TWO-STAGE PORTFOLIO OPTIMIZATION WITH HIGHER-ORDER CONDITIONAL MEASURES OF RISK

By Sıtkı Gülten

A dissertation submitted to the

Graduate School - Newark

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

Graduate Program in Management

Written under the direction of

Dr. Andrzej Ruszczyński

and approved by

_____

_____

_____

_____

_____

Newark, New Jersey

October, 2014

© 2014

Sıtkı Gülten

ALL RIGHTS RESERVED

ABSTRACT OF THE DISSERTATION

Two-Stage Portfolio Optimization with Higher-Order Conditional Measures of Risk

by Sıtkı Gülten

Dissertation Director: Dr. Andrzej Ruszczyński

In this study, an application of novel risk modeling and optimization techniques to daily portfolio management will be described.

In the first part, I develop and compare specialized methods for scenario generation and scenario tree construction. The quality of multi-stage stochastic optimization models depends heavily on the quality of the underlying scenario model. First, multivariate GO-GARCH model is used to generate adequate number of scenarios. Then, five different methods, a multi-facility location based backward scenario tree generation method, and forward and backward modified K-Means and Two-Step Cluster methods are used to generate scenario trees. Next, these five methods are tested on two-stage portfolio problems with different number of scenario sets. Finally, a Monge-Kantorovich transportation model is developed to compare the probability distribution of the GARCH-generated scenarios with the probability distribution in the constructed scenario trees.

In the second part, I construct a two-stage stochastic programming problem with conditional measures of risk, which is used to re-balance the portfolio on a rolling horizon basis, with transaction costs included in the model. A conditional risk mapping approach will be used in the model so that information from the previous investment period can be used in the decision for the next investment period.

Artzner et al. introduced coherent risk measures that reflect the interests of risk-averse investors. I will use coherent risk measures, such as semideviation risk function of order two or higher in this study. Next, the risk-averse multicut method, which is an extension of Bender's decomposition and proposed originally for first-order risk measure by Miller and Ruszczyński, will be generalized to higher order risk measures in order to solve two-stage mean-risk portfolio problem. Performance of this method with the stated risk functions are evaluated on the scenario tree which is constructed in the first part.

In the third part, I present an extensive simulation study on daily returns of Dow Jones companies by using several versions of the methodology. We show that two-stage models outperform single-stage models in terms of long-term performance. We also show that using high-order risk measures are superior to first-order measures.

Acknowledgements

I would like to first express my sincere gratitude to my advisor, Dr. Andrzej Ruszczyński. I am very fortunate to be his student as a research and teaching assistant during my Ph.D work. I truly appreciated the continuous support, everlasting patience, encouragement and trust he gave me to pursue my Ph.D. degree. His advices are invaluable. Without his guidance this dissertation would not have been possible. I am also indebted to him for introducing me to this very interesting research topic.

I am very grateful to Dr. Michael N. Katehakis. His help and encouragement, especially during my job search, was substantial for me.

I want to thank Dr. Jonathan Eckstein for his helpful corrections and suggestions for the dissertation draft.

I also want to give my special thanks to the other members of my dissertation committee, Dr. Douglas H. Jones and Dr. Darinka Dentcheva. I want to thank them for their assistance and valuable comments during my study.

Many thanks to all my friends here and in Turkey, especially Emre, Yasemin, Gökhan, Koray, Mesut, Artun, Grace, Nilofar, Laurens, Rose, and Natalia. Life, especially during my Ph.D. study, would not have been so enjoyable without them.

I would like to thank to Gonçalo Filipe, Monnique Desilva and Luz Kosar for their help and the sincere, friendly working environment they have provided.

The exact six years it has taken to complete my dissertation has required support that extends beyond research. Last, but not the least, I would like to thank my parents, Ayşe and İbrahim Gülten, my sister Burcu, and my nephew Eymen. I

dedicate this dissertation to them.

Finally, I would like to acknowledge the funding sources that allowed me to complete my graduate study without any financial concern. I have been supported as a teaching assistant during the first four years of my Ph.D and part-time Instructor for the last two years by Rutgers Business School. This research has also been supported by National Science Foundation under the grant number DMS-1312016 and by Air Force Office for Scientific Research under the grant number FA9550-11-1-0164.

Sıtkı Gülten

Newark, NJ

August 11, 2014

Table of Contents

List of Tables

List of Figures

Chapter 1
Introduction

1.1 Motivation

The importance of portfolio optimization and risk management in both theoretical and practical finance have been increasing significantly. Markowitz developed the first important approach to the portfolio problem. In that study, he argued that portfolio performance can be measured by using the mean of the portfolio return and the variation of the return which is known as risk. In this mean-risk approach there are two type of objectives: minimizing the risk while having a fixed value of mean or maximizing the mean value while having a fixed value of risk. This model provided the basis for portfolio optimization research. However, standard deviation, which is used as a risk function in Markowitz theory, is not an appropriate choice since it penalizes overperformance return and it does not reflect the fat tails in loss distribution.

The main objective of this dissertation is to evaluate the usefulness of several risk modeling and optimization techniques for daily stock portfolio optimization.

The question of which risk function to use is examined extensively in the literature. In this study, risk functions with second-order stochastic dominance consistency relation or coherent risk functions, such as higher-order semi-deviation and weighted deviation from quantile, will be taken into consideration while formulating the portfolio problem.

Even though the portfolio optimization problem over multiple periods is studied in

the literature, the problem with an option to rebalance is not considered significantly. In the literature, the portfolio optimization problem with rebalancing is modeled by using a tree structure. However, the recent experiments showed that this structure is not efficient in larger instances, especially when there are many possible outcomes while rebalancing.

Scenario trees are developed to describe the uncertainty in processes such as asset returns in the portfolio problem. The quality of scenario trees is a very important factor in multi-stage stochastic optimization problems. Scenario trees with better quality usually bring multi-stage stochastic optimization models with better quality. Hence, this study will focus on constructing a better scenario tree based on Monge-Kantorovich metric for two-stage portfolio optimization problem so that portfolio optimization problem with rebalancing can be solved in a more time-efficient way when coherent risk measures are used.

Another important issue is how to solve a large-scale two-stage stochastic programming problem since the number of variables and constraints increase due to a large scenario tree structure. The problem becomes a large-scale nonlinear optimization problem when higher-order risk measures are used. In order to handle computational complexity and nonlinear optimization problem issues, a Risk-Averse Multicut method will be used.

1.2 The Portfolio Problem

In the portfolio optimization problem in its simplest form, the return rates of n assets are represented by an n-dimensional random vector R, with R_j denoting the return rate of asset $j = 1, \ldots, n$. The n-dimensional vector z represents the distribution of the capital among assets: z_j is equal to the fraction of the capital invested in asset j.

The total return rate of the portfolio at the end of the investment period is

$$R^\top z = \sum_{j=1}^{n} R_j z_j.$$

The portfolio problem is to find an "optimal" way to distribute the initial capital among the n assets, under the condition that $z \in Z$, where $Z \subset \Re^n$ is a convex and compact set of feasible asset allocations. As the return rates of the assets are random, the portfolio return rate is a random variable, and thus the meaning of "optimal" depends of the modeling approach.

Risk is defined as the chance of loss connected with a given action in Brachinger and Weber [9]. Investments with higher potential returns are usually associated with higher risks. Therefore, investors need to make tough decisions to maximize their returns while limiting the amount of risk. This brings the risk and return trade-off problem. In the next section, we will provide a brief history of risk measures in the following part. In order to give a basic history on risk measures, we will divide the timeline of risk measures into three stages:

1. Markowitz Portfolio Theory based risk measures
2. Value at Risk and related risk measures
3. Coherent Risk Measures

1.2.1 Markowitz Portfolio Theory based risk measures

In a pioneering study, Markowitz [29] argued that portfolio performance can be measured by using two scalar characteristics: the mean of the portfolio return, $E\left[R^\top z\right]$, and the variance of the return, $\mathrm{Var}\left[R^\top z\right]$, which characterizes its riskiness. We can then minimize the variance for a fixed value of the mean, or maximize the mean, while keeping the variance bounded. Since then, numerous theoretical and practical

studies evaluated the usefulness of the mean–variance approach in portfolio optimization. Markowitz used standard deviation as the risk function in mean-risk portfolio problem. By using the standard deviation, the mean-risk portfolio problem can be formulated as parametric quadratic programming problem so that standard solution techniques can be used. The most important criticism to standard deviation as a risk function is that it penalizes the overperformance equally to underperformance. Also, it does not account for fat tails in loss distribution. Fat-tailed distributions usually have large skewness and kurtosis compared to normal distribution. When the distribution of returns are fat-tailed, the probability to have larger losses and gains is higher than a normal distribution. That's why a fat-tailed distribution is important to consider the up and down movements in financial markets. In order to eliminate the penalty on overperformance, downside risk measures were developed.

1.2.2 Value at Risk and related risk measures

It is well known that financial crises are usually followed by regulatory responses. In the early 1970s, risk in financial institutions, such as investment banks, increased due to derivative markets and floating exchange rates. Therefore, the SEC refined capital requirements with a new rule called Uniform Net Capital Rule (UNCR). This rule divided the financial assets that financial institutions held into twelve classes based on risk level. UNCR required each new class to have different capital requirements ranging from 0% in short term treasuries to 30% in equities.

The first step to create Value at Risk (VaR) came with the regulatory measures proposed in 1980. In this regulatory measure, the SEC brought a requirement that the capital requirements of financial firms must be linked to the losses that would incur over a thirty day interval in different security classes with a confidence interval of 95%. Financial institutions used the historical returns to compute these potential losses calculations. Since the portfolios of financial institutions started to become

more volatile during 1980s, a sophisticated risk measure is required to solve this problem. Actually, most of the financial institutions developed basic measures of Value at Risk by the early 1990s to compute the maximum loss they could get from their investments. However, there was not any unified risk measure amongs all institutions. After the collapse of Barings, the oldest British investment bank, as a result of disastrous loss associated with derivatives and leverage, firms accelerated the search for a more comprehensive risk measure. J. P. Morgan initiated the first step to develop a risk measure in this direction. In 1995, J.P. Morgan provided data on the variances and co-variances across different security and asset classes that it used to manage the risk in their internal operations. It named this service as "RiskMetrics" and used the term Value at Risk to describe the risk measure used. Many commercial and investment banks welcomed this new risk measure. Then, the Basel Committee on Banking Supervision [52], which is responsible for international banking regulations, developed a market risk capital requirement based on VaR in 1995. Nowadays, VaR has being widely used by financial and non-financial firms.

Value at Risk measures the potential loss in value of an asset over a given time horizon with a given confidence interval. Let random variable X represent the loss; for a given probability level $\alpha \in (0, 1)$, VaR_α measures the minimum loss incurred in the α percent worst cases of a portfolio.

We will start with a basic example of VaR. If we have a portfolio with a VaR of \$1,000 for one day with a confidence level of 95%. This means that there is a 5% probability that the portfolio's loss will be more than \$1,000 in one day. In other words, we can say that there will be a loss of \$1,000 or more in one of the next 20 days.

Acerbi and Tasche [1] defined VaR_α at probability level α as follows:

$$\mathrm{VaR}_\alpha(X) = -x^\alpha \tag{1.1}$$

where the upper quantile x^{α}

$$x^{\alpha} = \sup\{x : P[X \leq x] \leq \alpha\}. \tag{1.2}$$

VaR satisfies the Law Invariance property. Cheng et al. [10] defined that VaR satisfies this property as follows:

$$\text{If } P[X \leq t] = P[Y \leq t] \ \forall t \in \mathbb{R}, \text{ then } \rho[X] = \rho[Y]; \tag{1.3}$$

The financial interpretation of law invariance property is if two random variables are identically distributed, law invariant risk measures allocate the same riskiness to financial positions.

There are mainly four different approaches to compute Value at Risk: VaR Historical Simulation, VaR Parametric Approach, VaR Monte Carlo Simulation, and VaR Variance-Covariance Method. Details of Value at Risk and the approaches to compute VaR can be found in [14], [33].

1.2.3 Coherent Risk Measures

Further improvement was made by considering more general mean–risk models, with different measures of variability [24]. By considering consistency with stochastic dominance, the papers [34, 35, 36], introduced a family of mean–semideviation models, which are particularly useful for portfolio models (see, e.g., [28, 48]).

Generally, semi-moments can be expressed as follows:

$$E[\max(c - X, 0)^p] \tag{1.4}$$

The parameter c is the target for below which outcomes are penalized. The parameter p is the relative impact of the deviations. When $c = E[X]$ and $p = 1$, it is

known as semi-deviation and when $c = E[X]$ and $p = 2$, it is known as semi-variance.

Most recently, axiomatic models of risk have been studied extensively. Two important axiomatic models are second-order stochastic dominance theory and coherence axioms.

Whitmore [55] used stochastic dominance based on an axiomatic model of risk-averse preferences. Hanoch and Levy [18], and Rothschild and Stiglitz [42] extended stochastic dominance to general distributions. In the second-order stochastic dominance approach, risk-averse preference models with larger outcomes are consistent with the second-order stochastic dominance relation. And, a risk-averse investor's preferences are described as concave nondecreasing utility function.

Definition 1.1. X dominates Y in the second order, $X \geq_{SSD} Y$, if

$$E[u(X)] \geq E[u(Y)]$$

for every concave non-decreasing function $u(\cdot)$.

In this case, any risk-averse investor would prefer position X over Y. The consistency relation between mean-risk models and second-order stochastic dominance is analyzed in [34], [35], and [36]. A mean-risk model is *SSD consistent* with a constant γ, for all X and Y if the following relation holds,

$$X \geq_{SSD} Y \Rightarrow E[X] - \gamma \cdot r(X) \geq E[Y] - \gamma \cdot r(Y) \tag{1.5}$$

Mean-risk model with absolute semideviation risk function is found to be SSD consistent when $\gamma = 1$ in [34].

In the last decade, axiomatic models of risk have been studied extensively, in particular, *coherent risk measures*, introduced by Artzner et al. [4]. A coherent measure of risk satisfies the following four properties: Convexity, Monotonicity, Translation

Property, and Positive Homogeneity. In this chapter, while formulating a general two-stage stochastic programming maximization model, uncertain outcomes will represent profits.

Definition 1.2. Let $\mathcal{X}$ be the space of all uncertain outcomes, $\mathbb{1}$ denotes the sure gain of 1 and $X = (R^T z)$. A coherent measure of risk is a functional $\rho : \mathcal{X} \to \Re$ which satisfies the following axioms.

