# A CVaR Robust Portfolio Optimization Model with Transaction Costs

by

Tong Li

An essay
presented to the University of Waterloo
in fulfillment for the degree of
Master of Mathematics
in
Computational Mathematics

Supervisor: Prof. Thomas F. Coleman

Waterloo, Ontario, Canada, 2020

I hereby declare that I am the sole author of this essay. This is a true copy of the essay, including any required final revisions, as accepted by my examiners.

I understand that my essay may be made electronically available to the public.

# Abstract

In this essay, a CVaR robust portfolio optimization model with transaction costs is introduced to manage equity portfolios in a real market. We review the traditional mean-variance portfolio optimization model and the CVaR robust portfolio optimization model without transaction cost first. A twice continuously differentiable smoothing technique based on an exponential approximation is proposed to help solve the CVaR optimization problem. Then transaction cost functions are introduced into the CVaR portfolio optimization model. Because most of transaction cost functions are piecewise-linear with several non-differentiable kink points, smoothing techniques can be applied to transaction cost functions as well. Detailed smoothing approaches are illustrated by two typical cost function examples. Finally, we conduct numerical experiments to show the effect of the CVaR robust portfolio optimization model with transaction costs.

# Acknowledgements

I would like to express my sincere gratitude to my supervisor, Professor Thomas F. Coleman for his patient guidance and enthusiastic encouragement throughout these two years. My grateful thanks are also extended to my second reader Prof. Stephen A. Vavasis for his valuable suggestions and assistance.

I would also like to thank all my friends and staff in Computational Mathematics program for their support during my master period.

Finally, I wish to thank my friends in Waterloo city for their help in my daily life.

# Table of Contents

# List of Figures

# Chapter 1

# Introduction

The trade-off between return and risk is always considered as a major issue when investors set up their portfolios. Optimal portfolios should have the highest expected return with a given level of risk or the lowest risk with a given expected return. To determine such optimal portfolios, the traditional mean-variance (MV) portfolio model was proposed by Markowitz (1952).

Consider a portfolio optimization problem: invest wealth $W$ in $n$ assets $S_1,...,S_n$, $n \geq 2$ in a given period. Let $x=(x_1,\ldots,x_n)\in\Re^n$ denote a portfolio with $e^T x = W$. Let random return vector $r$ denote the return at the end of this period for a unit portfolio $(1,\ldots,1)$ and $\bar{r}$ denote the expected values of $r$. Obviously, the random return of this portfolio is $r^T x$ and the expected return is $\bar{r}^T x$. Assume the sample observation is a sample return matrix $R=(r_1,...,r_m)$ where $m$ is the number of observations of random return vector $r$. Let $H$ denote the covariance matrix of $R$. There are several estimations of the expected return $\bar{r}^T x$ based on the observed sample data $R$. The most straightforward method used in the original MV portfolio optimization model is estimating $\bar{r}$ by taking the mean of the sample set of returns:

$$\bar{r} = \frac{\sum_{i=1}^{m} r_i}{m} \tag{1.1}$$

Then the MV portfolio optimization problem can be expressed as a Quadratic Programming (QP) problem:

$$\begin{aligned} &\min_{x} -\bar{r}^T x + t x^T H x \\ &\text{subject to} \quad e^T x = W \end{aligned} \tag{1.2}$$

where $t \geq 0$ is a parameter presupposed by investors to represent their tolerance to risk. Note that we allow the components of $x$ to be negative, which means shorting of assets (i.e. negative investments) is permitted. Moreover, there is another common formulation, the mean-standard deviation (MSD) formulation, with standard deviation type of risk as below:

$$\begin{aligned} &\min_{x} -\bar{r}^T x + t\sqrt{x^T H x} \\ &\text{subject to} \quad e^T x = W \end{aligned} \tag{1.3}$$

We use the MSD formulation in the following because the standard deviation type of risk has the same unit (dollar) with expected returns in the objective function.

However, the risk tolerance parameter $t$ is difficult to be determined in practice when we do portfolio rebalancing rather than constructing the efficient frontier. We consider another equivalent formulation of the MSD portfolio optimization model without parameter $t$. There are several alternative formulations of the optimization problem (1.3). The most common choices are the risk minimization formulation and the expected return maximization formulation. These formulations are equivalent because the optimal solutions of these formulations are in the same efficient frontier. In this essay, we use the risk minimization formulation which achieves a target expected return $\tau$ :

$$
\begin{aligned}
&\min_{x} \ \sqrt{x^T H x} \\
&\text{subject to} \ \ \bar{r}^T x = \tau \\
&\qquad\qquad\quad e^T x = W
\end{aligned}
\tag{1.4}
$$

The optimal portfolio with a target expected return and the minimum of risk can be found by solving this optimization problem. Note that this optimization problem is convex (See Appendix A) with a nonlinear differentiable objective function and linear constraints.

Moreover, the estimation of expected returns (1.1) is highly sensitive to the observed data. A popular method to protect against such sensitivity is the Conditional Value-at-Risk (CVaR) robust portfolio optimization method proposed by Rockafellar et al. (1999). The expected return of a portfolio is replaced by the average of several worst return scenarios with a specified confidence level $\beta$ (e.g., 95%). This CVaR robust portfolio optimization model is described in Chapter 2.

To solve the CVaR optimization problem, an approach is transforming the CVaR function into a piecewise-linear function and then solving it by adding auxiliary variables (auxiliary method). Unfortunately, this method tends to be inefficient when the simulation number is large because the number of variables depends on the number of simulated scenarios. There is another method, the smoothing method, proposed by Alexander et al. (2006) to avoid large number of variables. It is suggested to smooth the CVaR function with a quadratic approximation function. However, the second order derivative of this quadratic approximation function is discontinuous, which leads to discontinuous gradient of constraints in the risk minimized MSD portfolio optimization model. To solve this problem, we propose an exponential smoothing method with continuous second order derivatives. These two smoothing methods are described in Chapter 3.

Moreover, transaction costs are always involved when buying and selling assets in a real market. Generally, transaction cost functions are piecewise-linear and there are a set of kink points where cost functions are non-differentiable. Therefore, quadratic and exponential smoothing techniques can also be applied in the approximation of transaction cost functions. Details are illustrated with V-shape and Butterfly-shape cost function examples. Figure 1.1 and Figure 1.2 give examples of V-shape and Butterfly-shape cost functions respectively. Then we formulate the CVaR robust portfolio optimization problem with smoothed transaction cost functions in Chapter 4. In Chapter 5, we conduct numerical experiments to show the effect of the CVaR portfolio optimization model with transaction costs and conclude this essay in Chapter 6.

