w7	3774.44	-71431.16	856.68	3774.44	-71431.16	856.68	0.00%	0.00%	0.00%
w8	3837.02	-56505.21	999.46	3837.02	-56505.21	999.46	0.00%	0.00%	0.00%
w9	3930.82	-34877.46	1142.24	3930.82	-34877.46	1142.24	0.00%	0.00%	0.00%
w10	4052.81	-12266.47	1285.02	4052.81	-12266.47	1285.02	0.00%	0.00%	0.00%
w11	4167.19	-680.83	1399.24	4167.19	-680.83	1399.24	0.00%	0.00%	0.00%

Table C.7: The utility function values for calculating the EVPI of the MOSKP Instance 7.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	4142.01	-855.33	1560.16	4142.01	-855.33	1515.08	0.00%	0.00%	2.88%
w2	4047.26	-16389.35	1432.80	4047.26	-16389.35	1391.40	0.00%	0.00%	2.88%
w3	3945.26	-48994.28	1273.60	3945.26	-48994.28	1236.80	0.00%	0.00%	2.88%
w4	3869.11	-81007.29	1114.40	3869.11	-81007.29	1082.20	0.00%	0.00%	2.88%
w5	3832.86	-102619.79	955.20	3832.86	-102619.79	927.60	0.00%	0.00%	2.88%
w6	3839.75	-110804.88	796.00	3839.75	-110804.88	765.85	0.00%	0.00%	3.78%
w7	3894.06	-101813.65	919.02	3894.06	-101813.65	919.02	0.00%	0.00%	0.00%
w8	3988.31	-79367.11	1072.19	3988.31	-79367.11	1072.19	0.00%	0.00%	0.00%
w9	4117.94	-47841.95	1225.36	4117.94	-47841.95	1225.36	0.00%	0.00%	0.00%
w10	4276.74	-16269.98	1378.53	4276.74	-16269.98	1378.53	0.00%	0.00%	0.00%
w11	4416.86	-858.37	1501.07	4416.86	-858.37	1501.07	0.00%	0.00%	0.00%

Table C.8: The utility function values for calculating the EVPI of the MOSKP Instance 8.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	4100.21	-847.16	1435.99	4100.21	-847.16	1435.99	0.00%	0.00%	0.00%
w2	4008.95	-14945.00	1318.77	4008.95	-14945.00	1318.77	0.00%	0.00%	0.00%
w3	3919.10	-42699.54	1172.24	3919.10	-42699.54	1172.24	0.00%	0.00%	0.00%
w4	3858.73	-69537.54	1025.71	3858.73	-69537.54	1025.71	0.00%	0.00%	0.00%
w5	3831.50	-88152.68	879.18	3831.50	-88152.68	879.18	0.00%	0.00%	0.00%
w6	3843.80	-94727.94	773.90	3843.80	-94727.94	773.90	0.00%	0.00%	0.00%
w7	3895.98	-88031.48	928.68	3895.98	-88031.48	928.68	0.00%	0.00%	0.00%
w8	3990.09	-68340.57	1083.46	3990.09	-68340.57	1083.46	0.00%	0.00%	0.00%
w9	4113.30	-40936.73	1238.24	4113.30	-40936.73	1238.24	0.00%	0.00%	0.00%
w10	4263.20	-13919.82	1393.02	4263.20	-13919.82	1393.02	0.00%	0.00%	0.00%
w11	4397.86	-751.09	1516.84	4397.86	-751.09	1516.84	0.00%	0.00%	0.00%

Table C.9: The utility function values for calculating the EVPI of the MOSKP Instance 9.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	4051.94	-702.71	1354.36	4051.82	-702.71	1354.36	0.00%	0.00%	0.00%
w2	3955.78	-12376.00	1243.80	3955.78	-12376.00	1243.80	0.00%	0.00%	0.00%
w3	3858.46	-34643.38	1105.60	3858.46	-34643.38	1105.60	0.00%	0.00%	0.00%
w4	3790.77	-55754.13	967.40	3790.77	-55754.13	967.40	0.00%	0.00%	0.00%
w5	3752.94	-70848.45	829.20	3752.94	-70848.45	829.20	0.00%	0.00%	0.00%
w6	3746.30	-76555.39	713.90	3746.30	-76555.39	691.00	0.00%	0.00%	3.21%
w7	3774.44	-71431.16	856.68	3774.44	-71431.16	856.68	0.00%	0.00%	0.00%
w8	3837.02	-56505.21	999.46	3837.02	-56505.21	999.46	0.00%	0.00%	0.00%
w9	3930.82	-34877.46	1142.24	3930.82	-34877.46	1142.24	0.00%	0.00%	0.00%
w10	4052.81	-12266.47	1285.02	4052.81	-12266.47	1285.02	0.00%	0.00%	0.00%
w11	4167.19	-680.83	1399.24	4167.19	-680.83	1399.24	0.00%	0.00%	0.00%

Table C.10: The utility function values for calculating the EVPI of the MOSKP Instance 10.

	First Approach			Second Approach			Difference		
	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff	Linear	Quadratic	Tchebycheff
w1	4422.01	-890.60	1655.42	4422.01	-890.60	1655.42	0.00%	0.00%	0.00%
w2	4293.16	-17069.92	1520.28	4293.16	-17069.92	1520.28	0.00%	0.00%	0.00%
w3	4148.52	-51787.43	1351.36	4148.52	-51787.43	1351.36	0.00%	0.00%	0.00%
w4	4027.75	-87597.62	1182.44	4027.75	-87597.62	1182.44	0.00%	0.00%	0.00%
w5	3942.28	-113530.30	1013.52	3942.28	-113530.30	1013.52	0.00%	0.00%	0.00%
w6	3902.75	-123759.79	844.60	3902.75	-123759.79	844.60	0.00%	0.00%	0.00%
w7	3910.92	-116626.50	935.22	3910.92	-116626.50	935.22	0.00%	0.00%	0.00%
w8	3966.05	-91697.52	1091.09	3966.05	-91697.52	1091.09	0.00%	0.00%	0.00%
w9	4064.74	-56229.54	1246.96	4064.74	-56229.54	1246.96	0.00%	0.00%	0.00%
w10	4193.27	-19101.19	1402.83	4193.27	-19101.19	1402.83	0.00%	0.00%	0.00%
w11	4313.03	-1023.66	1527.53	4313.03	-1023.66	1527.53	0.00%	0.00%	0.00%