Convexity: $\rho(\alpha X + (1-\alpha)Y) \leq \alpha\rho(X) + (1-\alpha)\rho(Y), \quad \forall X, Y \in \mathcal{X},\ \alpha \in [\prime, \infty]$;

Monotonicity: If $X, Y \in \mathcal{X}$,and $X \leq Y$, then $\rho(X) \geq \rho(Y)$;

Translation Invariance Property: If $a \in \Re$ and $X \in \mathcal{X}$, then $\rho(X + a\mathbb{1}) = \rho(X) - a$;

Positive Homogeneity: If $\beta \geq 0$ and $X \in \mathcal{X}$, then $\rho(\beta X) = \beta\rho(X)$.

While formulating risk-averse two-stage portfolio problem in Chapter 4, uncertain outcomes will represent cost. In the following definition, the uncertain outcomes X and Y represent losses, and $\mathbb{1}$ denotes the sure loss of 1. We can rewrite the axioms as follows:

Definition 1.3. Let $X = -R^\top z$. Coherent risk measures are functionals $\rho : \mathcal{X} \to \Re$ defined on a suitable vector space $\mathcal{X}$ of random outcomes, which satisfy the following axioms:

Convexity: $\rho(\alpha X + (1-\alpha)Y) \leq \alpha\rho(X) + (1-\alpha)\rho(Y), \quad \forall X, Y \in \mathcal{X},\ \alpha \in [\prime, \infty]$;

Monotonicity: If $X, Y \in \mathcal{X}$, and $X \leq Y$, then $\rho(X) \leq \rho(Y)$;

Translation Invariance Property: If $a \in \Re$ and $X \in \mathcal{X}$, then $\rho(X + a\mathbb{1}) = \rho(X) + a$;

Positive Homogeneity: If $\beta \geq 0$ and $X \in \mathcal{X}$, then $\rho(\beta X) = \beta\rho(X)$.

The inequality in the monotonicity axiom is understood in the almost sure sense.

We will explain each axiom briefly in intuitional terms applied to finance. Convexity is the most important axiom because it makes sure that a coherent risk measures takes into account portfolio diversification. The axiom says that investing both portfolio X and Y will have an overall lower risk than the sum of the risks in investing

portfolio X and portfolio Y separately. The monotonicity axiom means higher risks are associated with higher losses. The translation invariance property means that a riskless bond that is added to a portfolio X will have no loss with a probability of 1. Therefore, the initial amount invested will be always received. The reason is initial investment amount is added because risk measures measure loss as a positive amount. The positive homogeneity axiom states that risk cannot be increased or decreased by investing different amounts in the same stock. In other words, risk does not depend on the quantity purchased.

Mean-risk models with the variance and standard deviation risk functions are found to be not coherent in Acerbi and Tasche [1] because they fail to satisfy the monotonicity axiom. In the same study, they also found that Value-at-Risk is not coherent because it does not satisfy convexity axiom of coherent risk measures. This means higher risk can result from diversification.

Important examples of coherent risk measures are models of the form

$$\rho(x) = E[X] + \gamma r[X], \tag{1.6}$$

where the risk measure $\rho[\cdot]$ represents cost, $r(\cdot)$ is the upper semideviation of order $p \geq 1$, given in equation (1.7), or a weighted mean-deviation from quantile in equation (1.8):

$$r(X) = E[(X - E[X])_+^p]^{1/p}, \tag{1.7}$$

$$r_\alpha(X) = \min_\eta E\left[\max\left(\frac{1-\alpha}{\alpha}(X-\eta), \eta - X\right)\right], \quad \alpha \in (0,1). \tag{1.8}$$

For these both cases, when $\gamma \in [0,1]$, the mean-risk model is coherent [47]. If we have a maximization problem where $\rho[\cdot]$ represents profit, we will use the lower semideviation of order $p \geq 1$, given in equation (1.9) or a weighted mean-deviation from quantile in equation (1.10).

$$r(X) = E[(E[X] - X)_+^p]^{1/p}, \tag{1.9}$$

$$r_\alpha(X) = \min_\eta E\left[\max\left(\frac{1-\alpha}{\alpha}(\eta - X), X - \eta\right)\right], \quad \alpha \in (0,1). \tag{1.10}$$

Since VaR is not coherent, a few coherent equivalent VaR risk measures are proposed. These are TVaR (tail value at risk) by Artzner et al. [6], WCE (worst conditional expectation) by Inoue [23], and CVaR (conditional value at risk). CVaR became a very popular risk measure since it is very similar to VaR. A very basic intuitive explanation of CVaR is that it can find how bad things can get if the VaR loss is exceeded. CVaR will let an investor to compute the expected loss given that VaR loss is exceeded. Rockafellar and Uryasev [40], [41] defined CVaR as:

$$\mathrm{CVaR}_\alpha[X] = \frac{1}{\alpha}\int_0^\alpha \mathrm{VaR}_\beta[X]; d\beta$$

Next, we will show an intuitive representation of CVaR_α. First, we will define the generalized inverse function of $F(x)$ as follows:

$$F_X^{-1}(\beta) = \inf\{x : F(x) \geq \beta\}$$

Rockafellar and Uryasev [40], [41] showed that CVaR_α can be written as follows:

$$\mathrm{CVaR}_\alpha[X] = \frac{1}{\alpha}\int_{1-\alpha}^1 F_X^{-1}(\beta)\, d\beta$$

Then, it can be expressed that CVaR (see, [40, 41]) is related to the deviation from quantile (1.8) by the formula (cf. [49, sec. 6.2.4])

$$E[X] + r_\alpha[X] = \mathrm{CVaR}_\alpha[X] = \frac{1}{\alpha}\int_{1-\alpha}^1 F_X^{-1}(\beta)\, d\beta = \min_\eta \left\{\eta + \frac{1}{\alpha}E\big[(X-\eta)_+\big]\right\}.$$

Here $F_X^{-1}(\cdot)$ is the quantile function of X.

In CVaR, portfolio weights can easily optimized by linear programming formulations (see, [40]) to minimize CVaR.

Another research direction in risk measures is copula functions. The importance of dependencies between stocks made the copula functions popular in risk management. Copula functions map a set of marginal distributions into a multivariate distribution and vice versa. There are a few popular copulas such as Gaussian copula by Frey et al. [17] and Clayton copula by Cuvelier et al. [13]. However, copulas to find the dependency between stocks are not extensively studied in the literature.

1.3 One-Stage and Two-Stage Portfolio Problems

We can formulate the general one-stage portfolio problem with a risk measure $\rho(\cdot)$ as an objective function as follows:

$$\min_{z \in Z} \rho[-R^\top z]. \tag{1.11}$$

where

$$\rho(-R^T z) = -E[R^T \cdot z] + \gamma \cdot r[-R^T \cdot z] \tag{1.12}$$

In this study, we adapt the convention that the argument of the risk measure $\rho[\cdot]$ represents cost (losses) and that is why we use the minus sign in front of the return rate, and vector z represents the distribution of the capital among assets. The mean-risk approach is first introduced by Markowitz [29]. The term $E[R^T \cdot z]$ is the expectation of the portfolio return, and the term $r[-R^T \cdot z]$ is a measure of the uncertainty of the portfolio return. A fundamental modeling issue is to choose the risk function $r(\cdot)$ used in this model. For the measures of risk (1.7) with $p = 1$ and (1.8),

the resulting optimization problem (1.11) is a linear programming problem, which can be efficiently solved by specialized techniques [28]. The parameter γ in (1.12) represents the risk aversion constant. If γ is equal to 0, the investor is a risk-taker and chooses his or her portfolio based on only performance of asset returns without considering the risk associated with assets. If γ is equal to 1, the objective function is from a most risk-averse investor's perspective. A risk-averse investor chooses his or her portfolio with a more emphasis on the risk associated with assets, such as assets with lower returns and lower risks. Parametric methods of [48] allow for generating a family of solutions, corresponding to a range of values of the parameter γ in (1.6).

In most recent and related study, Miller [32] formulated two-stage portfolio optimization problem with coherent risk measures as a linear programming problem solved using Bender's decomposition technique for only $p = 1$ in (1.7).

So far, we discussed the one-period portfolio problems with risk measures. However, if a portfolio optimization model is used in a rolling horizon fashion, as in [30], with re-balancing in regular time intervals, it makes sense to include the re-balancing action and associated transaction costs into the model. In this study, we will examine the portfolio optimization problem when there is an option to rebalance. And, in this type of portfolio problem, some information may only become available at some interim time period.

Ruszczynski and Shapiro [46], [49] found that such information may change an investor's perception of risk from the previous investment period, and develop conditional risk mappings to model this change in perception. Additionally, Artzner et al [4] and Riedel [38] have developed similar axioms for one-period coherence axioms.

To address this issue in the simplest possible way, a *two-stage model* can be formulated. In this model, an option to re-balance the portfolio at the end of the first period is available. Let us denote by R_j^t the return rate of asset $j = 1, \ldots, n$ in stage $t \in \{1, 2\}$. Asset allocations are denoted by n-dimensional vectors z and y, where

z_j represents the amount of capital invested in asset j at the first stage, and y_j the amount invested at the second stage. The vector y may depend on the observations gathered in the first stage. The end portfolio value in the first stage is given by $(\xi^1)^\top z$ and the end value at the second stage is $(\xi^2)^\top y$, where

$$\xi^t = \mathbb{1} + R^t,$$

with $\mathbb{1}$ denoting the sure outcome of 1. The random vectors ξ^1 and ξ^2 are, in general, dependent. If they have finite numbers of realizations, the most transparent way is to represent them in a form of a *scenario tree*. An example of such a tree is depicted in Figure 1.1.

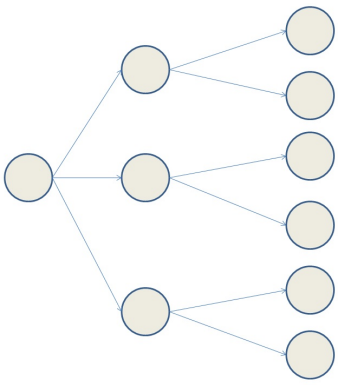

Figure 1.1: Scenario Tree

The nodes at stages one and two represent realizations of ξ^1 and ξ^2, respectively. The node at stage zero is known as the root node and represents the beginning of the process. Each node at stage one represents a different realization of ξ^1. It is connected to a set of children nodes at stage two, which represent possible outcomes of ξ^2, following the first stage outcome. With each arc of the tree, a probability is associated. Probabilities of arcs leading to nodes at stage one are the probabilities of realizations of ξ^1. The probabilities of arcs leading to nodes at stage 2, are conditional probabilities of realizations of ξ^2.

The two-stage portfolio problem allows us to model the re-allocation option within an optimization problem. In the first stage, asset allocations z are to be determined. Then a realization of ξ^1 is observed, and the allocations can be changed to y. In the scenario tree setting, there can be a different value of y for each node at level 1. Finally, the realization of ξ^2 is observed. As a result, the final portfolio value can be calculated. Such an approach, with the use of dynamic measures of risk, has been first developed in [32, 31]. We formally define the two-stage model in Chapter 3.

1.4 The Risk-Averse Portfolio Problem

In this section, first, a general linear programming model will be explained for the one-period portfolio problem. Then, the model will be extended to two investment periods. Finally, conditional risk measures used in both models will be explained.

We can formulate general nonlinear programming model for one period problem as:

$$\begin{aligned} \max \;\; & E[R(z)] - \gamma \cdot r[R(z)] \\ \text{s.t.} \;\; & \sum_j z_j = C \\ & z_j \geq 0, \;\; \forall j \end{aligned} \tag{1.13}$$

where C is the capital and r is the mean lower semideviation or mean weighted deviation from quantile. Next, we will explain the parameters and variables before the formulation the risk-averse portfolio problem.

$\hat{r}_{ij}$: return of asset j in the scenario i

p_i : probability of scenario i

z_j : the amount in dollars invested in asset j

$$[R(z)]_i = \sum_j z_j \hat{r}_{ij}$$

$$E[R(z)] = \sum_i \sum_j p_i z_j \hat{r}_{ij}$$

$E[R(z)]$ is the expected return for the given all assets and $[R(z)]_i$ is the return in scenario i with probability p_i, and γ is defined as risk aversion constant. We can use any coherent risk measure to formulate the risk-averse portfolio problem. The risk averse portfolio problem based on mean-risk model with the upper semideviation risk measure r of a random variable $X = -R^T z$, when the risk measure $\rho(X)$ represents cost is defined as

$$\rho(X) = E[X] + \gamma \cdot r[X], \tag{1.14}$$

where

$$r[X] = E[(X - E[X])_+^p]^{1/p} \tag{1.15}$$

Since it was shown in [47] that $\rho(X)$ is consistent with second order stochastic dominance [34] and coherent when $\gamma \in [0, 1]$, the portfolio problem can be written as,

$$\min - \left\langle p, R^T z \right\rangle + \gamma \sum_i p_i \max(\langle \hat{r}_i, z \rangle - \left\langle p, R^T z \right\rangle, 0)$$

where p_i is the probability of the i^{th} outcome of random variable X, and $\hat{r}_i$ is the

vector of asset returns of outcome i. Then, we can convert this problem into a linear programming problem and apply linear programming methods to solve the problem.

In the two-stage portfolio problem, we are interested in case where there is a possibility to rebalance the portfolio in between the two time period.

In the two-stage portfolio problem, there are n assets, and the return of the assets in each stage is an n-dimensional random variable. R_j^t represents the return of the asset j in stage t, where $t \in \{1, 2\}$. The asset allocation for the first and second stages are denoted by n-dimensional vectors z and y, respectively, where z_j represents the amount of capital invested in asset j during the first-stage, and y_j represents the amount of capital invested in the second-stage. The capital at the end of the first stage is $(\xi^1)^T \cdot z$ where $\xi^1 = \mathbb{1} + R^1$.

In the decision process of the two-stage portfolio problem with rebalancing, first we will decide z and observe the realization of ξ^1 for the first stage, and then we will do the same for the second stage.

Let p^1 be the probability vector, where p_i^1 is the probability of outcome i in the first stage, and p_i^2 be the probability vector in the second stage for each node i in the first stage where p_{il}^2 is the probability of moving to node l in the second-stage from node i.

The two-stage stochastic programming problem can be formulated as

$$\begin{aligned} \min \ & c^T z + \rho_1[Q(z)] \\ \text{s.t.} \ & Az = b \\ & z \geq 0 \end{aligned} \tag{1.16}$$

where $Q_i(z)$ is the optimal value of the i^{th} second stage problem, ρ_1, is a coherent risk

measure, and $Q(z)$ is the random variable taking the value $Q_i(z)$ with probability p_i^1

$$Q_i(z) = \min c_{2i}^T \cdot y_i + \rho_{2i}(-\xi_i^2 \cdot y_i)$$

where we can use conditional semideviation,

$$\rho_{2i}(-Z) = -E_{p_i^2}[Z] + \gamma_i \cdot E_{p_i^2} \max((Z - E_{p_i^2}[Z]), 0),\ \gamma_i \in [0, 1] \tag{1.17}$$

where $Z = \xi_i^2 \cdot y_i$.