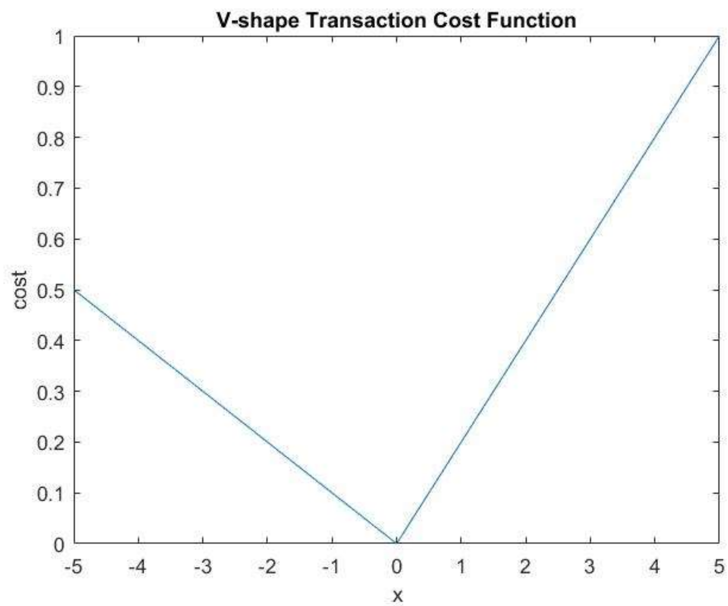


**FIGURE 1.1:** V-shape transaction cost function example

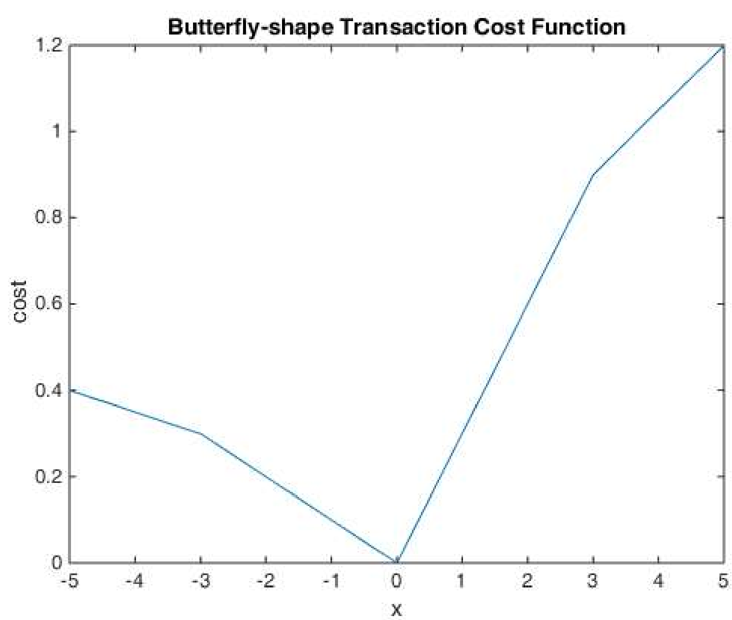


**FIGURE 1.2:** Butterfly-shape transaction cost function example

# Chapter 2

# CVaR Robust Portfolio Optimization

The estimation of expected returns using one sample set is highly sensitive to the sample observations $R$. Inspired by sampling techniques, min-max robust portfolio optimization was introduced by Goldfarb et al. (2003) to reduce the dependence on a specific data set. They considered finite uncertainty sets of resampled returns (scenarios) to achieve a better performance in the worst scenario. However, focusing on the worst scenario makes the min-max robust optimization still sensitive to observations, especially for some extreme points. To reduce such sensitivity, Rockafellar et al. (1999) proposed a CVaR estimation of the expected return $\bar{r}^T x$ with a specified confidence level $\beta$ (e.g., 95%) :

$$C_\beta^r(-r^T x) = \min_\alpha (\alpha + (1-\beta)^{-1} E[-r^T x - \alpha]^+) \tag{2.1}$$

where $E$ is the expectation operator and $[z]^+ = \max(z, 0)$.

Then the expected return $\bar{r}^T x$ in the original MSD model (1.3) can be replaced by $C_\beta^r(-r^T x)$, which produces the MSD type of CVaR robust portfolio optimization problem:

$$\begin{aligned} &\min_{x,\alpha} \alpha + (1-\beta)^{-1} E[-r^T x - \alpha]^+ + t\sqrt{x^T H x} \\ &\text{subject to} \quad e^T x = W \end{aligned} \tag{2.2}$$

To estimate the expectation operator $E$ in $C_\beta^r(-r^T x)$, resampling a set of return scenarios is required.

Assume the return vector $r \in \Re^n$ has an underlying joint normal distribution. $M$ resampled return matrices $R_j = (r_1^S, ..., r_m^S),\ j = 1:M$ can be generated with corresponding mean vectors $\overline{r}_j^S,\ j = 1:M$. The following is a bootstrap resampling process:

1. Factor $H = LL^T$
2. For $j = 1:M$
   a. For $i = 1:m$

      choose $\varepsilon_i \in \Re^n$ from a standard joint normal distribution $N(0,1)$

      Set $r_i^S = \overline{r} + L\varepsilon_i$

   b. $R_j = (r_1^S, ..., r_m^S)$

      $\overline{r}_j^S = \dfrac{\sum_{i=1}^{m} r_i^S}{m}$

Note that $E(r_i^S) = \overline{r}$ and $(r_i^S - \overline{r})^T H^{-1} (r_i^S - \overline{r}) = \varepsilon_i^T \varepsilon_i \approx n$.

Then $E[-r^T x - \alpha]^+$ can be estimated by:

$$E[-r^T x - \alpha]^+ = \frac{\sum_{j=1}^{M} [-\overline{r}_j^S x - \alpha]^+}{M} \tag{2.3}$$

(2.2) and (2.3) specify the CVaR robust mean-standard deviation portfolio optimization problem as:

$$\min_{x,\alpha} \alpha + \frac{1}{M(1-\beta)} \sum_{j=1}^{M} [-\overline{r}_j^S x - \alpha]^+ + t\sqrt{x^T H x}$$
$$\text{subject to} \quad e^T x = W \tag{2.4}$$

The corresponding equivalent risk minimization formulation which achieves a target expected return $\tau$ is:

$$\min_{x,\alpha} \sqrt{x^T H x}$$
$$\text{subject to} \quad \alpha + \frac{1}{M(1-\beta)} \sum_{j=1}^{M} [-\overline{r}_j^S x - \alpha]^+ = \tau \tag{2.5}$$
$$e^T x = W$$

To deal with $[-\overline{r}_j^S x - \alpha]^+$, a well-known method is introducing auxiliary variables $z_j$ to replace $[-\overline{r}_j^S x - \alpha]^+$ with additional constraints $z_j \geq 0$ and $z_j \geq -\overline{r}_j^S x - \alpha$. Considering apply this auxiliary method in (2.4):

$$\min_{x,\alpha,z} \alpha + \frac{1}{M(1-\beta)} \sum_{j=1}^{M} z_j + t\sqrt{x^T H x}$$
$$\text{subject to} \quad z_j \geq -\overline{r}_j^S x - \alpha, \quad j = 1, \ldots, M \tag{2.6}$$
$$z_j \geq 0, \qquad j = 1, \ldots, M$$
$$e^T x = W$$

This optimization problem contains $O(M+n)$ variables and $O(M)$ constraints where $M$ is the number of simulations and $n$ is the number of assets. Usually the simulation number $M$ is large to get a better estimation, which makes this auxiliary method inefficient. The same issue arises when applying this method to the equivalent risk minimization formulation (2.5). To avoid this inefficient issue, we introduce the smoothing method in the next chapter.

# Chapter 3

# Smoothing Techniques

### 3.1 Quadratic Smoothing

Following the approach suggested by Alexander et al., the smoothing method can also be used to approximate $[-\bar{r}_j^S x - \alpha]^+$. It is introduced with a quadratic approximation form:

$$[z]^+ \cong \rho_\varepsilon(z) = \begin{cases} z & z \geq \varepsilon \\ \frac{z^2}{4\varepsilon} + \frac{z}{2} + \frac{\varepsilon}{4} & \text{-}\varepsilon \leq z \leq \varepsilon \\ 0 & \text{otherwise} \end{cases} \tag{3.1}$$

Note that $\rho_\varepsilon(z)$ is continuously differentiable. Specifically,

$$\rho_\varepsilon^{'}(z) = \begin{cases} 1 & z \geq \varepsilon \\ \frac{z}{2\varepsilon} + \frac{1}{2} & \text{-}\varepsilon \leq z \leq \varepsilon \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

However, the second order derivatives of $\rho_\varepsilon(z)$ is discontinuous with jump points at $z = -\varepsilon$ and $z = +\varepsilon$:

$$\rho_\varepsilon^{''}(z) = \begin{cases} \frac{1}{2\varepsilon} & \text{-}\varepsilon \leq z \leq \varepsilon \\ 0 & \text{otherwise} \end{cases} \tag{3.3}$$

Denote $\rho_\varepsilon^S(\alpha, x) = \frac{1}{M(1-\beta)} \sum_{i=1}^{M} \rho_\varepsilon(-(\bar{r}_i^S)^T x - \alpha))$. After quadratic smoothing, the CVaR robust expected return becomes:

$$
\begin{aligned}
C_{\beta}^{r}(-r^{T}x) &\simeq \min_{\alpha}(\alpha+\tfrac{1}{M(1-\beta)}\sum_{i=1}^{M}\rho_{\varepsilon}(-(\overline{r}_{i}^{S})^{T}x-\alpha)) \\
&= \min_{\alpha}(\alpha+\rho_{\varepsilon}^{S}(\alpha,x)) \\
&= \alpha_{*}+\rho_{\varepsilon}^{S}(\alpha_{*},x)
\end{aligned} \tag{3.4}
$$

where $\alpha_*$ is the optimal solution of $\min_{\alpha}(\alpha+\rho_{\varepsilon}^{S}(\alpha,x))$. Therefore, after taking derivative with respect to $\alpha$, the optimal solution $\alpha_*$ satisfies:

$$
1+\rho_{\varepsilon}^{S}(\alpha_{*},x)_{\alpha}=0 \tag{3.5}
$$

Then the risk minimization formulation (2.5) becomes:

$$
\begin{aligned}
&\min_{\alpha,x}\sqrt{x^{T}Hx} \\
&\text{subject to } \ 1+\rho_{\varepsilon}^{S}(\alpha\ ,x)_{\alpha}=0 \\
&\qquad\qquad\ \ \alpha+\rho_{\varepsilon}^{S}(\alpha,x)=\tau \\
&\qquad\qquad\ \ e^{T}x=W
\end{aligned} \tag{3.6}
$$

Compared with the auxiliary method, this smoothing method is much more efficient with only $O(n)$ variables and 3 constraints. However, the derivatives of the first constraint in (3.6) with respect to $\alpha$ contains the discontinuous second order derivatives of $\rho_{\varepsilon}(z)$, which makes many optimization algorithms perform poorly.

## 3.2 Exponential Smoothing

To avoid the discontinuity problem above, we propose a new exponential smoothing method which is twice continuously differentiable:

$$
[z]^{+}\cong\rho_{e}(z)=\begin{cases}\alpha_{1}e^{\alpha_{2}z} & z<0\\ z+\alpha_{1}e^{(\frac{\alpha_{1}\alpha_{2}-1}{\alpha_{1}})z} & z\geq 0\end{cases} \tag{3.7}
$$

where $\alpha_1>0,\ \alpha_2>0$ are approximation parameters to control the accuracy of approximation with $0<\alpha_1\cdot\alpha_2<1$. Note that $\rho_e(z)$ is continuous with $\rho_e(0)=\alpha_1$. In addition, $\rho_e(z)\to 0$ as $z\to-\infty$ and $(\rho_e(z)-z)\to 0$ as $z\to+\infty$. This exponential smooth function also has continuous first order derivative:

$$\rho_e'(z) = \begin{cases} \alpha_1\alpha_2 e^{\alpha_2 z} & z < 0 \\ 1 + (\alpha_1\alpha_2 - 1) e^{(\frac{\alpha_1\alpha_2 - 1}{\alpha_1})z} & z \geq 0 \end{cases} \tag{3.8}$$

with $\rho_e'(0) = \alpha_1\alpha_2$.

Consider the second order derivative of $\rho_e(z)$:

$$\rho_e''(z) = \begin{cases} \alpha_1\alpha_2^2 e^{\alpha_2 z} & z < 0 \\ \dfrac{(\alpha_1\alpha_2 - 1)^2}{\alpha_1} e^{(\frac{\alpha_1\alpha_2 - 1}{\alpha_1})z} & z \geq 0 \end{cases} \tag{3.9}$$

Define $t_1(z) \triangleq \alpha_1\alpha_2^2 e^{\alpha_2 z}$ and $t_2(z) \triangleq \dfrac{(\alpha_1\alpha_2 - 1)^2}{\alpha_1} e^{(\frac{\alpha_1\alpha_2 - 1}{\alpha_1})z}$. Then $t_1(0)=\alpha_1\alpha_2^2$ and $t_2(0) = \dfrac{(\alpha_1\alpha_2 - 1)^2}{\alpha_1}$ are equal when $\alpha_1\alpha_2 = 0.5$, which yields a continuous second order derivative for $\rho_e(z)$.

Therefore, the exponential smoothing formulation of CVaR robust mean-standard deviation portfolio model can be formulated by replacing $\rho_\varepsilon(\alpha, x)$ in (3.6) with $\rho_e(\alpha, x)$:

$$\begin{aligned} & \min_{\alpha,x} \sqrt{x^T H x} \\ \text{subject to} \quad & 1 + \rho_e^S(\alpha \ , x)_\alpha = 0 \\ & \alpha + \rho_e^S(\alpha, x) = \tau \\ & e^T x = W \end{aligned} \tag{3.10}$$

where $\rho_e^S(\alpha, x) = \frac{1}{M(1-\beta)} \sum_{i=1}^{M} \rho_e(-(\bar{r}_i^S)^T x - \alpha))$.

Then off-the-shelf optimization algorithms can be used to solve (3.10) because the convex objective function and all equality constraint functions are continuously differentiable.

# Chapter 4

# Optimization with Transaction Costs

Portfolio rebalancing requires buying and selling which incur transaction costs. Involving transaction costs in portfolio optimization problems is necessary for investors. Typically, transaction costs can be classified into two types: fixed costs and variable costs. Fixed costs are unrelated to the trading volume and it can be modeled easily in optimization problems. Hence, we only discuss variable costs in this essay. Usually larger transaction costs are associated with larger amounts of trading (buying and selling), which determines a general piecewise-linear characteristic of transaction cost functions with several non-differentiable kink points.

However, when transaction cost functions are involved in portfolio optimization problems, the non-differentiability problem hampers optimization algorithms. To solve this problem, we apply both quadratic smoothing and exponential smoothing techniques into transaction cost functions. The most common transaction cost function is the V-shape function (shown as Figure 1.1) which means costs are proportional to the absolute values of trading amounts with one kink point at zero. Apart from this, transaction costs in real market can be more complicated than V-shape functions. For instance, cost functions may have more kink points reflecting lower cost rates because discounts can be awarded for high-volume trading. We consider a Butterfly-shape cost function (shown as Figure 1.2) with two more kink points than the V-shape function in this essay. Smoothing methods for cost functions are illustrated with these two typical V-shape and Butterfly-shape transaction cost functions in the following. Moreover, this method can be adapted to other types of transaction cost functions.

## 4.1 V-shape Transaction Cost Function

### 4.1.1 Quadratic smoothing of V-shape transaction cost function

Let $t_V(x)$ denote a V-shape cost function with the slope $m_1^+$ to the right of zero and the slope $-m_1^-$ to the left of zero where $m_1^+ > 0,\ m_1^- > 0$. Consider the amount of trading $x$ where $x > 0$ corresponds to buying and $x < 0$ corresponds to selling.

$$t_V(x) = \begin{cases} -m_1^- x & x \le 0 \\ m_1^+ x & x > 0 \end{cases} \tag{4.1}$$

Because this function is non-differentiable at the kink point $x=0$, smoothing techniques need to be applied here. The quadratic smoothing of this cost function on the range of $[-\varepsilon, \varepsilon]$ with $\varepsilon > 0$ around the kink point at $x=0$ is:

$$q_\varepsilon(x) \triangleq \left(\frac{m_1^+ + m_1^-}{4\varepsilon}\right)x^2 + \left(\frac{m_1^+ - m_1^-}{2}\right)x + \left(\frac{m_1^+ + m_1^-}{4}\right)\varepsilon \tag{4.2}$$

Then the V-shape transaction cost function after quadratic smoothing becomes:

$$\overline{t}_V^{\,q}(x) = \begin{cases} -m_1^- x & x < -\varepsilon \\ q_\varepsilon(x) & -\varepsilon \le x \le \varepsilon \\ m_1^+ x & x > \varepsilon \end{cases} \tag{4.3}$$

Note that $\overline{t}_V^{\,q}(x)$ is continuously differentiable. Specifically,

$$(\overline{t}_V^{\,q})'(x) = \begin{cases} -m_1^- & x < -\varepsilon \\ \left(\frac{m_1^+ + m_1^-}{2\varepsilon}\right)x + \left(\frac{m_1^+ - m_1^-}{2}\right) & -\varepsilon \le x \le \varepsilon \\ m_1^+ & x > \varepsilon \end{cases} \tag{4.4}$$

Considering an example setting of $m_1^+ = 0.3,\ m_1^- = 0.1,\ \varepsilon=2$, we illustrate the quadratic smoothed V-shape cost defined by (4.2) and (4.3) in Figure 4.1 and the derivative function (4.4) of this approximation in Figure 4.2.

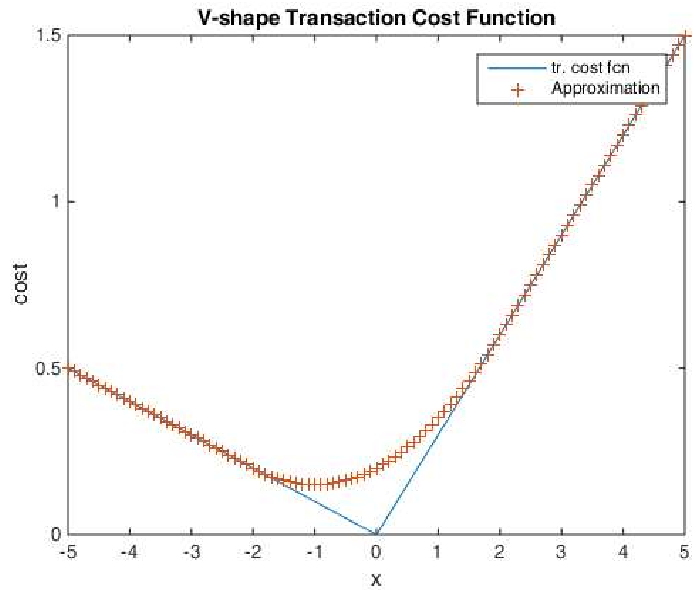


**FIGURE 4.1:** Quadratically smoothed V-shape cost

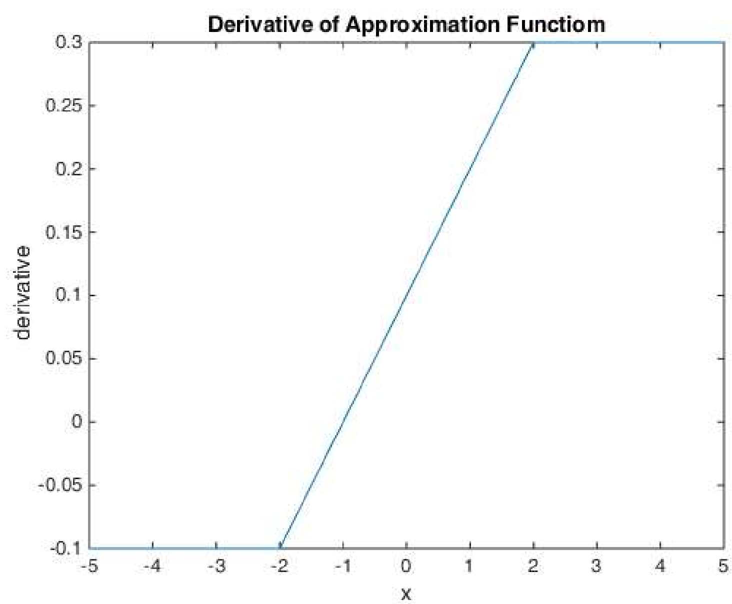


**FIGURE 4.2:** 1st derivative of quadratically smoothed V-shape cost

Note that the second order derivative of $\overline{t}_V{}^q(x)$ is discontinuous:

$$(\overline{t}_V{}^q)''(x) = \begin{cases} \frac{m_1^+ + m_1^-}{2\varepsilon} & -\varepsilon \le x \le \varepsilon \\ 0 & \text{otherwise} \end{cases} \qquad (4.5)$$

### 4.1.2 Exponential smoothing of V-shape transaction cost function

As an alternative approximation method, the exponential smoothing can be applied to the V-shape transaction cost function:

$$\overline{t}_V^{\,e}(x) = \begin{cases} f_1(x) \triangleq m_1^+ \cdot x + \alpha_1 e^{-\frac{1}{\alpha_1} m_1^+ x} & x > 0 \\ f_2(x) \triangleq -m_1^- \cdot x + \alpha_1 e^{\frac{1}{\alpha_1} m_1^- x} & x \le 0 \end{cases} \tag{4.6}$$

where $\alpha_1 > 0$. Note that $\overline{t}_V^{\,e}(x) - t_V(x) \to 0^+$ as $|x| \to \infty$.

Differentiating $f_1(x)$ and $f_2(x)$ we get:

$$f_1'(x) = m_1^+ - m_1^+ e^{-\frac{1}{\alpha_1} m_1^+ x} \tag{4.7}$$

and

$$f_2'(x) \triangleq -m_1^- + m_1^- e^{\frac{1}{\alpha_1} m_1^- x} \tag{4.8}$$

Clearly, $\overline{t}_V^{\,e}(x)$ is a continuously differentiable function with $f_1(0) = \alpha_1 = f_2(0)$ and $f_1'(0) = 0 = f_2'(0)$. After taking the derivative again, we have:

$$f_1''(x) = \tfrac{1}{\alpha_1} (m_1^+)^2 e^{-\frac{1}{\alpha_1} m_1^+ x} \tag{4.9}$$

and

$$f_2''(x) = \tfrac{1}{\alpha_1} (m_1^-)^2 e^{\frac{1}{\alpha_1} m_1^- x} \tag{4.10}$$

with $f_1''(0) = \frac{1}{\alpha_1}(m_1^+)^2$ and $f_2''(0) = \frac{1}{\alpha_1}(m_1^-)^2$. Therefore, $\overline{t}_V^{\,e}(x)$ is twice continuously differentiable if $m_1^+ = m_1^-$ with $\overline{t}_r''(0) = \frac{(m_1^+)^2}{\alpha_1} = \frac{(m_1^-)^2}{\alpha_1}$.

Consider an example with $m_1^+ = m_1^- = 0.25$ and $\alpha_1 = 0.5$. Figures 4.3, 4.4 and 4.5 illustrate the exponential approximation, the corresponding first order derivative and the second order derivative function.

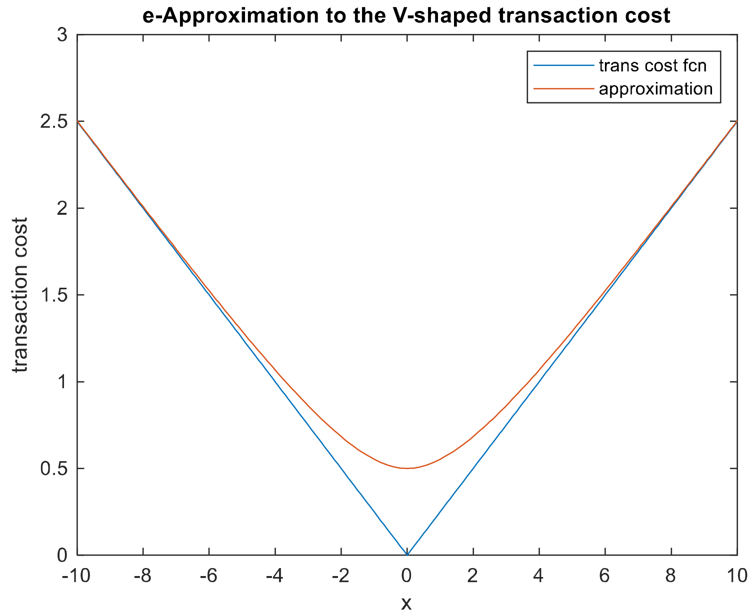


**FIGURE 4.3:** Exponentially smoothed V-shape cost (same slopes)

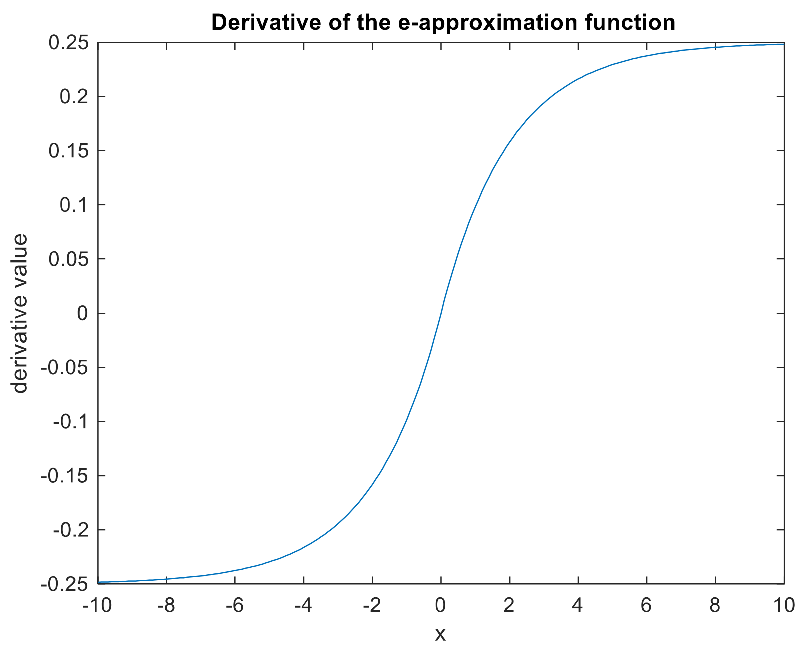


**FIGURE 4.4:** 1st derivative of exponentially smoothed V-shape cost (same slopes)

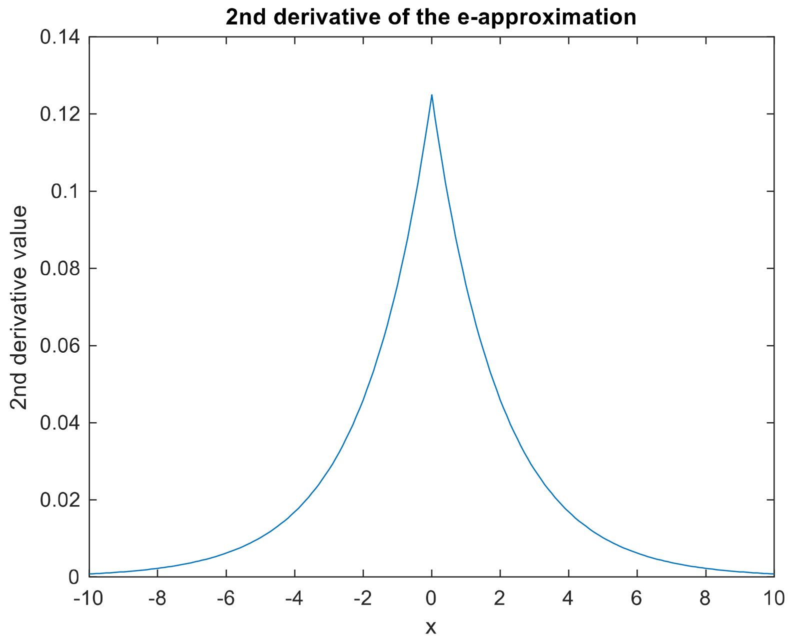


**FIGURE 4.5:** 2nd derivative of exponentially smoothed V-shape cost (same slopes)

Then we can take an example with different slopes. Figures 4.6, 4.7 and 4.8 illustrate the exponential approximation and the corresponding derivative and 2nd-derivative functions when $m_1^+ = 0.2,\ m_1^- = 0.3,\ \alpha_1 = 0.5$.