Moreover, by using the Lagrangian duality, the obtained stochastic program can be formulated as a Lagrangian dual model. Benders' Decomposition technique can be used to solve the Lagrangian dual. We will explain Benders' decomposition in more detail in Chapter 3.

In Chapter 2, several scenario tree generation methods are described. First, a multivariate GARCH model is used to generate an adequate number of scenarios to model the random returns. Then, three different tree construction algorithms are developed: K-Means and Two-Step cluster algorithms in forward and backward forms, and a backward multi–facility location algorithm. In order to evaluate the quality of the scenario trees, Monge–Kantorovich transportation model is formulated to compare the probability distributions of the "original" probability distribution (empirical distribution supported on the scenarios generated) with the probability distributions supported on the constructed scenario trees.

In Chapter 3, a two-stage portfolio problem with an option to re-balance is modeled by using higher-order conditional risk measures. A risk-averse multicut method is proposed to solve this model.

In Chapter 4, computational results will be presented to compare the scenario

generation techniques based on Monge-Kantorovich metric. Then, a simulation analysis will be presented to compare the performance of various portfolios with the Dow Jones index.

Finally, at the conclusion of this study, an overall summary of our contribution and a list of some possible future research directions will be presented.

Chapter 2
Scenario Tree Generation Problem

2.1 Literature Review

A substantial body of literature exists about generating scenario trees for stochastic optimization models. Heitsch and Römisch [19] proposed a theory-based heuristics for generating scenario trees from an initial set of scenarios, and applied these heuristics in electric power management. Their proposed heuristics have a recursive scenario reduction algorithm and also bundling steps based on forward or backward scenario tree generation methods. They used the stability result in multi-stage stochastic programs from the study in Heitsch, Römisch and Strugarek [20] to compare the closeness of the original probability distribution to its scenario tree approximation. The conditions on the initial approximation in applications is constructed from a discrete probability distribution by using a sampling method or from a general probability distribution by using discretization schemes. However, the algorithm can be used as a heuristics for scenario tree generation in other applications.

Hochreiter and Pflug [21] showed that the problem of obtaining accurate and valuable scenario tree approximations can be viewed as the problem of optimally approximating a given distribution by using a distance function. In that paper, it is found that the best approach is to use the Wasserstein distance in tree approximation. The resulting optimization problem can be formulated as a multi-dimensional facility location problem, and then well-known heuristic algorithms for multi-facility location problems can be applied. They also showed that a scenario tree is constructed as a

nested facility location problem to use in multi-stage stochastic programs. A multi-stage stochastic mean-risk financial programming problem is used to test the model. They concluded that if the objective of the approximation is to achieve a controlled matching of certain moments and a controllable coverage of heavy tails, scenario tree generation based on multidimensional facility location will be the best fit.

Our approach builds on these contributions, with the intention to be able to handle huge trees arising in financial applications.

2.2 Scenario Generation

Financial data are insufficient for construction optimization models based on empirical distributions alone. It is imperative to generate scenarios that were not observed in practice, but are possible according to statistics. Since investors hold many risky assets in their portfolios, they have to assess the possibility of severe losses. Therefore, we need a fat-tailed distribution for financial asset returns that will consider both larger losses and gains. For this purpose, we need a model for scenario generation, and our model should take fat tails into account.

The first step in scenario generation is to obtain asset returns data to construct scenarios. We want to obtain data that will include both up and down movements in the market. In order to incorporate this idea into our model, daily stock price data of the 30 companies included in the Dow Jones index are obtained for a period of three years, from September 2, 2008, to November 30, 2011. This period includes both the financial crisis in 2008 and recovery period after the crisis. After the data is obtained, we will use two consecutive daily returns of a stock vector to create one scenario.

Next, in order to model the probabilistic information in the data, an adequate number of scenarios must be generated. As we are interested in an adequate modeling of the tail behavior and co-movements of returns, multivariate GARCH models

appear to be particularly useful. Multivariate GARCH models parametrize the covariance matrix by using large number of parameters at a minimum loss of generality. However, since these parameters are hard to estimate, it brings convergence issues for the estimation algorithms. The selection of the multivariate GARCH model usually depends on practical needs such as easy estimation of the model and easy interpretation of the model parameters. Multivariate GARCH models also should be flexible to represent dynamics of both conditional variances and covariances. Silvennoinen and Terasvirta [51] divided multivariate GARCH models in four categories based on parametric formulations: Models of the conditional covariance matrix such as VEC and BEKK models, factor models such as O-GARCH and GO-GARCH models, models of conditional variances and correlations such as Bollerslev's CCC model, and nonparametric and semiparametric models.

In this study, we will use GO-GARCH model which is one of the factor models. Economic theories are the most underlying principle behind factor models. Engle et al. [16] introduced the first factor GARCH model by assuming that the observations are generated by factors which are conditionally heteroskedastic. Then, Alexander and Chibumba [2] introduced the O-GARCH model which is mainly used to model conditional covariances of the financial data. Since O-GARCH model can remain feasible for large covariance matrices, it has been a popular choice in the financial world. In the O-GARCH model, observed data can be linearly transformed into a set of uncorrelated components by using an orthogonal matrix. However, O-GARCH model can have issues to identify an orthogonal matrix when the data has weak correlation.

GO-GARCH model is a generalization of the O-GARCH model. In the GO-GARCH model, observed data can be linearly transformed into a set of uncorrelated components by using a matrix that is constant over time and invertible. However, this matrix does not have to be an orthogonal matrix. It is not possible to have

fewer factors than number of assets in the GO-GARCH model. GO-GARCH model can parametrize large covariance matrices with a large degree of freedom and avoid convergence difficulties of estimation algorithms.

In this study, we will use multivariate GO-GARCH(1,1) [53] model to generate scenarios. Its structure can be summarized as follows.

We assume that the observed vector-valued time series $\{x_t\}$ (of dimension $m = 30$) is a linear combination of unobserved m-dimensional normal vectors $\{y_t\}$ having uncorrelated components, that is,

$$x_t = Zy_t, \quad t = 0, 1, 2, \ldots$$

The square matrix Z is assumed to be constant over time and invertible.

Unobserved components have a diagonal covariance matrix $H_t = \text{diag}\{h_{i,t},\ i = 1, \ldots, m\}$, and thus the covariance matrix of x_t is $V = E\big[x_t x_t^T\big] = ZH_tZ^T$. The crucial component of the model is the evolution of the diagonal elements $h_{i,t}$ of the covariance matrix H_t:

$$h_{i,t+1} = (1 - \alpha_i - \beta_i) + \alpha_i y_{i,t}^2 + \beta_i h_{i,t}, \quad i = 1, ..., m, \quad t = 0, 1, 2, \ldots,$$

where the initial matrix $H_0 = I$.

The historical data are used to calculate the least-squares estimates of the matrix Z and the coefficients (α_i, β_i), $i = 1, \ldots, m$.

Once the model is constructed, it can be used to generate an arbitrary number of scenarios. Assuming that the data were collected for the period $t = 0, 1, \ldots, T$, we generate scenarios for times $T+1, T+2, \ldots$. In our case, we use only two next steps, that is, $T + 1$ and $T + 2$, to prepare scenarios for ξ^1 and ξ^2.

2.3 Scenario Tree Generation Problem

In stochastic programming, values of some parameters are not known, and these values are replaced by some probability distributions. Therefore, to solve stochastic programming models, we need to know the description of the stochasticity. Stochastic programs can only handle discrete samples of limited size. However, the number of scenarios for a problem can be very large. In order to solve this problem, approximated distributions need to be used, and to approximate, scenario trees will be used. In this way, it is easier to deal with large sized scenario models.

Raw scenarios are not suitable for two-stage optimization models, because after the first-stage, while deciding about allocations y for the second-stage, we would know not only the past return realizations ξ^1, but also the future realizations ξ^2 (see the left part of Figure 2.1). Constructing a scenario tree eliminates this problem as it can be seen in the right part of Figure 2.1.

Definition 2.1. Basic Definitions:
A *scenario* is a path from root node to a leaf node.
A *stage* is a moment when the decisions are taken.
A *period* is a time interval between two stages.

There are two different ways to design scenario tree generation methods, forward methods and backward methods. In a forward tree construction, starting at the first stage, one merges selected nodes into clusters and moves forward until the last stage. Forward tree construction is explained in Figures 2.1 and 2.2. In a backward tree construction, one starts at the last stage, merges selected nodes into clusters and this will join all their predecessors as well. In this way, we move backward until the first stage. Backward tree construction is explained in Figures 2.3 and 2.4.

In scenario tree generation, it is important to maintain probability information while reducing the number of scenarios, that is, to assign to a scenario representing

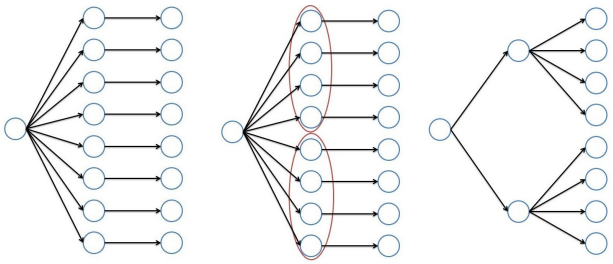

Figure 2.1: First-Stage Clustering in a Forward Tree Construction

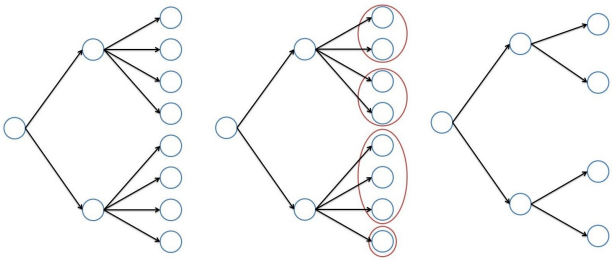

Figure 2.2: Second-Stage Clustering in a Forward Tree Construction

a group of scenarios the sum of their probabilities.

Since, the quality of multi-stage stochastic optimization models depends heavily on the quality of the underlying scenario model, this study will focus on constructing a scenario tree for two-stage portfolio optimization problem so that portfolio optimization problem with rebalance can be solved in a more time-efficient way when coherent risk measures are used.

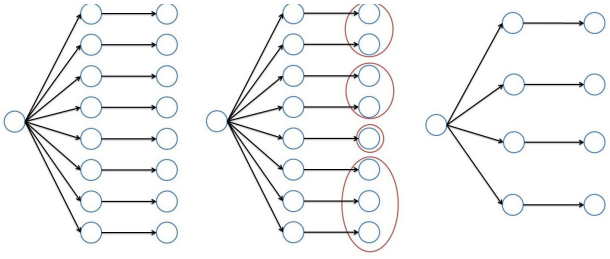

Figure 2.3: Second-Stage Clustering in a Backward Tree Construction

Five different scenario generation methods are used in this study. K-means and

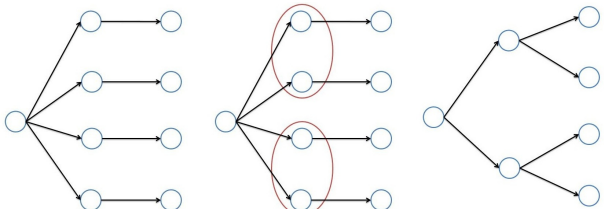

Figure 2.4: First-Stage Clustering in a Backward Tree Construction

two-step clustering methods in forward and backward versions, and a backward scenario generation method based on the idea of a multi–facility location problem. Then, these five models will be compared by using a mass-transportation problem.

2.3.1 Multi–Facility-Location-Based Scenario Tree Generation Method

The multi–facility location problem is defined as a way to locate a number of facilities to serve a given number of customers optimally when their locations are given. Each customer is assigned to only one facility. The idea is to determine the locations of the facilities so that the sum of the weighted distances from the facilities to assigned customers is minimized.

The clustering problem is to group a given set of objects with k attributes into clusters so that objects in the same cluster are more similar than other clusters and objects in different clusters are dissimilar. The objects are represented as points in the k-dimensional Euclidean space.

If we define the set of customers served by the same facility as a cluster, the multi–facility location problem can be formulated as a specialized clustering algorithm. However, the issue of locating the facilities optimally within their clusters in the multi–facility location problem does not exist in the clustering problem. Also, the multi–facility location problem is usually a two-dimensional problem. On the other hand, the clustering problems are k-dimensional where k is the number of attributes.

We will use a hybrid method which is composed of two different algorithms within a backward scenario tree generation method. In the first part, a nearest centroid based heuristic is used to form a given number of clusters in the second stage.

Choosing initial centroids is very important in a clustering algorithm since bad initialization can lead to poor results. On the other hand, a better initialization method can lead to an optimal solution in a shorter period of time. The fastest and easiest way to choose initial centroids is pure random. However, it will very likely lead to poor results since it can choose these initial centroids close to each other. In the initialization part of our algorithm, we want to choose the initial centroids that are spread out from each other. We will use a very similar initialization approach as in [3]. During the iteration part, we will compute the centroid (cluster mean) as centers by using Weiszfeld method [54]. This method is a parallel heuristic method similar to Cooper's method [12]. Since it is possible that Weiszfeld method can fail to converge when calculated centroid falls on one of the data points given, we will handle this issue by choosing that data point as centroid. In this method, cluster centroids can be computed in parallel subroutines. Therefore, this method requires less CPU time. Nearest centroid based heuristic is explained in Algorithm 1.

Algorithm 1 Nearest Centroid Heuristic

Initialization:
The first cluster mean is chosen randomly from the data points.
Next cluster mean is chosen from the remaining data points with a probability given to each point based on squared distance from its closest existing cluster mean.
Iteration:
Find the center for each cluster.
For each node, calculate the distance to the each center constructed before. If there is a closer center for that node, reassign the selected node to its closest center.
Continue until no new reassignments.

The distance measure used in this algorithm is as follows:

$$d_{ij} = \sqrt{\sum_{n=1}^{30} [(r^1(i,n) - r^1(j,n))^2 + (r^2(i,n) - r^2(j,n))^2]}, \tag{2.1}$$

where r^1 represents the return rate at the first stage, r^2 represents the return rate at the second stage, i and j represent the scenarios, and n represents the security.

In the second part of this hybrid method, in order to aggregate first-stage nodes, a multi–facility location problem is formulated.

Notation

We will set the network topology used in this facility-location problem as follows:

We denote by J the total number of scenarios (after the aggregation by Algorithm 1 and by $I < J$ the desired number of first-stage nodes. For $i, j = 1, \ldots, J$ we use d_{ij} to denote the distance calculated according to formula (2.1).