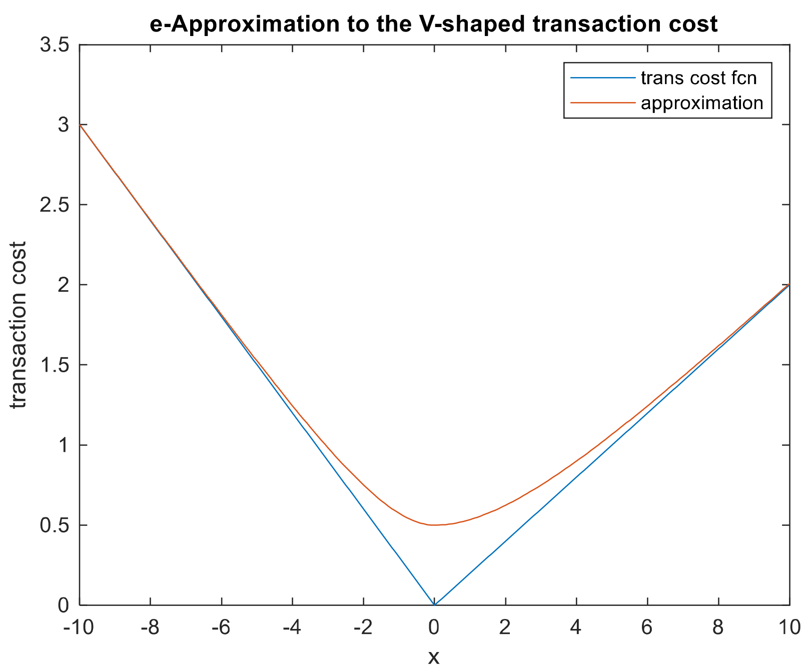


**FIGURE 4.6:** Exponentially smoothed V-shape cost (different slopes)

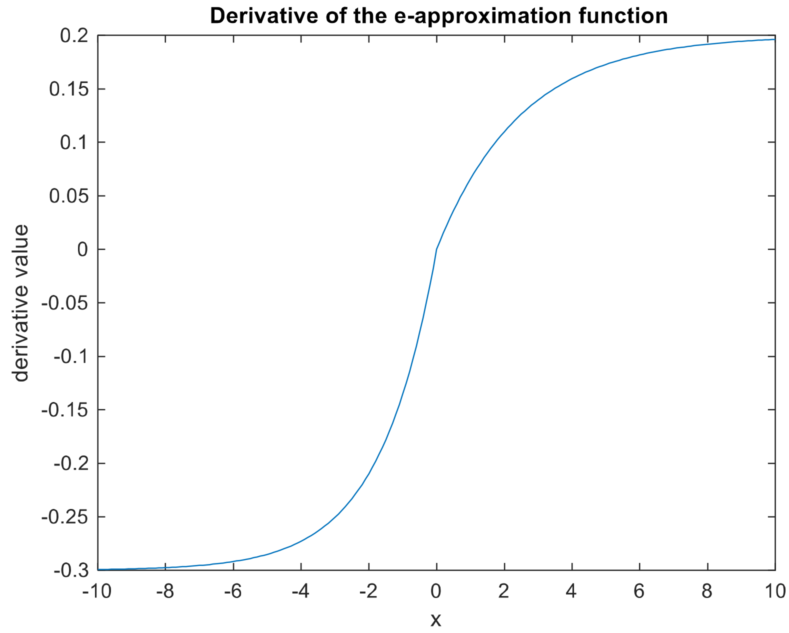


**FIGURE 4.7:** 1st derivative of exponentially smoothed V-shape cost (different slopes)

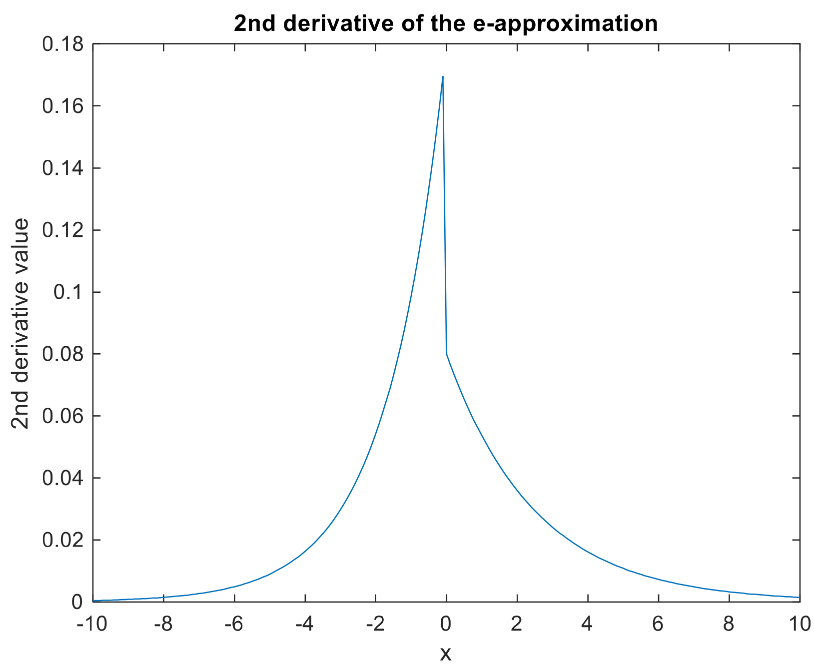


**FIGURE 4.8:** 2nd derivative of exponentially smoothed V-shape cost (different slopes)

Notice that the 2nd derivative is discontinuous at $x = 0$.

## 4.2 Butterfly-shape Transaction Cost Function

In practice, transaction costs can be more complicated than simple V-shape costs. Instead of constant transaction cost rates for buying and selling, piecewise constant rates with more kink points is becoming more and more popular in real trading market. No matter how many kink points they have, all of these piecewise-linear cost functions can be approximated by adapting smoothing techniques. In this essay, we consider an example with two more kink points than the V-shape cost function.

Assume there are two kinks at $x^+ > 0$ and $x^- < 0$. Let $m_1^+ > 0$ denote the cost rate of the line segment in $[0, x^+]$, and $m_2^+ > 0$ denote the cost rate of the line segment in $[x^+, \infty)$. Similarly, $m_1^- > 0$ and $m_2^- > 0$ represent the cost rates of the line segments in $[x^-, 0]$ and $(-\infty, x^-]$ respectively. Usually higher volumes of trading correspond to lower cost rates, which indicates $m_2^+ < m_1^+$, $m_2^- < m_1^-$ like the shape of a butterfly (shown in Figure 1.2). Without loss of generality, we can assume $|x^+| = |x^-| = k_1$. Then the formula of butterfly-shape transaction cost function is:

$$t_B(x) = \begin{cases} m_1^+ k_1 + m_2^+ (x - k_1) & x > k_1 \\ m_1^+ x & 0 \le x \le k_1 \\ -m_1^- x & -k_1 \le x < 0 \\ m_1^- k_1 + m_2^- (-k_1 - x) & x < -k_1 \end{cases} \qquad (4.11)$$

Note that this cost function is not convex because $m_2^+ < m_1^+$ and $m_2^- < m_1^-$.

### 4.2.1 Quadratic smoothing of Butterfly-shape transaction cost function

Compared with V-shape cost functions, Butterfly-shape cost functions have more kink points which introduce more non-differentiability. Therefore, if the quadratic smoothing method is used, additional quadratic smoothing need to be added at these kink points to construct a continuously differentiable approximation. Considering the kink point $x^+ = k_1$, we apply the quadratic approximation as:

$$\bar{t}_B(x) = \frac{m_2^+ - m_1^+}{4\varepsilon}[(x-k_1)^2 + \left(\frac{m_1^+ + m_2^+}{m_2^+ - m_1^+}\right)2\varepsilon(x-k_1) + \varepsilon^2] + m_1^+ k_1 \quad \text{for} \quad x \in [k_1 - \varepsilon, k_1 + \varepsilon] \ .$$

Similarly, the quadratic approximation can be applied at $x^- = -k_1$ with the same pattern. Then, after incorporating the smoothing function in V-shape at $x = 0$, the quadratic approximation to the Butterfly-shape cost function becomes:

$$\bar{t}_B^{\,q}(x) = \begin{cases} m_1^- k_1 + m_2^-(-k_1 - x) & x < -k_1 - \varepsilon \\ \frac{m_2^- - m_1^-}{4\varepsilon}[(x+k_1)^2 + \left(\frac{m_1^- + m_2^-}{m_1^- - m_2^-}\right)2\varepsilon(x+k_1) + \varepsilon^2] + m_1^- k_1, & -k_1 - \varepsilon \le x \le -k_1 + \varepsilon \\ -m_1^- x & -k_1 + \varepsilon < x < -\varepsilon \\ (\frac{m_1^+ + m_1^-}{4\varepsilon})x^2 + \left(\frac{m_1^+ - m_1^-}{2}\right)x + \left(\frac{m_1^+ + m_1^-}{4}\right)\varepsilon & \text{-}\varepsilon \le x \le \varepsilon \\ m_1^+ x & \varepsilon < x < k_1 - \varepsilon \\ \frac{m_2^+ - m_1^+}{4\varepsilon}[(x-k_1)^2 + \left(\frac{m_1^+ + m_2^+}{m_2^+ - m_1^+}\right)2\varepsilon(x-k_1) + \varepsilon^2] + m_1^+ k_1, & k_1 - \varepsilon \le x \le k_1 + \varepsilon \\ m_1^+ k_1 + m_2^+(x - k_1) & x > k_1 + \varepsilon \end{cases} \quad (4.12)$$

It is straightforward to show this quadratic approximation function $\bar{t}_B^{\,q}(x)$ is continuous. Differentiating (4.12) we get:

$$(\bar{t}_B^{\,q})'(x) = \begin{cases} -m_2^- & x < -k_1 - \varepsilon \\ \frac{m_2^- - m_1^-}{2\varepsilon}[x + k_1 + \left(\frac{m_1^- + m_2^-}{m_1^- - m_2^-}\right)\varepsilon] & -k_1 - \varepsilon \le x \le -k_1 + \varepsilon \\ -m_1^- & -k_1 + \varepsilon < x < -\varepsilon \\ (\frac{m_1^+ + m_1^-}{2\varepsilon})x + \left(\frac{m_1^+ - m_1^-}{2}\right) & -\varepsilon \le x \le \varepsilon \\ m_1^+ & \varepsilon < x < k_1 - \varepsilon \\ \frac{m_2^+ - m_1^+}{2\varepsilon}[x - k_1 + \left(\frac{m_1^+ + m_2^+}{m_2^+ - m_1^+}\right)\varepsilon] & k_1 - \varepsilon \le x \le k_1 + \varepsilon \\ m_2^+ & x > k_1 + \varepsilon \end{cases} \quad (4.13)$$

Clearly, the derivatives are matched at all kink points with:

$$\begin{cases} (\overline{t}_B{}^q)'(-k_1-\varepsilon) = -m_2^- \\ (\overline{t}_B{}^q)'(-k_1+\varepsilon) = -m_1^- \\ (\overline{t}_B{}^q)'(-\varepsilon) \quad = -m_1^- \\ (\overline{t}_B{}^q)'(\varepsilon) \quad = m_1^+ \\ (\overline{t}_B{}^q)'(k_1-\varepsilon) = m_1^+ \\ (\overline{t}_B{}^q)'(k_1+\varepsilon) = m_2^+ \end{cases} \tag{4.14}$$

It follows that $\overline{t}_B{}^q(x)$ is a continuously differentiable approximation to the Butterfly-shape transaction cost function.

Figures 4.9 and 4.10 illustrate the quadratic approximation to the Butterfly transaction cost function defined by (4.12) and the corresponding derivative function (4.13) respectively.

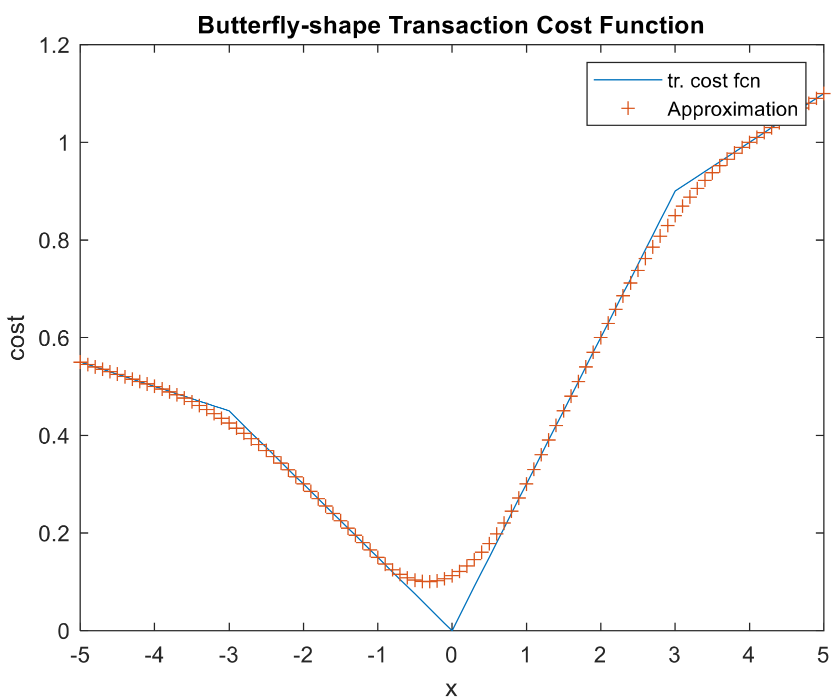


**FIGURE 4.9:** Quadratically smoothed B-shape cost

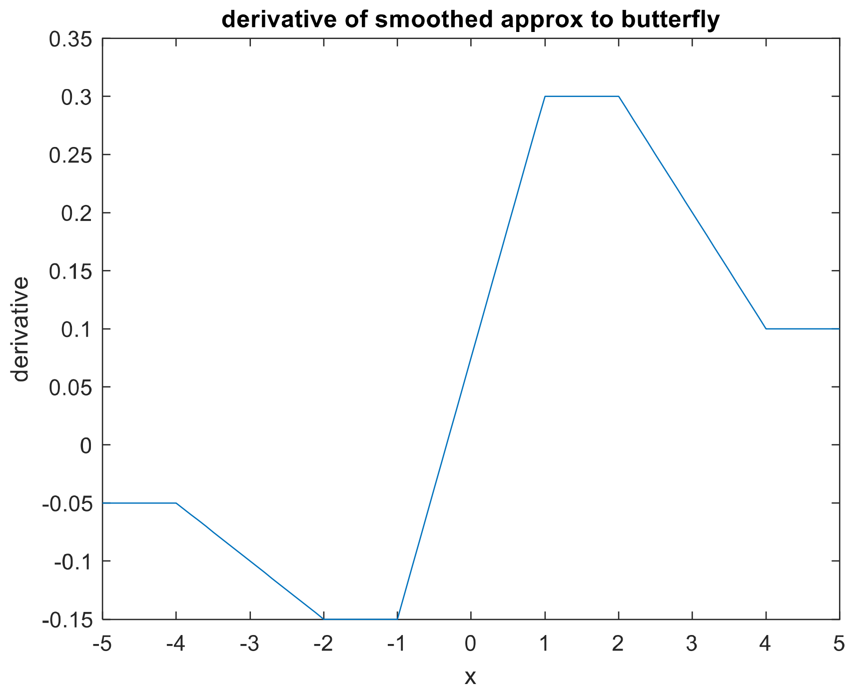


**FIGURE 4.10:** 1st derivative of quadratically smoothed B-shape cost

### 4.2.2 Exponential smoothing of Butterfly-shape transaction cost function

Moreover, we can also apply the exponential smoothing method to the Butterfly-shape transaction cost function. Recall that the Butterfly-shape cost function defined in (4.11) contains three kink points $x=0$, $x^{+}=k_1$ and $x^{-}=-k_1$. To use the exponential smoothing technique at all three kink points, we consider to divide the Butterfly-shape function into three V-shape functions based on intervals $x\in(-\infty,-k_1+\varepsilon]$, $x\in[-k_1+\varepsilon,k_1-\varepsilon]$ and $x\in[k_1-\varepsilon,+\infty)$ with a small $\varepsilon>0$. Note that $\varepsilon\ll k_1$ is required here. The central V-shape part in $[-k_1+\varepsilon,k_1-\varepsilon]$ with the kink point $x=0$ can be approximated by the exponential function (4.6). Then we propose an exponential approximation to the right V-shape part as:

$$f^{+}(x)\triangleq m_2^{+}x+(m_1^{+}-m_2^{+})\cdot k_1-[(m_1^{+}-m_2^{+})\varepsilon+\alpha_1 e^{\frac{1}{\alpha_1}m_1^{+}(\varepsilon-k_1)}]e^{-\alpha^{+}m_2^{+}(x+(\varepsilon-k_1))} \qquad (4.15)$$

for any $\alpha^{+} > 0$ where $\alpha_1 > 0$ is the smoothing parameter used in the central V-shape part. Similarly, the left V-shape part can be approximated by:

$$f^{-}(x) \triangleq -m_2^{-}x + (m_1^{-} - m_2^{-}) \cdot k_1 - [(m_1^{-} - m_2^{-})\varepsilon + \alpha_1 e^{\frac{1}{\alpha_1}m_1^{-}(\varepsilon - k_1)}]e^{\alpha^{-}m_2^{-}(x+(k_1-\varepsilon))} \tag{4.16}$$

for any $\alpha^{-} > 0$.