Decision Variables

$$x_{ij} := \begin{cases} 1, & \text{if node } i \text{ is assigned to scenario } j \\ 0, & \text{otherwise} \end{cases}$$

$$\forall i, j \in J, i \neq j$$

$$v_i := \begin{cases} 1, & \text{if scenario } i \text{ is used as a first-stage node} \\ 0, & \text{otherwise} \end{cases}$$

$$\forall j \in I$$

The multi-facility location problem can be formulated as follows:

$$\min \sum_{i=1}^{J}\sum_{j=1}^{J} d_{ij}x_{ij}$$

$$\text{s.t.} \sum_{i=1}^{J} x_{ij} = 1, \quad \forall j = 1, \ldots, J, \tag{2.2}$$

$$\sum_{i=1}^{J} v_i = I, \tag{2.3}$$

$$x_{ij} \leq v_i, \quad \forall i, j = 1, \ldots, J, \tag{2.4}$$

$$\text{all } x \text{ and } v \text{ variables are binary.} \tag{2.5}$$

Constraint (2.2) ensures that each node in the second-stage is connected to one of the node in first-stage.
Constraint (2.3) ensures that only I number of nodes are selected in the first-stage.
Constraint (2.4) ensures that if a node i in the second stage is connected to a node j in the first-stage, then node j must be selected in the first-stage.

It is a large-scale problem, and we solve it by a greedy method using the linear programming relaxation, in which the binary variables x and v are allowed to take any values in [0,1]. After a relaxed problem is solved, all $v_i = 1$ are permanently fixed; if none is equal to 1, we choose the one that is closest to 1, and fix it at 1. After that, a reduced problem with a smaller number of variables is solved, etc. Our experience indicates that this procedure does not lead to significant errors and allows for processing large data sets.

2.3.2 K-Means Scenario Tree Generation Method

K-means is a clustering method in which a set of n observations $S = \{x_1, x_2, ..., x_n\}$ are partitioned into k clusters ($k \leq n$). Each scenario is a d-dimensional vector, where $d = 30$ in this study (the number of Dow Jones stocks). Let $S = S_1 \cup S_2 \cup \cdots \cup S_k$

be the partition of the set; the objective of the K-means method is to minimize the sum of squares within clusters:

$$\begin{aligned} \min_{S_1,\ldots,S_k} \quad & \sum_{i=1}^{k} \sum_{x_j \in S_i} \|x_j - m_i(S_i)\|^2 \\ \text{subject to} \quad & S_1 \cup S_2 \cup \cdots \cup S_k = S, \\ & S_i \cap S_j = \emptyset \text{ if } i \neq j. \end{aligned}$$

In the problem above, $m_i(S_i)$ is the mean of the points in S_i.

The first k initial means are randomly selected from the scenario set. Then, every scenario is associated with the nearest mean. Next, the centroid of each cluster becomes the new mean for that cluster. Finally, when no new centroids are created, the method stops.

The K-Means algorithm [27] is used to construct scenario trees in two-stage stochastic portfolio problem is given in Algorithm 2.

In the K-means model, Euclidean distance is used as a metric, and the number of first-stage clusters k, and second stage clusters l, are input parameters. Therefore, good results from this method depend on the appropriate choice of k and l. Another important issue with K-means algorithm is that bad solutions can be found because of its pure initialization part.

In the Figure 2.5, $k = 3$ clusters is given as an input parameter. First, three random initial means selected (see the top-left part of Figure 2.5). Next, three clusters are constructed by assigning each data point to its nearest mean (see the top-right part of the figure). Then, centroids for each cluster are computed using the Algorithm 2 (see the bottom-left part of the figure). These last two steps are repeated until there is no new reassignments.

Algorithm 2 K-Means

First-Stage

Initialization: Given an initial set of k means $m_1^{(1)}, m_2^{(1)}, \ldots, m_k^{(1)}$

Assignment Step: Assign each observation to the cluster with the closest mean.

$$S_i^{(t)} = \left\{ x_j : \left\| x_j - m_i^{(\tau)} \right\| \leq \left\| x_j - m_{i*}^{(\tau)} \right\|, \ \forall i^* = 1, \ldots, k \right\}$$

Update Step: Calculate the new means to be the centroid of the observations in the cluster.

$$m_i^{(\tau+1)} = \frac{1}{\left| S_i^{(\tau)} \right|} \sum_{x_j \in S_i^{(\tau)}} x_j$$

Stop when the assignments do not change.

Second-Stage

Let e_i be the number of child nodes for each cluster in the first-stage, $i = 1, \ldots, k$

for $i = 1$ to k **do**

Initialization: From e_i observations select an initial set of l_i means $m_1^{(1)}, m_2^{(1)}, \ldots, m_{l_i}^{(1)}$.

Assignment Step: Assign each observation to the cluster with the closest mean.

$$S_i^{(\tau)} = \left\{ x_j : \left\| x_j - m_i^{(\tau)} \right\| \leq \left\| x_j - m_{i*}^{(\tau)} \right\|, \ \forall i^* = 1, \ldots, l_i \right\}$$

Update Step: Calculate the new means to be the centroid of the observations in the cluster.

$$m_i^{(\tau+1)} = \frac{1}{\left| S_i^{(\tau)} \right|} \sum_{x_j \in S_i^{(\tau)}} x_j$$

Calculate the probabilities and conditional of each first-stage and second-stage scenarios, respectively.

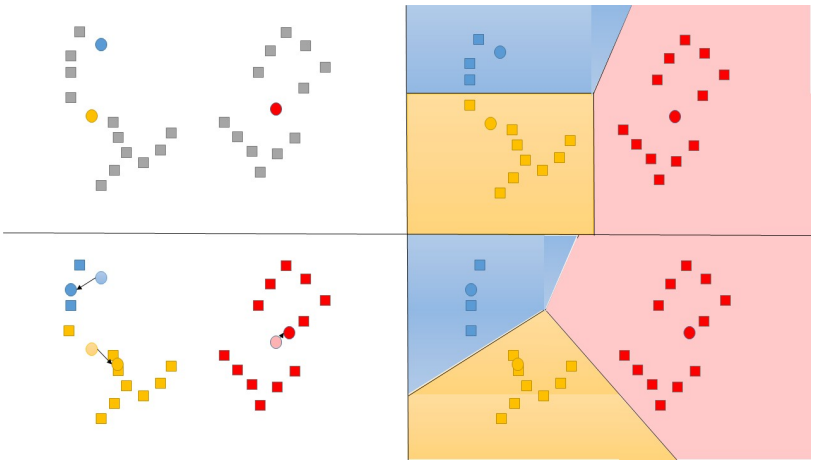

Figure 2.5: K-Means Algorithm

2.3.3 Two-Step Clustering Scenario Tree Generation Method

IBM's [22] two-step clustering scenario tree generation method is designed for very large data sets. The method requires only one pass of the data, and has two major steps. In the first step, scenarios are grouped into many small preclusters. In the second step, these preclusters are clustered into a desired number of clusters by using agglomerative hierarchical clustering. First, we will give a brief explanation of agglomerative hierarchical clustering.

Hierarchical clustering only requires a measure of similarity (i.e. distance) between groups of data points. However, k-means requires a number of clusters, an initial assignment of given data to clusters, and a distance measure.

In agglomerative hierarchical clustering, the idea is to put each data point into its own group in the beginning. Then, merge the two closest groups iteratively until all the data are merged into optimal number of clusters within distance limit or desired number of clusters. Agglomerative hierarchical clustering is explained with the following figures in 2.6 and 2.7.

Suppose we are given data as follows in 2.6 and we want to use agglomerative

hierarchical clustering with Euclidean distance on the given data.

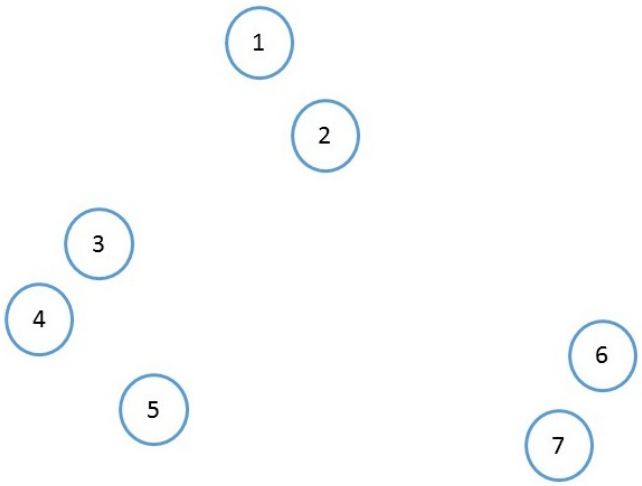

Figure 2.6: Agglomerative Hiearchical Clustering - Data

In the given data above, we have seven data points. First, two closest data points are merged into a cluster according to the Euclidean distance. Based on the distance matrix, suppose we merged data points 1 and 2, 3 and 4, and 6 and 7. We have clusters {1, 2}, {3, 4}, {5}, and {6, 7} after the first merge. Next, we will merge these further based on the constructed distance matrix between clusters. The new clusters will be {1, 2}, {3, 4, 5}, and {6, 7}. This process will continue until we reach to the number of clusters we desire.

The two-step clustering method, which is based on agglomerative hierarchical clustering, is explained in Algorithm 3.

As in the K-means model, Euclidean distance is used as a metric, and the number of clusters at the first stage k, and at the second stage l, are input parameters.

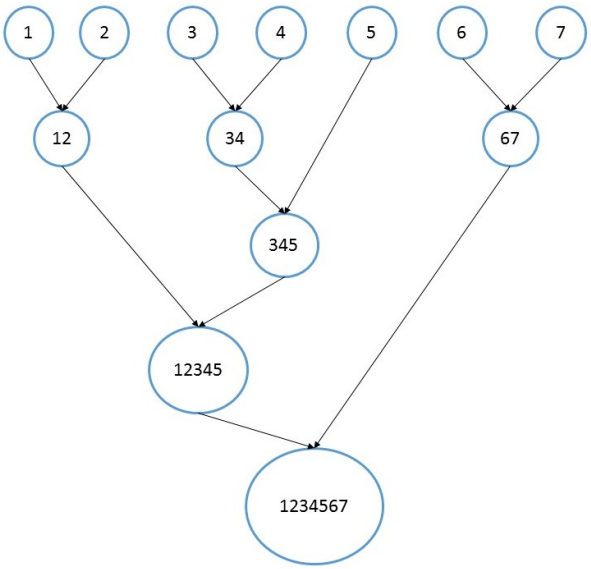

Figure 2.7: Agglomerative Hiearchical Clustering - Dendrogram

2.4 Quality of Scenario Tree Generation Methods

No universally good scenario tree generation method exists. Therefore, we need a measure to evaluate the quality of the trees generated. One possibility is use error values of methods or convergence of the objective function values can be used. Pflug defined an approximation error in [37] as follows. Let the original problem be

$$\min_{x \in X} \quad F(x; \tilde{\xi}),$$

and the tree-based problem be

$$\min_{x \in X} \quad F(x; \tilde{\eta}),$$

with the solution x^*. Then, the approximation error can be calculated as follows:

$$e_f(\tilde{\eta}, \tilde{\xi}) = F(x^*; \tilde{\xi}) - \min_x F(x; \tilde{\xi}) \geq 0.$$

Algorithm 3 Two-Step Clustering Algorithm

First-Stage

Step 1: Formation of Preclusters

Form preclusters by making each scenario a cluster in itself.

For each scenario, find if it should be merged with a previously formed precluster or there should be a new precluster based on a selected distance measure considering the weights (agglomerative hierarchical clustering).

When preclustering is complete, all scenarios in one precluster become a single entity.

Step 2: Clustering of Preclusters

Take the preclusters obtained in Step 1 as an input, and group them into desired number of clusters by using agglomerative hierarchical clustering considering the weights.

Second-Stage

Let e_i be the number of children nodes for each cluster in the first-stage, $i = 1, \ldots, k$

for $i = 1$ to k **do**

Apply Step 1 & 2 explained above to each cluster i with e_i observations.

Calculate the probabilities and conditional probabilities of the first-stage and second-stage scenarios.

However, to calculate this error function is not easy because we need to find the true objective value for a given solution x, and the true optimal solution, which is practically impossible. Therefore, in this study a *mass transportation model* will be formulated to compare the original distribution with the distribution in the scenario tree. The model evaluates the *Monge–Kantorovich metric*.

In the model below, we use the following data:

p_i - the probability of path i in the original distribution

q_j - the probability of path j in the tree

d_{ij} - euclidean distance between path i in the original distribution and path j in the tree based on Formula (2.1)

We look for the cheapest way to move the probability mass from the "original" distribution (the empirical distribution supported on the generated scenarios) and the distribution on the tree. The variables f_{ij} represent the probability mass to be moved

from path i in the original distribution to path j in the tree:

$$\begin{aligned} \min \quad & \sum_i \sum_j d_{ij} f_{ij} \\ \text{subject to} \quad & \sum_j f_{ij} = p_i, \quad \forall\, i, \\ & \sum_i f_{ij} = q_j, \quad \forall\, j, \\ & f_{ij} \geq 0, \quad \forall i, j. \end{aligned} \tag{2.6}$$

The optimal value of this problem will be used as a measure of the quality of the tree. We will compare the scenario tree generation methods based on their optimal values in the Monge-Kantorovich problem above and choose the method which has the lowest optimal value.

The first constraint in problem (2.6) makes sure that total probability mass moved from each path in the original distribution to all paths in the tree is same with the probability of that path in the original distribution. The same idea applies to the second constraint. Total probability mass moved from each path in the tree to all paths in the original distribution should be same with the probability of that path in the tree.

Chapter 3
Two-Stage Portfolio Optimization Problem

3.1 Literature Review

The general one-stage problem, when the risk function is a semideviation of order 1, or weighted mean-deviation from quantile, can be formulated and solved as a linear programming problem (see [28, 48] and also [31] for additional insights). We shall focus on the two-stage version. In what follows, we assume that the sequence of returns (R^1, R^2) has a distribution supported on a scenario tree.

In this chapter, first a general linear two-stage stochastic programming model will be introduced. Next, a two-stage portfolio optimization problem will be formulated based on general linear two-stage stochastic program. Here, mean-semideviation higher-order risk functions will be used. Then, the risk-averse multicut method, which is a decomposition method to solve two-stage portfolio problem, will be introduced. Finally, the mean-semideviation model will be solved with the risk-averse multicut method, and results will be analyzed.

Miller and Ruszczyński stated general linear two-stage stochastic programming model as follows in [31]:

Let $\omega = \{1, ..., N_2\}$ be a finite probability space with a σ-algebra F of all possible subsets of ω with probabilities $P[j] > 0, j = 1, ..., N_2$, and $F_1 \subset F$ by a σ-algebra given by events $\omega_s, s = 1, ..., N_1$, and $p_s = P[\omega_s] > 0, s = 1, ..., N_1$. then, for each event $j \in \omega_s$, the conditional probability $p_{sj} = P[j|\omega_s] = P[j]/P[\omega_s]$.

Let $\mathcal{A}$ be a deterministic matrix of size $m_x \times n_x$, $b \in \Re^{m_x}$ a deterministic vector, $c \in \Re^{n_x}$, $q \in \Re^{n_y}$, and $h \in \Re^{m_y}$ be random vectors, T be a matrix of size $m_y \times n_x$, W be a matrix of size $m_y \times n_y$. While q is only F-measurable, c, h, T, and W are F_1-measurable.

In portfolio problem, return of the portfolio $X = -R^T z$, risk measure $\rho(X)$, and $\mathcal{X} = L_p(\omega, F, P), p \in [1, +\infty)$, and we can define the domain

$$\mathrm{dom}(\rho) := \{X \in \mathcal{X} : \rho(X) < +\infty\}$$

The function ρ is convex and real-valued. Therefore, it is continuous [47]. The conjugate of a risk function $\rho : \mathcal{X} \to \Re$ is defined in [39] as $\rho^* : \mathcal{X}^* \to \Re$

$$\rho^*(\mu) := \sup_{X \in \mathcal{X}} \{\langle \mu, X \rangle - \rho(X)\}$$

In [39], it is proved that ρ^* is proper. So,

$$\rho(X) = \sup_{\mu \in \mathcal{A}} \{\langle \mu, X \rangle - \rho^*(\mu)\}$$

where $\mathcal{A} = \mathrm{dom}(\rho^*)$.

In [32], it is found that if ρ is a coherent risk measure, and is proper and lower semi-continuous, then the representation in (3.1) holds. Also, by using positive homogeneity, $\mathcal{A} = \partial\rho(0)$. Therefore, representation of coherent risk measures can be followed by the theory of convex functions.

$$\rho(X) = \sup_{\mu \in \mathcal{A}} \langle \mu, X \rangle \tag{3.1}$$

Then, by using the representation theorem, we can rewrite the equation (3.1) as follows:

$$\rho(-R^T z) = -\inf_{\mu \in \mathcal{A}} \langle \mu, R^T z \rangle \tag{3.2}$$

Therefore, mean-risk models with semideviation risk functions satisfy (3.2), and we can write the portfolio optimization problem as

$$-\max_{z \in Z} \inf_{\mu \in \mathcal{A}} \langle \mu, R^T z \rangle$$

If ρ is continuous then $\mathcal{A}$ is bounded. We also can change "inf" to "min" because $\mathcal{A}$ is convex, closed and compact. Since Z is compact, "min" and "max" can be interchanged [47]. The dual problem can be formulated as

$$\min_{\mu \in \mathcal{A}} \max_{z \in Z} \langle \mu, R^T z \rangle \tag{3.3}$$

We will use the formulation (3.3) while developing the risk-averse multicut method in the portfolio problem later.

In the two-stage stochastic program, there are two decision vectors. The first-stage decision vector $z \in \Re^{n_z}$ will be selected before any random data are observed. The second-stage decision vector $y \in \Re^{n_y}$ is selected after an event in F_1 is observed. A risk-neutral linear two-stage stochastic program can be formulated as follows:

$$\begin{aligned} \min_{x,Y} \ & \sum_{s=1}^{N_1} p_s[c_s^T z + \sum_{j\in\omega_s} p_{sj} q_{sj}^T y_s] \\ \text{s.t.} \ & Az = b \\ & T_s z + W_s y_s = h_s, \ \ s = 1, ..., N_1 \\ & y_s \geq 0, \ \ s = 1, ..., N_1 \\ & z \geq 0 \end{aligned} \tag{3.4}$$

The risk-neutral problem (3.4) can be written in a nested form by using interchangeability and conditioning (see, [39] and Chapter 2 in [15]).

$$\begin{aligned} \min_{x} \ & \sum_{s=1}^{N_1} \hat{c}^T z + \sum_{s=1}^{N_1} p_s V_s(z) \\ \text{s.t.} \ & Az = b \\ & z \geq 0 \end{aligned} \tag{3.5}$$

where $\hat{c} = \sum_{s=1}^{N_1} p_s c_s$, $V(z)$ is the random variable taking the value $V_s(z)$ with probability p_s. Here, $V_s(z)$ is the optimal value of the second-stage problem which is defined as

$$\begin{aligned} \min_{y} \ & \sum \hat{q_s}^T y \\ \text{s.t.} \ & T_s z + W_s y_s = h_s, \ \ s = 1, ..., N_1 \\ & y_s \geq 0, \ \ s = 1, ..., N_1 \end{aligned} \tag{3.6}$$

where $\hat{q}_s = \sum_{j\in\omega_s} p_{sj} q_{sj}$ and the optimal value of (3.6) is denoted by $V_s(z)$.

Since all data in the second-stage problem is F_1 measurable, it can be solved separately for each s. Next, risk-averse two-stage portfolio optimization problem will

be derived from the risk-neutral linear two-stage stochastic program mentioned above.

$$\begin{aligned} \min\ & \hat{c}^T z + \rho_1(V(z)) \\ \text{s.t.}\ & Az = b \\ & z \geq 0 \end{aligned} \tag{3.7}$$

where $\hat{c} = \sum_{s=1}^{N_1} p_s c_s$, $V(z)$ is the random variable taking the value $V_s(z)$ with probability p_s. The second-stage problem is as follows:

$$\begin{aligned} \min\ & \rho_{2s}(\hat{q_s}^T y) \\ \text{s.t.}\ & T_s z + W_s y_s = h_s, \quad s = 1, ..., N_1 \\ & y_s \geq 0, \quad s = 1, ..., N_1 \end{aligned} \tag{3.8}$$

where $\hat{q}_s = \sum_{j \in \omega_s} p_{sj} q_{sj}$.

3.2 Time Consistency

An investor makes portfolio decisions dynamically at discrete times such as once a day. Therefore, an investor has to consider his or her risk level both at intermediate stages and during the entire time horizon for his or her portfolio allocation decisions. That is why time-consistency of a model is important for an investor who makes decisions over time.

Time-consistency has been studied in various contexts in the past in [25], [26], [5].

In another approach, Boda and Filar [8] define time-consistency by using the "principle of optimality" of dynamic programming. According to their definition, a risk measure is time consistent if a decision-maker uses a risk measure to minimize a multistage policy for the n-stage problem, then a part of that policy at t^{th}-stage (for any $t = 1, .., n$) should also be a risk measure that minimizes the multi-stage

policy for the remaining $(n-t+1)$-stages. This property makes sure that an investor relies on the future to make decisions rather than the past. The relation between time-consistency and the structure of dynamic measures of risk was investigated in [45].

In this study, we adapt the time-consistency approach which is similar to Cheridito et al. [11].

Definition 3.1. Let Y_T and V_T be random variables observed at T^{th}-stage. A dynamic convex risk measure $(\rho_t)_{0\leq s\leq t\leq T}$ is time-consistent if any of the following conditions holds:

1. $\forall s,t$ such that $0 \leq s < t \leq T$

$$\rho_{sT}(Y_T) = \rho_{sT}(\rho_{tT}(Y_T))$$

2. $\forall s,t$ such that $0 \leq s < t \leq T$

$$\rho_{tT}(Y_T) \leq \rho_{tT}(V_T) \Rightarrow \rho_{sT}(Y_T) \leq \rho_{sT}(V_T)$$

Miller and Ruszczyński [31] stated that the two-stage portfolio optimization problem with the mean-semideviation risk measure is time-consistent. Every node in the first stage is associated with a coherent risk measure applied to the children nodes in the second stage. This means if an investor is at node i after the first stage, the information available at node i tells future risk by reducing the outcome space of end portfolio returns.

3.3 The Mean-Semideviation Model

In this section, risk averse two-stage portfolio problem will be formulated with the mean-semideviation risk function of order r.

Definition 3.2. Indices:

$i :=$ First-stage scenarios, $\quad i = 1, ..., I$

$j :=$ Securities, $\quad j = 1, ..., n$

$l :=$ Second-stage scenarios for each first-stage scenario, $\quad l = 1, ..., L(i)$

Definition 3.3. Parameters:

$p_i :=$ probability of a first-stage scenario i

$p_{il} :=$ conditional probability of second-stage scenario l after the first-stage scenario i

$R_{ji} :=$ return rate of security j in first-stage scenario i

$R_{jil} :=$ return rate of security j in second-stage scenario l

$\epsilon_j :=$ relative transaction cost of security j

$\gamma :=$ risk aversion constant

Definition 3.4. Decision Variables:

First-Stage Problem

$z_j :=$ amount invested in security j in the first-stage

$\xi :=$ auxiliary variables representing shortfalls at the first-stage

$u :=$ expectation at the first-stage

Second-Stage Problem

y_{ji} := new position of security j in scenario i after the first-stage

b_{ji} := amount spent to buy security j in scenario i after the first-stage

s_{ji} := value of security j sold in scenario i after the first-stage

σ := auxiliary variables representing shortfalls at the second-stage

m := conditional expectation at the second-stage

We assume that the initial capital is 1, and thus (in the simplest version)

$$z \in Z = \left\{ z \in \mathbb{R}^n : \sum_{j=1}^{n} z_j = 1,\ z_j \geq 0,\ j = 1, \ldots, n \right\};$$

more complex restrictions on the initial investments are possible as well, as long as they define a polyhedral set Z. In order to estimate the (relative) transaction costs, the following bid-ask spread formula is used:

$$\epsilon_j = \frac{(\text{AskPrice}_j - \text{BidPrice}_j)/2}{\text{AskPrice}_j}, \quad j = 1, \ldots, n.$$

This formula assumes that a "fair price" is half-way between the bid and the ask prices, and ignores transaction costs due to the price impact of large trades. We calculate transaction cost for each security by averaging on the data set.

The link between the first-stage variables z and the second-stage variables y is provided by the cash balance equation:

$$y_{ji} = (1 + R_{ji}) z_j + (1 - \epsilon_j) b_{ji} - (1 + \epsilon_j) \cdot s_{ji}, \tag{3.9}$$

in which we symmetrically assign transaction costs to the sales and purchases.

This equation ensures that we included the rebalancing action in the two-stage stochastic program. However, since we will not have the option to buy and sell in the one-stage stochastic program (static model), we will make variables b_{ji} and s_{ji} equal to 0.

The first-stage problem with mean-semideviation risk functions of order $r \geq 1$ can be now formulated as follows:

$$\min_{z \in Z} \quad \sum_{i=1}^{I} p_i V_i(z) + \gamma\Big(\sum_{i=1}^{I} p_i \big(\max(V_i(z) - p^\top V(z), 0) \big)^r \Big)^{1/r},$$

where $V_i(z)$ is the optimal value of ith second stage minimization problem of function (1.17). The first-stage problem can be rewritten as follows:

$$\begin{aligned} \min \quad & u + \gamma\Big(\sum_{i=1}^{I} p_i \xi_i^r \Big)^{1/r} \\ \text{s.t.} \quad & u = \sum_{i=1}^{I} p_i V_i(z), \\ & u \leq V_i(z) + \xi_i, \quad i = 1, \ldots, I, \\ & \xi \geq 0, \\ & z \in Z. \end{aligned} \tag{3.10}$$

The second-stage problem with mean-semideviation risk function of order $r \geq 1$ is formulated in scenario i given in model (3.11).

$$
\begin{aligned}
\min \quad & -m_i + \gamma\Big(\sum_{l=1}^{L(i)} p_{il}\sigma_{il}^r\Big)^{1/r} \\
\text{s.t.} \quad & y_{ji} = (1+R_{ji})z_j + (1-\epsilon_j)b_{ji} - (1+\epsilon)s_{ji}, \quad j = 1,\dots,n, \\
& \sum_{j=1}^{n}\big(b_{ji} - d_{ji}\big) = 0, \\
& W_{il} = \sum_{j=1}^{n}(1+R_{jil})y_{ji}, \quad l = 1,\dots,L(i), \\
& m_i = \sum_{l=1}^{L(i)} p_{il}W_{il}, \\
& m_i \le W_{il} + \sigma_{il}, \quad l = 1,\dots,L(i), \\
& b \ge 0, \quad d \ge 0, \quad \sigma \ge 0.
\end{aligned}
\tag{3.11}
$$

The optimal value of this problem is denoted by $V_i(z)$. In a more general formulation, we may use different risk-aversion parameters γ in the first-stage problem (3.10) and in the second stage problems (3.11), making them dependent on the scenario i. We can also add to problem (3.11) additional restrictions on the allocations y_{ji}, $j = 1,\dots,n$, as long as they define a nonempty polyhedral set. In particular, keeping the first-stage investments unchanged, that is, setting $y_{ji} = (1+R_{ji})z_j$, $j = 1,\dots,n$, should be feasible for problem (3.11).

3.4 Solution Method

3.4.1 Benders' Decomposition

An important issue in large-scale optimization problems is the increasing need for computer memory and computational effort. When the number of variables and constraints increase, it is not possible to solve a large-scale optimization problem exactly to find the decisions. Benders' decomposition [7] is an exact solution algorithm

to mitigate this computational difficulty. The logic behind the Benders' algorithm is to solve a series of smaller problems rather than a large problem so that computational resources are efficiently used.

Benders' decomposition algorithm divides the decision-making process into stages in a large-scale optimization problem. In the first stage, a master problem is formulated and solved with a subset of variables and then the values of the variables in the first-stage problem are used while solving second-stage problems to compute the values of the remaining variables. If the second stage problem finds that the values of first stage decisions are not feasible, then more constraints are added to the master problem. This process continues until the values of the first stage problem are feasible and optimal solution is obtained. Decomposition methods are explained in detail in [50] (see Chapter 3).

In this study, we will use Risk-Averse Multicut Method for Higher-Order Conditional Measures of Risk which is a decomposition algorithm based on Benders' decomposition.

3.4.2 Risk–Averse Multicut Method for Higher-Order Conditional Measures of Risk

If $r = 1$, the problems (3.10) and (3.11) (for $i = 1, \ldots, I$) can be put together into a large-scale linear programming problem, with V_i, $i = 1, \ldots, I$, treated as variables (given by the formula in the first row of (3.11)). The dimension of this problem, however, is of order $I \times (L+n)$ variables and constraints, which becomes unmanageable for realistic sizes of scenario trees. If $r > 1$, an additional complication arises from the fact that we have to deal with a large-scale *nonlinear* optimization problem. In this study, we will develop a general risk-averse multicut method that can solve large-scale nonlinear portfolio problems when higher-order risk measures are used.