Notice that $t_B(x) - f^{+}(x) \to 0^{+}$ as $x \to \infty$ and $t_B(x) - f^{-}(x) \to 0^{+}$ as $x \to -\infty$. Combining these three exponentially smoothed V-shape functions together, we propose the Butterfly-shape cost function with exponential smoothing as:

$$\overline{t}_B^{\,e}(x) = \begin{cases} f^{+}(x) = m_2^{+}x + (m_1^{+} - m_2^{+}) \cdot k_1 - [(m_1^{+} - m_2^{+})\varepsilon + \alpha_1 e^{\frac{1}{\alpha_1}m_1^{+}(\varepsilon - k_1)}]e^{-\alpha^{+}m_2^{+}(x+(\varepsilon-k_1))} & x > k_1 - \varepsilon \\ f_1(x) = m_1^{+}x + \alpha_1 e^{-\frac{1}{\alpha_1}m_1^{+}x} & 0 \le x \le k_1 - \varepsilon \\ f_2(x) = -m_1^{-}x + \alpha_1 e^{\frac{1}{\alpha_1}m_1^{-}x} & \varepsilon - k_1 \le x < 0 \\ f^{-}(x) = -m_2^{-}x + (m_1^{-} - m_2^{-}) \cdot k_1 - [(m_1^{-} - m_2^{-})\varepsilon + \alpha_1 e^{\frac{1}{\alpha_1}m_1^{-}(\varepsilon - k_1)}]e^{\alpha^{-}m_2^{-}(x+(k_1-\varepsilon))} & x < \varepsilon - k_1 \end{cases} \tag{4.17}$$

for any $\alpha_1 > 0$, $\alpha^{+} > 0$ and $\alpha^{-} > 0$.

Clearly, $\overline{t}_B^{\,e}(x)$ is continuous as:

$$\begin{cases} f_1(0) = f_2(0) = \alpha_1 \\ f_1(k_1 - \varepsilon) = f^{+}(k_1 - \varepsilon) = m_1^{+}(k_1 - \varepsilon) + \alpha_1 e^{\frac{1}{\alpha_1}m_1^{+}(\varepsilon - k_1)} \\ f_2(\varepsilon - k_1) = f^{-}(\varepsilon - k_1) = m_1^{-}(k_1 - \varepsilon) + \alpha_1 e^{\frac{1}{\alpha_1}m_1^{-}(\varepsilon - k_1)} \end{cases} \tag{4.18}$$

Differentiating $\overline{t}_B^{\,e}(x)$ we get:

$$(\overline{t}_B^{\,e})'(x) = \begin{cases} (f^{+})'(x) = m_2^{+} + \alpha^{+}m_2^{+}[(m_1^{+} - m_2^{+})\varepsilon + \alpha_1 e^{\frac{1}{\alpha_1}m_1^{+}(\varepsilon - k_1)}]e^{-\alpha^{+}m_2^{+}(x+(\varepsilon-k_1))} & x > k_1 - \varepsilon \\ (f_1)'(x) = m_1^{+} - m_1^{+}e^{-\frac{1}{\alpha_1}m_1^{+}x} & 0 \le x \le k_1 - \varepsilon \\ (f_2)'(x) = -m_1^{-} + m_1^{-}e^{\frac{1}{\alpha_1}m_1^{-}x} & \varepsilon - k_1 \le x < 0 \\ (f^{-})'(x) = -m_2^{-} - \alpha^{-}m_2^{-}[(m_1^{-} - m_2^{-})\varepsilon + \alpha_1 e^{\frac{1}{\alpha_1}m_1^{-}(\varepsilon - k_1)}]e^{\alpha^{-}m_2^{-}(x+(k_1-\varepsilon))} & x < \varepsilon - k_1 \end{cases} \tag{4.19}$$

To check the continuity of $(\overline{t}_B^{\,e})'(x)$, we consider the values at demarcation points:

$$\begin{cases} (f_1)'(0)=0 \\ (f_2)'(0)=0 \\ (f^+)'(k_1-\varepsilon)=m_2^+ + \alpha^+ m_2^+[(m_1^+ - m_2^+)\varepsilon + \alpha_1 e^{\frac{1}{\alpha_1}m_1^+(\varepsilon-k_1)}] \\ (f_1)'(k_1-\varepsilon)=m_1^+ - m_1^+ e^{\frac{1}{\alpha_1}m_1^+(\varepsilon-k_1)} \\ (f_2)'(\varepsilon-k_1)=-m_1^- + m_1^- e^{\frac{1}{\alpha_1}m_1^-(\varepsilon-k_1)} \\ (f^-)'(\varepsilon-k_1)=-m_2^- - \alpha^- m_2^-[(m_1^- - m_2^-)\varepsilon + \alpha_1 e^{\frac{1}{\alpha_1}m_1^-(\varepsilon-k_1)}] \end{cases} \tag{4.20}$$

It follows that $(\overline{t}_B^{\,e})'(x)$ is continuous if we take:

$$\begin{cases} \alpha^+ = \dfrac{m_1^+ - m_2^+ - m_1^+ e^{\frac{1}{\alpha_1}m_1^+(\varepsilon-k_1)}}{m_2^+[(m_1^+ - m_2^+)\varepsilon + \alpha_1 e^{\frac{1}{\alpha_1}m_1^+(\varepsilon-k_1)}]} \\ \alpha^- = \dfrac{m_1^- - m_2^- - m_1^- e^{\frac{1}{\alpha_1}m_1^-(\varepsilon-k_1)}}{m_2^-[(m_1^- - m_2^-)\varepsilon + \alpha_1 e^{\frac{1}{\alpha_1}m_1^-(\varepsilon-k_1)}]} \end{cases} \tag{4.21}$$

In conclusion, the exponentially smoothed Butterfly-shape function $\overline{t}_B^{\,e}(x)$ defined in (4.17) and (4.21) is a continuously differentiable approximation to the Butterfly-shape transaction cost function $t_B(x)$.

Figures 4.11 and 4.12 illustrate the exponential approximation and its corresponding first order derivative of a sample Butterfly-shape transaction cost function with $m_1^+ = 0.2,\ m_2^+ = 0.2,\ m_1^- = 0.5,\ m_2^- = 0.25,\ k_1 = 10,\ \varepsilon = 0.5$.