We shall, therefore, develop a *decomposition method* for solving problem (3.10)–(3.11), based on Benders decomposition. In order to describe this method, we have to recall the *dual representation* of measures of risk. For a coherent measure of risk $\rho : \mathcal{X} \to \Re$, where $\mathcal{X}$ is the vector space of random variables on a finite probability space having I elementary events, a closed convex set $\mathcal{A}$ of probability measures on this space exists, such that

$$\rho(V) = \max_{\mu \in \mathcal{A}} \sum_{i=1}^{I} \mu_i V_i, \quad V \in \mathcal{X}. \tag{3.12}$$

This representation, first proved in [4], is valid in a much more general setting as well (see, [47, 49] and the references therein). The set $\mathcal{A}$ is the subdifferential of $\rho(\cdot)$ at zero. Analytical expressions for the sets $\mathcal{A}$ for popular measures of risk (including the mean–semideviation measure) are available (see [49]).

Owing to (3.12), the first-stage problem (1.16) becomes

$$\min_{z \in Z} \max_{\mu \in \mathcal{A}} \sum_{i=1}^{I} \mu_i V_i(z), \tag{3.13}$$

Two main issues arise from this formulation. First, solving the "max" problem above is hard by using all of the elements in $\mathcal{A}$, especially, when $r > 1$. Second, there is no easy expression for $V_i(z)$, $i = 1, \ldots, I$, which are optimal values of problems (3.11).

In order to handle the first issue, rather than using $\mathcal{A}$, its approximation from within, $\mathrm{conv}\big(\{\mu^0, \mu^1, ..., \mu^{k-1}\}\big)$ will be used. Here $\mathrm{conv}(C)$ denotes the convex hull of a set C, and $\mu^1, ..., \mu^{k-1}$ are elements of $\mathcal{A}$ collected in iterations $1, \ldots, k-1$ of the method. For μ^0 we substitute the nominal probability distribution p, which is an element of $\mathcal{A}$ for all practically relevant measures of risk, including the mean–semideviation measure.

We construct an approximation of problem (3.13) as

$$\min_{z\in Z} \max_{\kappa=0,1,..,k-1} \sum_{i=1}^{I} \mu_i^{\kappa} V_i(z).$$

It is an approximation from below, because the maximum is evaluated over a subset of $\mathcal{A}$ rather than over $\mathcal{A}$. Equivalently, the problem above can be written as *linear* programming problem:

$$\begin{aligned} \min \quad & \alpha \\ \text{s.t.} \quad & \alpha \geq \sum_{i=1}^{I} \mu_i^{\kappa} V_i(z), \quad \kappa = 0, 1, ..., k-1. \end{aligned}$$

To deal with the second issue, the unknown functions $V_i(z)$ will be replaced with piecewise linear convex functions constructed from *cuts*, derived from the solutions of subproblems (3.11) at earlier iterations. This is a standard way of dealing with parameter-dependent subproblems, similar to expected-value two-stage problems (see, [43] and the references therein). In general, each cut is an inequality

$$V_i(z) \geq \hat{v}_i^{\kappa} + \left(g_i^{\kappa}\right)^{\top}(z - z^{\kappa}),$$

where $\hat{v}_i^{\kappa}$ is an optimal value of problem (3.11) for scenario i in iteration κ of the method, with $z = z^{\kappa}$. These cuts are also called objective cuts since they will help to improve the lower bound while solving the master problem. The subgradient $g_{i\kappa}$ can be calculated from the Lagrange multipliers π_i^{κ} associated with the constraints (3.9) of problem (3.11) involving the parameters z:

$$g_{ij}^{\kappa} = (1 + R_{ji})\pi_{ji}^{\kappa}. \tag{3.14}$$

We can always find a feasible solution to the two-stage portfolio problem. Therefore,

we will not need any feasibility cuts in the risk-averse multicut method.

The reader may consult [31] for details of the cut construction in two-stage risk-averse linear programming, which is identical to our case.

The algorithm for risk-averse multicut method is explained as follows:

Step 0: Set $k = 1$, $i = 1, \ldots, I$.

Step 1: Solve the master problem,

$$\begin{aligned} \min_{z,v,\alpha} \quad & \alpha \\ \text{s.t.} \quad & \alpha \geq \sum_{i=1}^{I} \mu_i^{\kappa} v_i, \quad \kappa = 0, 1, ..., k-1, \\ & v_i \geq \bar{v}_{i\kappa} + g_{\kappa}^{\top}(z - z^{\kappa}), \quad \kappa = 1, ..., k-1, \quad i = 1, ..., I, \\ & z \in Z, \quad v \geq v_{\min}. \end{aligned}$$

Denote the solution by z^k, α^k, v^k.

Step 2: For each $i = 1, \ldots, I$.

Solve the second-stage problem (3.11) and let $\hat{v}_i^k$ be its optimal value and π_{ji}^k the Lagrange multiplier associated with the rebalance constraint for security j in constraint (3.9). Then, calculate g_κ by using the equation (3.14).

Step 3: Calculate $\rho_1^k = \rho_1\left[\hat{v}^k\right]$ and $\mu^k \in \partial\rho_1\left[\hat{v}^k\right]$.

Step 4: If $\rho_1^k = \alpha^k$, then stop; otherwise, continue.

Step 5: Increase k by 1 and go to Step 1.

In the risk-averse multicut method we assume that the set Z is compact, and that we know lower bounds $\hat{v}_{\min}$ for the optimal values of each second stage problem.

From general convergence results in [44], convergence of this method is finite with the mean-semideviation risk function of order 1. If $r \geq 2$, convergence still can be

proved, by the general properties of cutting plane methods for convex programming (see, e.g., [44, Thm 7.7]).

Updating μ function depends on the chosen conditional risk measure. In this study, two different higher-order conditional risk measures will be used: Mean–semideviation and dual risk measures. In the next section, calculations how to derive μ function will be shown for both cases.

Risk-averse multicut method has two main differences compared to Benders' decomposition. First, separate objective cuts are generated for each first-stage scenario rather than generating an aggregate objective cut. Second, the method memorizes all measures μ^κ from the previous iterations to construct a more accurate lower bound for ρ_1. This reduces the number of iterations needed to find the optimal solution.

3.4.3 Calculation of μ

The calculation of μ in Step 3 depends on the risk measure applied. In this section, we will show how to calculate μ for mean-semideviation risk function of any order r and also dual higher-order risk measures.

We will first start with mean-semi deviation model. Let r be the order of the mean-semideviation model, the following process describes how to update μ function.

$$\rho_1^r(v) = \sum_i p_i v_i + \gamma \left[\cdot \sum_i p_i \cdot (\max(0, \hat{v}_i^k - \sum_{m=1}^{I} p_m \hat{v}_i^k))^r \right]^{1/r}$$

We calculate the shortfall values as follows:

$$\sigma_i = \max\left(0, \hat{v}_i^k - \sum_{m=1}^{I} p_m \hat{v}_i^k\right), \quad i = 1, \ldots, I,$$

Then, we can rewrite the semideviation risk function of order r as follows:

$$\varphi = \Big(\sum_{i=1}^{I} p_i (\sigma_i)^r \Big)^{1/r}.$$

Since $\mu \in A \Leftrightarrow \mu \in \partial\rho(0)$ holds, we can derive two consequences as a result of this relation:

1. $A = \partial\rho(0)$

2. ρ is positively homogeneous.

$\partial\rho(\hat{v}) \subseteq \partial\rho(0)$. For this method, a typical situation is to observe $\hat{v} \neq 0$. In this case, $\rho(.)$ is actually differentiable at $\hat{v} \neq 0$

Then, the derivative of φ with respect to σ_k will be

$$\frac{\partial \varphi}{\partial \sigma_k} = \frac{1 \cdot r \cdot (\sigma_k^+)^{r-1} \cdot p_k}{r \cdot (\sum\limits_i p_i \cdot (\sigma_i^+)^r)^{1-(1/r)}}$$

By taking the derivative of ρ with respect to $\hat{v}_i$

$$\begin{aligned}
\frac{\partial \rho}{\partial \hat{v}_i} &= p_i + \gamma \cdot \sum_k \frac{\partial \varphi}{\partial \sigma_k} \cdot \frac{\partial \sigma_k}{\partial \hat{v}_i} \\
&= p_i + \gamma \cdot \frac{\partial \varphi}{\partial \sigma_i} \cdot \frac{\partial \sigma_i}{\partial \hat{v}_i} + \gamma \cdot \sum_{k \neq i} \frac{\partial \varphi}{\partial \sigma_k} \cdot \frac{\partial \sigma_k}{\partial \hat{v}_i} \\
&= p_i + \gamma \cdot \frac{(\sigma_i)^{r-1} \cdot p_i}{(\varphi)^{r-1}} \cdot (1 - p_i) + \gamma \cdot \sum_{k \neq i} \frac{(\sigma_k^+)^{r-1} \cdot p_k}{\varphi} \cdot (-p_i)
\end{aligned}$$

Finally, we will obtain the general formula to update μ as follows:

$$\mu_i^k = p_i - \frac{\gamma \cdot p_i}{(\varphi)^{r-1}} \cdot \sum_{m=1}^{I} (\sigma_m^+)^{r-1} \cdot p_m + \gamma \cdot \frac{(\sigma_i^+)^{r-1} \cdot p_i}{(\varphi)^{r-1}}$$
$$= p_i \cdot \left[1 + \frac{\gamma}{(\varphi)^{r-1}} \cdot ((\sigma_i^+)^{r-1} - \sum_{m=1}^{k} (\sigma_m^+)^{r-1} \cdot p_m) \right] \tag{3.15}$$

In Step 3 of the risk-averse multicut method, μ will be updated based on formula (3.15). Next, we need to check two following properties for μ.

$$\sum_i \mu_i = 1$$
$$\mu_i \geq 0$$

We will start with first required condition.

$$\sum_i \mu_i = \sum_i p_i + \frac{\gamma}{(\varphi)^{r-1}} \cdot \left[\sum_s (\sigma_s^+)^{r-1} \cdot p_i - \sum_i \sum_k (\sigma_k^+)^{r-1} \cdot p_k \cdot p_i \right]$$
(Since, $\sum_i p_i = 1$)
$$\sum_i \mu_i = \sum_i p_i + \frac{\gamma}{(\varphi)^{r-1}} \cdot \left[\sum_i p_i \cdot (\sigma_i^+)^{r-1} - \sum_k p_k \cdot (\sigma_k^+)^{r-1} \right]$$
(Since, $\sum_i p_i \cdot \sigma_i^+ - \sum_k p_k \cdot \sigma_k^+ = 0$)
$$\sum_i \mu_i = \sum_i p_i = 1$$

Therefore, the condition $\sum_i \mu_i = 1$ is satisfied. Second,

$$\frac{\gamma}{(\varphi)^{r-1}} \cdot ((\sigma_i^+)^{r-1} - \sum_k p_k \cdot (\sigma_k^+)^{r-1}) \overset{?}{\geq} -1$$

$$\gamma \cdot ((\sigma_i^+)^{r-1} - \sum_k p_k \cdot (\sigma_k^+)^{r-1}) \overset{?}{\geq} -(\sum_k p_k \cdot (\sigma_k^+)^r)^{(r-1)/r}$$

$$-\gamma \cdot ((\sigma_i^+)^{r-1} - \sum_k p_k \cdot (\sigma_k^+)^{r-1}) \overset{?}{\leq} (\sum_k p_k \cdot (\sigma_k^+)^r)^{(r-1)/r}$$

$$\sum_k p_k \cdot (\sigma_k^+)^{r-1} - \min_i (\sigma_i^+)^{r-1} \overset{?}{\leq} (\sum_k p_k \cdot (\sigma_k^+)^r)^{(r-1)/r}$$

is sufficient to say

$$\sum_k p_k \cdot (\sigma_k^+)^{r-1} \overset{?}{\leq} (\sum_k p_k \cdot (\sigma_k^+)^r)^{(r-1)/r}$$

From Hölder inequality

$$E(\sigma^+)^{r-1} \leq E((\sigma^+)^r)^{(r-1)/r}$$

Therefore, the condition $\mu_i \geq 0$ is satisfied.

Next, we will show how to update μ function for dual higher-order risk measures. Even though, the formulation of the two-stage portfolio problem with dual higher-order risk measure is not given in this study, the risk-averse multicut method can be used to solve the two-stage portfolio problem with dual higher-order risk measures. Here, we will show how to update the μ function in Step 3 of the risk-averse multicut method when second order dual risk function is used. As a future work, we want to compare the results of two-stage portfolio problem with higher-order mean-semideviation risk function with dual order risk functions.

$$\rho^2(z) = \min_\eta \left\{ c(\sum_{i=1}^n p_i (z_j - \eta)_+^2)^{1/2} + \eta \right\} \text{where} \quad c \geq 1$$

The function $\rho^2(z)$ satisfies the convexity and monotonicity properties of coherent risk measures in [49], [15]. We need to check positive homogeneity and translation

equivariance properties of coherent risk measures.

$$\rho^2(z) = \min_{\eta} \left\{ c \left\| (Z-\eta)_+ \right\|_p + \eta \right\}$$

$$\rho^2(\alpha z) = \min_{\eta} \left\{ c \left\| (\alpha Z-\eta)_+ \right\|_p + \eta \right\}$$

$$\alpha > 0 \text{ and, let } \alpha\tau = \eta \Rightarrow \tau = (\eta/\alpha) \in \Re$$

$$= \min_{\tau} \left\{ c \left\| (\alpha Z-\alpha\eta)_+ \right\|_p + \alpha\tau \right\}$$

$$= \alpha \min_{\tau} \left\{ c \left\| (Z-\tau)_+ \right\|_p + \tau \right\} = \alpha\rho^2(z)$$

Therefore, $\rho^2(z)$ satisfies the positive homogeneity property.

$$\rho^2(z+a) = \min_{\eta} \left\{ c \left\| (Z+a-\eta)_+ \right\|_p + \eta \right\}$$

$$\text{Let } \eta - a = \tau$$

$$= \min_{\tau} \left\{ c \left\| (Z-\tau)_+ \right\|_p + \tau + a \right\} = a + \rho^2(z)$$

Therefore, $\rho^2(z)$ satisfies translation equivariance property. Since ρ^2 satisfies all of the conditions, ρ^2 is a coherent risk measure. In this case, $\rho^2(z)$ is actually differentiable.

Let

$$\sigma_i = z_i - \eta, \text{ and}$$

$$\varphi = (\sum_i p_i \cdot (\sigma_i^+)^2)^{1/2}$$

Then, derivative of φ with respect to σ_k will be,

$$\frac{\partial \varphi}{\partial \sigma_k} = \frac{1 \cdot 2 \cdot \sigma_k^+ \cdot p_k}{2 \cdot (\sum_i p_i \cdot (\sigma_i^+)^2)^{1/2}} = \frac{\sigma_k^+ \cdot p_k}{\varphi}$$

By taking the derivative of ρ with respect to z_i

$$\begin{aligned}
\frac{\partial \rho}{\partial z_i} &= c \cdot \sum_k \frac{\partial \varphi}{\partial \sigma_k} \cdot \frac{\partial \sigma_k}{\partial z_i} \\
&= c \cdot \frac{\partial \varphi}{\partial \sigma_i} \cdot \frac{\partial \sigma_i}{\partial z_i} + c \cdot \sum_{k \neq i} \frac{\partial \varphi}{\partial \sigma_k} \cdot \frac{\partial \sigma_k}{\partial z_i} \\
&= c \cdot \frac{\sigma_i^+ \cdot p_i}{\varphi} + c \cdot \sum_{k \neq i} \frac{\sigma_k^+ \cdot p_k}{\varphi} \cdot (0)
\end{aligned}$$

Finally,

$$\mu_i = c \cdot \frac{\sigma_i^+ \cdot p_i}{\varphi} \tag{3.16}$$

Since $\rho^2(\cdot)$ is coherent, these two following properties are satisfied.

$$\sum_i \mu_i = 1$$

$$\mu_i \geq 0$$

Therefore, we will only show the basics how these two properties are satisfied. We will start with the first requirement.

$$\sum_i \mu_i = c \cdot \frac{\sum_i p_i \cdot (\sigma_i^+)}{\varphi}$$

(Since $c \geq 1$ and $\varphi \geq \sum_i p_i \cdot (\sigma_i^+)$

$$\sum_i \mu_i = 1$$

Therefore, the condition $\sum_i \mu_i = 1$ is satisfied.

Second,

(Since $c \geq 1$ and $\varphi \geq \sum_i p_i \cdot (\sigma_i^+)$

$$\sum_i \mu_i \geq 0$$

Therefore, the condition $\mu_i \geq 0$ is satisfied.

Chapter 4
Computational Results

First, daily returns of Dow Jones companies' from September 2, 2008 to November 30, 2011 were used to calibrate a multivariate GO-GARCH(1,1) model of the returns. The model was used to generate a large number of scenarios for the next two days. Returns generated for the first day are used in the first-stage and generated second day values in the second-stage. The generated scenarios were used to construct two-stage scenario trees by employing all five scenario tree generation methods discussed in Chapter 2. These methods were compared in terms of their time and solution quality. The results in Table 4.1 show that two-step clustering methods are slightly faster in terms of CPU times compared to other methods.

Next, we want to find which scenario tree generation method is better in terms of Monge-Kantorovich metric. After scenario trees of the same size are constructed by using the each method in Chapter 2, we will solve the Monge-Kantorovich problem on each tree so that we can compare it with the original probability distribution. Because of the computational complexity of solving large-scale Monge-Kantorovich problems exactly, we will compare five scenario tree generation methods in small size data sets. In the following results, number of the original scenarios are restricted to 10,000 and 20,000 scenarios. Table 4.2 contains the comparison of the quality of scenario trees obtained by different methods: The optimal value of the Monge–Kantorovich metric given by (2.6). Here, a scenario tree generation method is called more accurate if it has lower optimal value in Monge-Kantorovich problem. The results show that tree quality is stable for each scenario tree generation technique because the difference

of optimal values for different scenario tree size is small. However, the multi-facility location clustering method is the most accurate with scenario trees constructed from both 10,000 and 20,000 scenarios. Therefore, in the next part of the study, we used the multi-facility clustering method to generate scenario trees in a rolling horizon fashion.

In the next part of the study, a simulation analysis was carried out. Each day, the preceding 619 days of data were used to calibrate a multivariate GO-GARCH(1,1) model. The model was then used to generate 80,000 scenarios for the following two days in R. Since we found that multi-facility location clustering method is the most accurate method in our previous computations, we will only use the multi-facility location clustering method to construct a scenario tree for the next two days during the simulation. Therefore, we do not need to solve Monge-Kantorovich problem during the simulation (step 4 in Figure 4.1). However, there is not any scenario tree generation method superior to other ones, we would need to solve Monge-Kantorovich problem to find which tree is the most accurate in each simulation day. Then the tree model with conditional measures of risk was solved, the investments were re-balanced, and the method continued. On the next day, new return data were available, new scenario trees were generated, new models solved, etc. The steps of the simulation study are depicted in Figure 4.1.

Table 4.1: CPU Time (in secs) of Scenario Tree Generation Methods

Original Scenario Size	Tree Scenario Size	K-Means (forward)	K-Means (backward)	Two-Step (forward)	Two-Step (backward)	Multi-facility Location
80,000	5,000	675.2	682.1	628.2	621.4	850.7
80,000	10,000	1210.8	1272.6	1152.4	1145.3	1675.4

Table 4.2: Accuracy of Scenario Tree Generation Methods

Original Scenario Size	Tree Scenario Size	K-Means (forward)	K-Means (backward)	Two-Step (forward)	Two-Step (backward)	Multi-facility Location
10,000	500	0.00719	0.00735	0.00859	0.00875	0.00705
20,000	500	0.00718	0.00733	0.00855	0.00869	0.00699

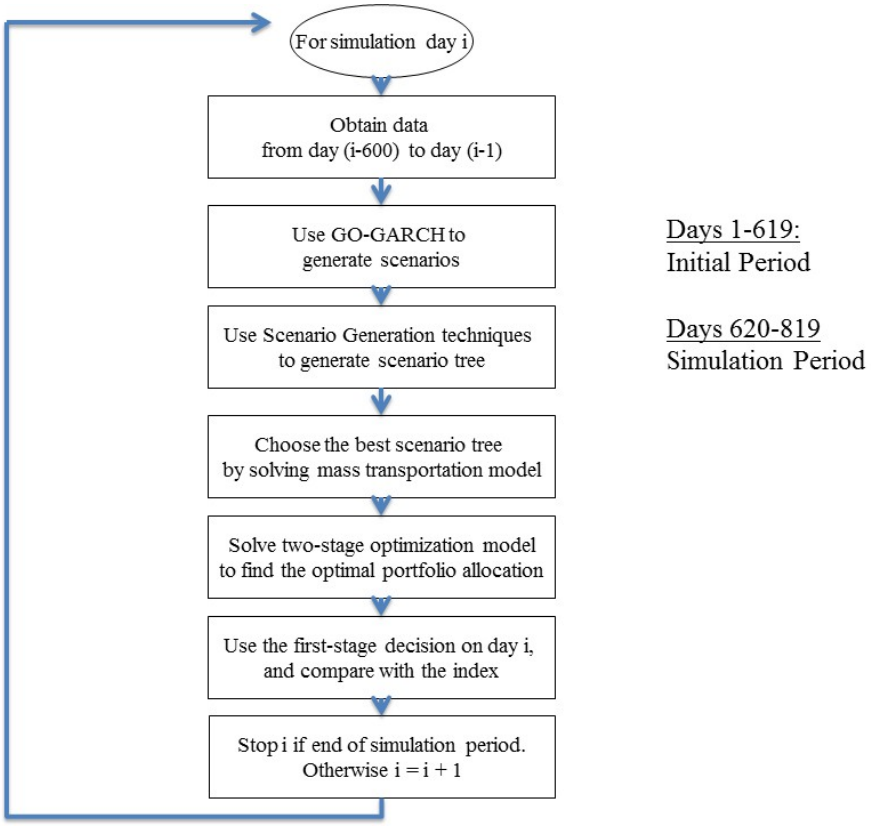

Figure 4.1: Simulation Analysis

The simulation study had two objectives. First, we compared the two-stage portfolio model with the static model where $b_{ji} = d_{ji} = 0$. Based on the cumulative wealth graphs in Figures 4.2 and 4.3, we can say that two-stage portfolio model performs better than the static model for both mean–semideviation risk functions of order 1 and 2. This was partly due to the reduced volume of trades, which resulted in significantly lower transaction costs, but also to a better portfolio composition.

Secondly, we compared two-stage portfolio models with the mean–semideviation risk measures of order 1 and 2 with static minimum variance model based on Markowitz's mean-variance portfolio theory where $b_{ji} = d_{ji} = 0$. In each case we used fixed $\gamma = 0.9$ and bid-ask spread transaction costs. The two-stage portfolio optimization problem was solved with the risk-averse multi-cut method, implemented in MATLAB with the

CPLEX solver. As we can see from Figures 4.3 and 4.4, using the second order methods leads to significant improvements in cumulative wealth trajectories compared to first order and Markowitz's minimum variance models. This is consistent with the findings of [30], where other higher order risk measures were employed (with a static model and no transaction costs).

Finally, we can compare the performance of the portfolios with mean–semideviation of order 1 and 2 with the Dow Jones Index. Based on the following graph, the simulation analysis shows that portfolio with mean–semideviation of order 2 performs better than other portfolios.

Next, we compare β values of the portfolios with mean–semideviation of order 2 when risk aversion constant is either $\gamma = 0.9$ or $\gamma = 0.5$. β is a measure of volatility of a portfolio compared to the market. It is calculated by using the regression analysis. If β is greater than 1, portfolio is more volatile than the market and if it is less than 1, portfolio is less volatile than the market. We computed the β values for both risk aversion constant using the linear regression. β value of portfolio (0.92) with $\gamma = 0.5$ is slightly more than β value of portfolio (0.85) with $\gamma = 0.9$. This means both portfolios generally move in the same direction as the benchmark (Dow Jones). However, movement of the mean-semideviation portfolio of order 2 with $\gamma = 0.9$ is generally less than the movement of the mean–semideviation portfolio of order 2 with $\gamma = 0.5$.

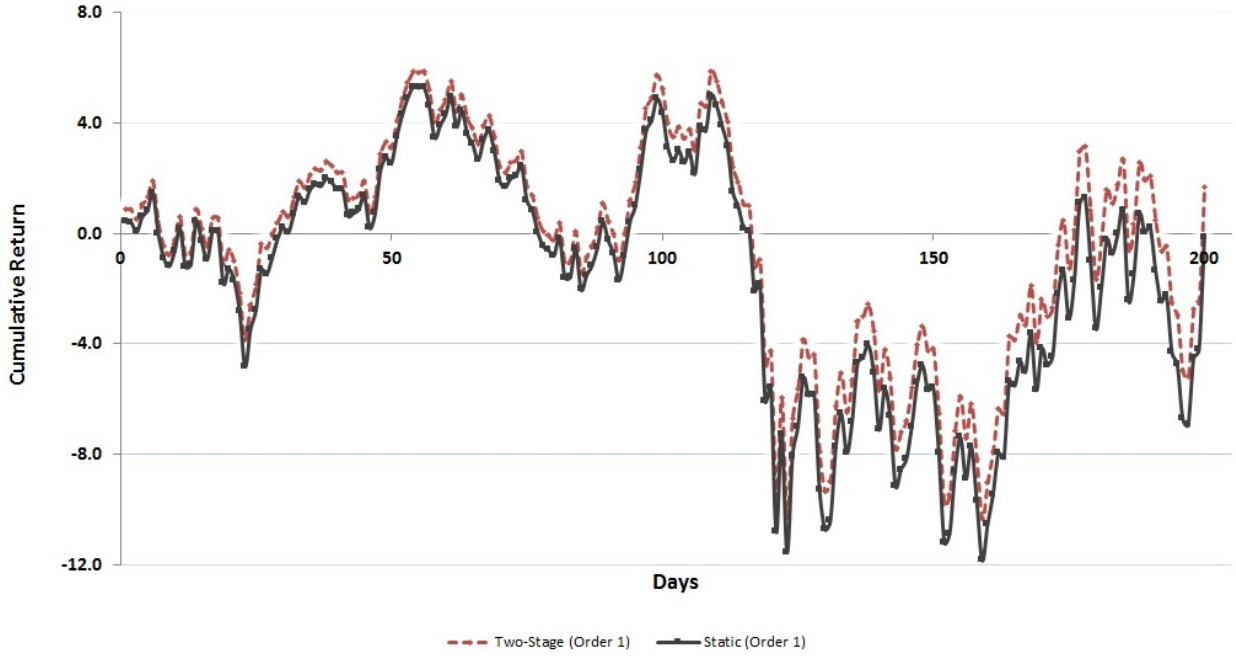

Figure 4.2: Performance of the Static and Two-Stage Portfolios with Mean-Semideviation (Order 1)

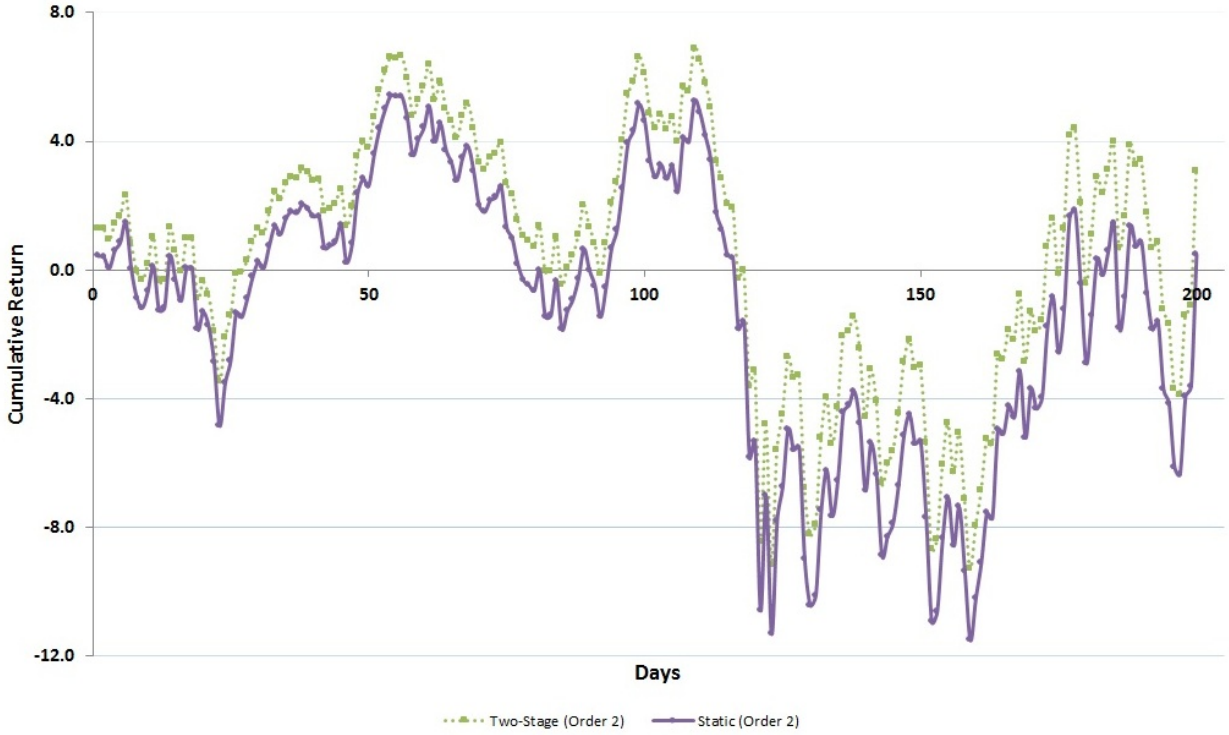

Figure 4.3: Performance of the Static and Two-Stage Portfolios with Mean-Semideviation (Order 2)

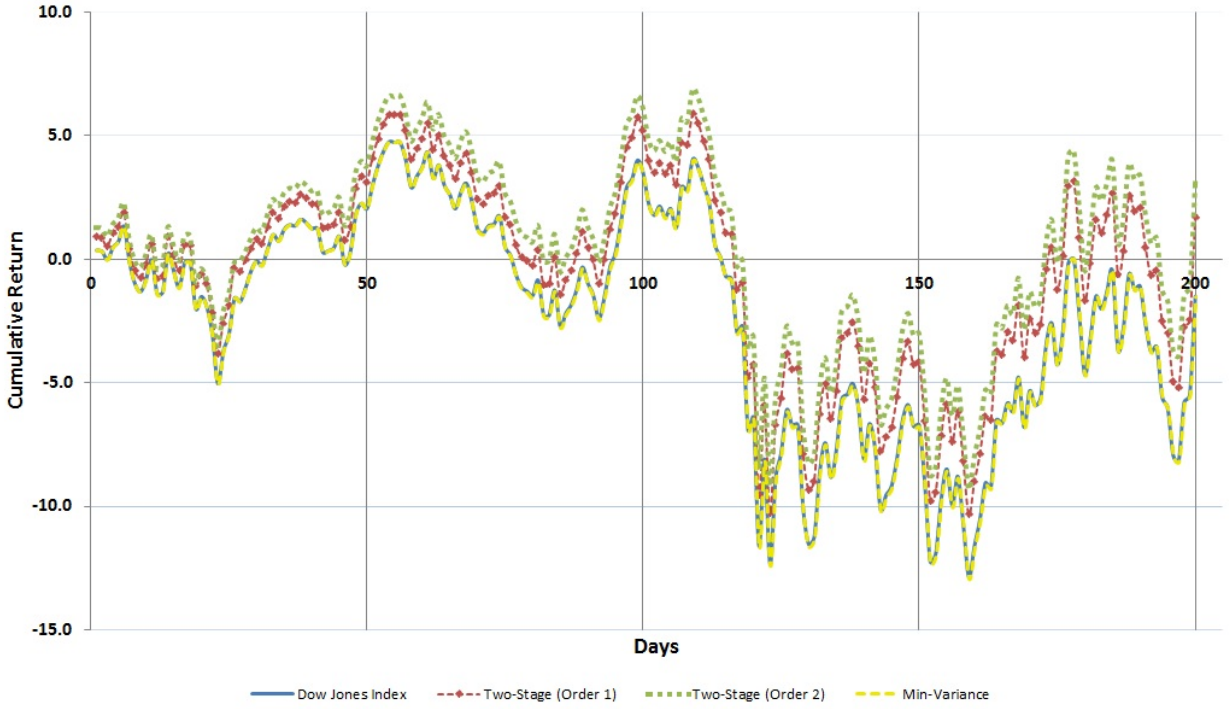

Figure 4.4: Performance of the Two-Stage Portfolios and Dow Jones Index

Chapter 5
Conclusion

In this study, two-stage portfolio models with higher-order conditional risk measures are studied. First, a multi-variate GARCH model is used to generate adequate number of scenarios to model the probabilistic information on random data. Next, scenario trees are constructed by different methods, and the best one is chosen based on Monge-Kantorovich metric between the probability distribution on the scenario tree and the empirical distribution on the raw scenarios. It is found that the two-step forward clustering method is most efficient in terms of the CPU time, because it passes over the data just once. However, that multi-facility location clustering method is the most accurate, in terms of the Monge-Kantorovich metric. Therefore, we used multi-facility location clustering method to generate scenario trees.

Next, conditional mean-semideviation risk functions of order 1 and order 2 are used to formulate the risk-averse two-stage portfolio problem on the trees generated from the multi-facility location clustering method. The problems are solved by a generic risk averse multicut algorithm for any higher-order risk function. The results show that the portfolio allocations for mean-semideviation models of order 1 and order 2 are similar. However, portfolio with the mean-semideviation of order 2 performs better compared to mean-semideviation of order 1, minimum-variance model, and the Dow Jones index. Both two-stage models outperform the static model.

For future research, it would be interesting to examine the question of making two portfolio allocation decisions; one after the stock market opens in the morning and one before stock market closes in the afternoon. In this case, we can consider the

moves in other stock markets after U.S. stock market closes.

Another future research topic can be to use of dual higher-order risk measures to compare with higher-order mean-semideviation risk measures in the two-stage portfolio problem. We explained how we can solve the two-stage portfolio problem with dual higher-order risk measures in Chapter 4.

References

[1] C. Acerbi and D. Tasche. Expected Shortfall: A natural coherent alternative to Value-At-Risk. *Economic Notes*, 31:379–388, 2003.

[2] C. O. Alexander and A. M. Chibumba. Multivariate orthogonal factor GARCH. *University of Sussex Discussion Papers in Mathematics*, 1997.

[3] D. Arthur and S. Vassilvitskii. k-means++: the advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics, 2007.

[4] P. Artzner, F. Delbaen, J. M. Eber, and D. Heath. Coherent Measures of Risk. *Mathematical Finance*, 9:203–228, 1999.

[5] P. Artzner, F. Delbaen, J. M. Eber, D. Heath, and H. Ku. Coherent multiperiod risk adjusted values and Bellman principle. *Annals of Operations Research*, 152:5–22, 2007.

[6] P. Artzner, F. Delbaen, Eber J. M., D. Heath, and H. Ku. Coherent multiperiod risk management. *International Journal of Business Continuity and Risk Management*, 1:125–135, 2010.

[7] J. F. Benders. Partioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4:238–252, 1962.

[8] K. Boda and J. A. Filar. Time Consistent Dynamic Risk Measures. *Mathematical Methods of Operations Research*, 63:169–186, 2006.

[9] H. W. Brachinger and M. Weber. Risk as a primitive: a survey of measures of perceived risk. *Operations Research-Spektrum*, 19:235–250, 1997.

[10] S. Cheng, Y. Liu, and S. Wang. Progress in Risk Management. *Advanced Modelling and Optimization*, 6(1):125–135, 2004.

[11] P. Cheridito, F. Delbaen, and M. Kupper. Dynamic monetary risk measures for bounded discrete-time processes. *Electron. J. Probab.*, 11:57–106, 2006.

[12] L. Cooper. Heuristic methods for location-allocation problems. *SIAM Review*, 6:37–53, 1964.

[13] E. Cuvelier and M. Noirhomme-Fraiture. Clayton copula and mixture decomposition. *ASMDA*, pages 699–708, 2005.

[14] A. Damodaran. Value at Risk (VaR): A big picture perspective, June 2014. http://people.stern.nyu.edu/adamodar/.

[15] D. Dentcheva, S. Penev, and A. Ruszczyński. Kusuoka representation of higher order dual risk measures. *Annals of Operations Research*, 181:325–335, 2010.

[16] R. F. Engle, V. K. Ng, and M. Rothschild. Asset pricing with a factor ARCH covariance structure: empirical estimates for treasury bills. *Journal of Econometrics*, 45:213–238, 1990.

[17] R. Frey, McNeil A. J., and Nyfeler M. Copulas and credit models. *Risk*, 14(10):111–114, 2001.

[18] G. Hanoch and H. Levy. The Efficiency Analysis of Choices Involving Risk. *Review of Economic Studies*, 36:335–346, 1969.

[19] H. Heitsch and W. Romisch. Scenario Tree Modeling for Multistage Stochastic Programs. *Math. Programming*, pages 371–406, 2009.

[20] H. Heitsch, W. Romisch, and C. Strugarek. Stability of multistage stochastic programs. *SIAM Journal on Optimization*, 17:511–525, 2006.

[21] R. Hochreiter and G. Ch. Pflug. Financial scenario generation for stochastic multi-stage decision processes as facility location problems. *Ann. Oper. Research*, 152:257–272, 2007.

[22] SPSS Inc. Cluster Analysis. *Technical Report*, 2004.

[23] A. Inoue. On the worst conditional expectation. *Journal of Mathematical Analysis and Applications*, 286(1):237–247, 2003.

[24] M. Kijima and M. Ohnishi. Mean-risk analysis of risk aversion and wealth effects on optimal portfolios with multiple investment opportunities. *Ann. Oper. Res.*, 45:147–163, 1993.

[25] T. C. Koopmans. Stationary ordinal utility and impatience. *Econometrica*, 28:287–309, 1960.

[26] M. K. Kreps and E. L. Porteus. Temporal resolution of uncertainty and dynamic choice theory. *Econometrica*, 46:185–200, 1978.

[27] S. LLoyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28:129–137, 1982.

[28] R. Mansini, W. Ogryczak, and M. G. Speranza. LP solvable models for portfolio optimization: a classification and computational comparison. *IMA J. Manag. Math.*, 14(3):187–220 (2004), 2003.

[29] H. M. Markowitz. Portfolio Selection. *Journal of Finance*, 7:77–91, 1952.

[30] Y. Matmoura and S. Penev. Multistage optimization of option portfolio using higher order coherent risk measures. *European Journal of Operational Research*, 227(1):190 – 198, 2013.

[31] N. Miller and A. Ruszczyński. Risk-Averse Two-Stage Stochastic Linear Programming: Modeling and Decomposition. *Operations Research*, 59:125–132, 2011.

[32] Naomi L. Miller. *Mean-Risk Portfolio Optimization Problems with Risk-Adjusted Measures*. PhD thesis, Rutgers University, 2008.

[33] S. Mitra and T. Ji. Risk measures in quantitative finance. *International Journal of Business Continuity and Risk Management*, 1(2):125–135, 2010.

[34] W. Ogryczak and A. Ruszczyński. From Stochastic Dominance to mean-risk models: Semideviations as risk measures. *European Journal of Operational Research*, pages 33–50, 1999.

[35] W. Ogryczak and A. Ruszczyński. On Consistency of Stochastic Dominance and Mean-Semideviation Models. *Mathematical Programming*, 89:217–232, 2001.

[36] W. Ogryczak and A. Ruszczyński. Dual Stochastic Dominance and Related Mean-Risk Models. *SIAM J. on Optimization*, 13:60–78, 2002.

[37] G. Ch. Pflug. Scenario tree generation for multiperiod financial optimization by optimal discretization. *Math. Programming*, 89:251–271, 2001.

[38] F. Riedel. Dynamic Coherent Risk Measures. *Stochastic Processes Appl.*, 112:185–200, 2004.

[39] R. T. Rockafellar. Conjugate Duality and Optimization. *SIAM*, 1987.

[40] R. T. Rockafellar and S. Uryasev. Optimization of conditional value at risk. *The Journal of Risk*, 2:21–41, 2000.

[41] R. T. Rockafellar and S. Uryasev. Conditional value at risk for general loss distributions. *Journal of Banking and Finance*, 26:1443–1471, 2002.

[42] M. Rothschild and J. E. Stiglitz. Increasing Risk: A Definition. *Journal of Economic Theory*, 2:225–243, 1970.

[43] A. Ruszczyński. *Decomposition methods, in:* Stochastic Programming*, A. Ruszczyński and A. Shapiro (eds.)*. Elsevier, 2003.

[44] A. Ruszczyński. *Nonlinear Optimization*. Princeton University Press, 2006.

[45] A. Ruszczyński. Risk-averse dynamic programming for Markov decision processes. *Mathematical Programming*, 125(2):235–261, 2010.

[46] A. Ruszczyński and A. Shapiro. Conditional Risk Mappings. *Mathematics of Operations Research*, 31:544–561, 2006.

[47] A. Ruszczyński and A. Shapiro. Optimization of convex risk functions. *Mathematics of Operations Research*, 31:433–452, 2006.

[48] A. Ruszczyński and R. Vanderbei. Frontiers of stochastically nondominated portfolios. *Econometrica*, 71:1287–1297, 2003.

[49] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory*. SIAM, 2009.

[50] A. Shapiro and A. Ruszczyński. *Handbooks in Operations Research and Management Science, 10: Stochastic Programming*. Elsevier, 2003.

[51] A. Silvennoinen and T.. Terasvirta. Multivariate GARCH Models. *Handbook of Financial Time Series*, pages 201–229, 2009.

[52] The Basel Committee on Banking Supervision. The New Basel Capital Accord, Basel Committee on Banking Supervision, Consulative Document. *BIS*, January 2001.

[53] Roy Van Der Weide. GO-GARCH: A Multivariate Generalized Orthogonal GARCH Model. *Journal of Applied Econometrics*, 17:549–564, 2002.

[54] E. Weiszfeld. Sur le point par lequel la somme des distances de n points donnés est minimum. *Tohoku Mathematical Journal*, 43:355–386, 1937.

[55] G. A. Mc. (Eds.) Whitmore. Stochastic Dominance: An Approach to Decision-Making under risk. *Heath*, 1978.

Curriculum Vitae

Sıtkı Gülten

1983	Born in Ankara, Turkey
2001-2005	B. Sc. in Business Administration at Hacettepe University, Turkey
2005-2008	M. Sc. in Industrial Engineering at Bilkent University, Turkey
2008-2014	Ph. D. in Management Science at Rutgers University, NJ, U.S.A

Articles

2014	Two-Stage Portfolio Optimization with Higher-Order Conditional Measures of Risk, revised and resubmitted.
2008	Channel Assignment and Routing for Multi-Radio Wireless Mesh Networks, M. Sc. Thesis

Honors

2013,2014	Conference Scholarships in SIAM Conference on Optimization 2014, and ICSP 2013
2012-2013	Dean's Distinguished Instructorship Award
2012	INFORMS New Jersey Chapter Student Contest Finalist
2009,2012	Summer Research Assistant, Rutgers Business School
2008-2012	Teaching Assistantship, Rutgers Business School
2006-2008	TUBITAK (The Scientific and Technological Research Council of Turkey), National Scholarship Programme for M.Sc. Students