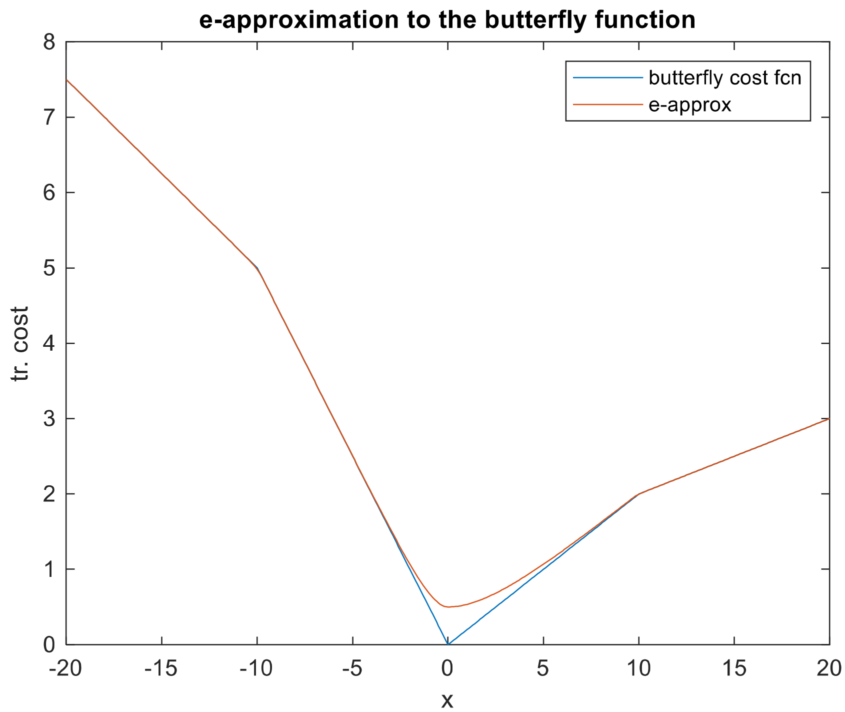


**FIGURE 4.11:** Exponentially smoothed B-shape cost

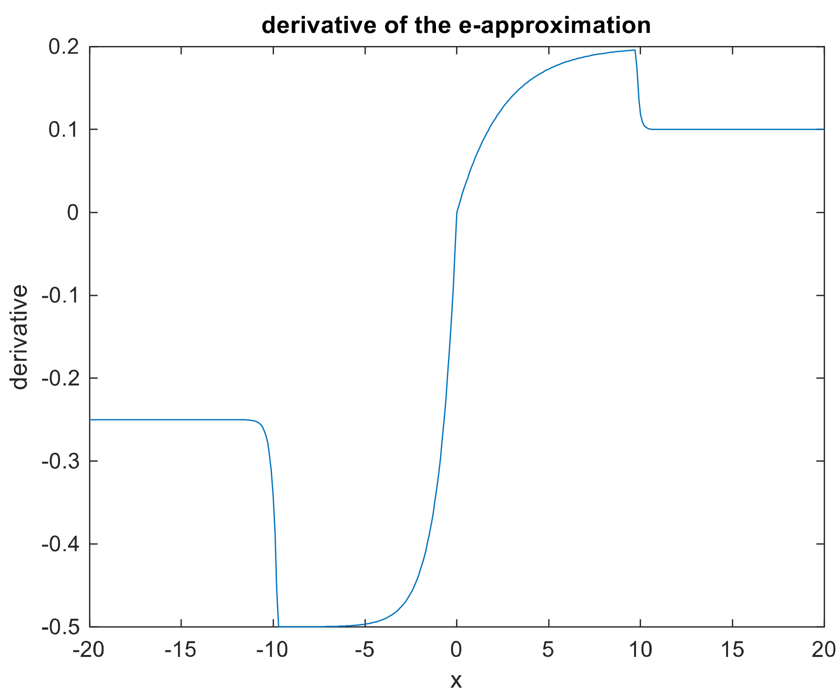


**FIGURE 4.12:** 1st derivative of exponentially smoothed B-shape cost

### 4.3 CVaR Portfolio Optimization Model with Transaction Costs

In the CVaR portfolio optimization, the standard-deviation type of risk is minimized as the potential loss while achieving the expected return. When transaction costs are involved, however, the loss consists of not only the potential risk loss but also transaction costs. Therefore, it is necessary to add transaction cost functions into objective functions of portfolio optimization models.

After smoothing transaction cost functions to avoid discontinuous differentiability, the CVaR robust mean-standard deviation portfolio optimization problem with transaction costs can be modeled as follows:

$$\begin{aligned} \min_{\alpha,x} \ & \sqrt{x^T H x} + \sum_{i=1}^{n} \bar{t}(x_i - x_i^p) \\ \text{subject to} \ & 1 + \rho_e^S(\alpha\ , x)_\alpha = 0 \\ & \alpha + \rho_e^S(\alpha, x) = \tau \\ & e^T x = W \end{aligned} \tag{4.22}$$

where $n$ is the number of assets , $\bar{t}$ is a smoothed transaction cost function and $x_i, x_i^p$ are the current and previous wealth allocations in $i^{th}$ asset respectively.

Note that the objective function in (4.22) is a convex nonlinear function when we use the V-shape transaction cost function. However, it is no longer convex if the Butterfly-shape transaction cost function is used. The first order derivatives of smoothed Butterfly-shape cost functions (Figures 4.10 and 4.12) show the non-convexity. Therefore, global optimization techniques are needed to find the global optimal solution. Moreover, the Hessian matrix of the standard deviation risk term $f(x) \triangleq (x^T H x)^{\frac{1}{2}}$ is positive definite in the linear subspace of interest. (See Appendix A)

# Chapter 5

# Numerical Experiments

In this chapter, we present some illustrative examples with different combinations of parameters to demonstrate the effect of CVaR portfolio optimization model with transaction costs. To be specific, we consider a portfolio rebalancing problem with 15 optional assets and 60-month synthetic monthly return data (shown in appendix). Monthly rebalancing portfolio optimizations are conducted in the last 24 months with a zero-starting portfolio. At the beginning of each period, 36-month previous data are used to get the corresponding expected returns and covariance matrices. The optimal portfolio considering CVaR-type expected return with transaction costs for each month can be solved by the optimization model (4.22) with MATLAB. The new wealth is calculated by multiplying the current portfolio and the return vector in that month. In the following experiments, transaction costs are subtracted at the end of each period. Then the wealth paths can be formed by repeating this procedure in the last 24 months. In order to illustrate the effect of this CVaR portfolio optimization model with transaction costs clearly, the typical CVaR portfolio optimization model without transaction cost is applied as well to make a comparison. Moreover, considering the simplest uniform portfolio (i.e., average allocations at first and holding this portfolio unchanged to the end) is necessary to show the general market trend.

In order to solve CVaR optimization problems with transaction costs accurately, the exponential approximation method with corresponding parameters $\alpha_1 = 0.3$, $\varepsilon = 1$ for CVaR terms and transaction cost functions are used in the following experiments. In general, we test three portfolio strategies with an initial wealth $W = 1000$. To calculate the CVaR term, we use $\beta = 0.95$ and $M = 1000$ in this essay. The first strategy (denoted as 'CVaR+TR portfolio') is rebalancing portfolios by the CVaR portfolio optimization model with transaction costs. The second strategy (denoted as 'CVaR portfolio') is based on the typical CVaR portfolio optimization model without transaction cost. Investing equally at the beginning and holding the portfolio unchangeably (denoted as 'holding portfolio') is the last strategy.

Firstly, we consider the impact of our CVaR portfolio optimization model with V-shape transaction cost function by using different combinations of expected return $\tau$ and transaction cost rates $m_1^+ > 0,\ m_1^- > 0$. To highlight the effect, both high-level and low-level expected returns and transaction cost rates are tested with the synthetic data. Set $\tau = 1.05$ as the high-level expected return and $\tau = 1.01$ as the low-level expected return. $m_1^+ = m_1^- = 0.05$ and $m_1^+ = m_1^- = 0.01$ represent high-level and low-level transaction cost rates respectively. Figures 5.1-5.4 are the results of three portfolio strategies under different settings of parameters.

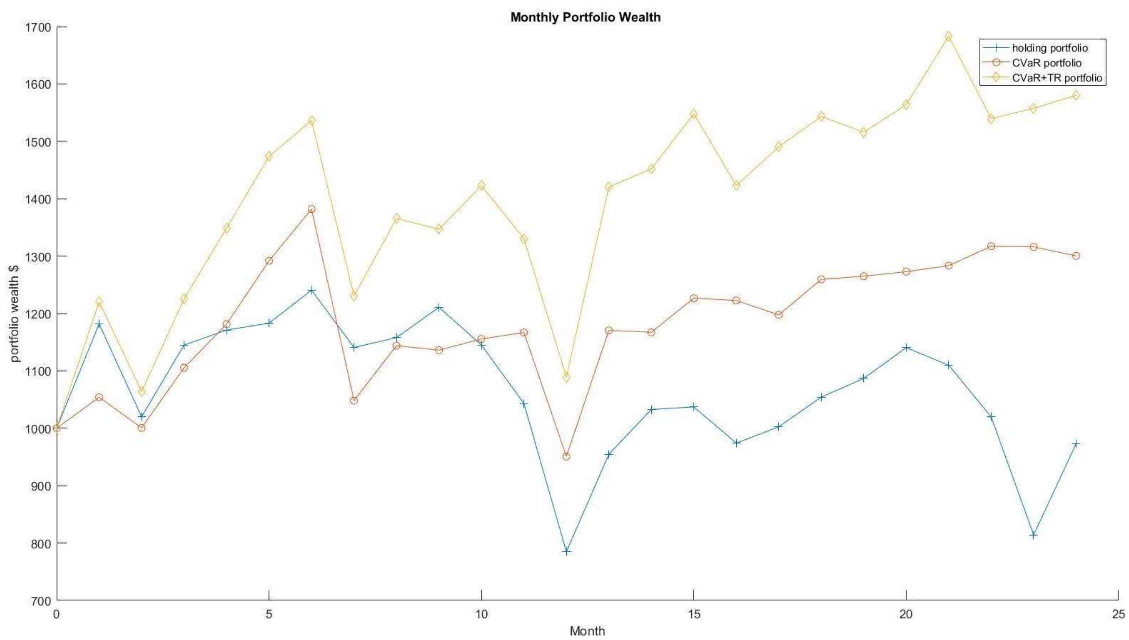


**FIGURE 5.1:** Wealth with $\tau = 1.05$ and V-shape cost $m_1^+ = m_1^- = 0.05$

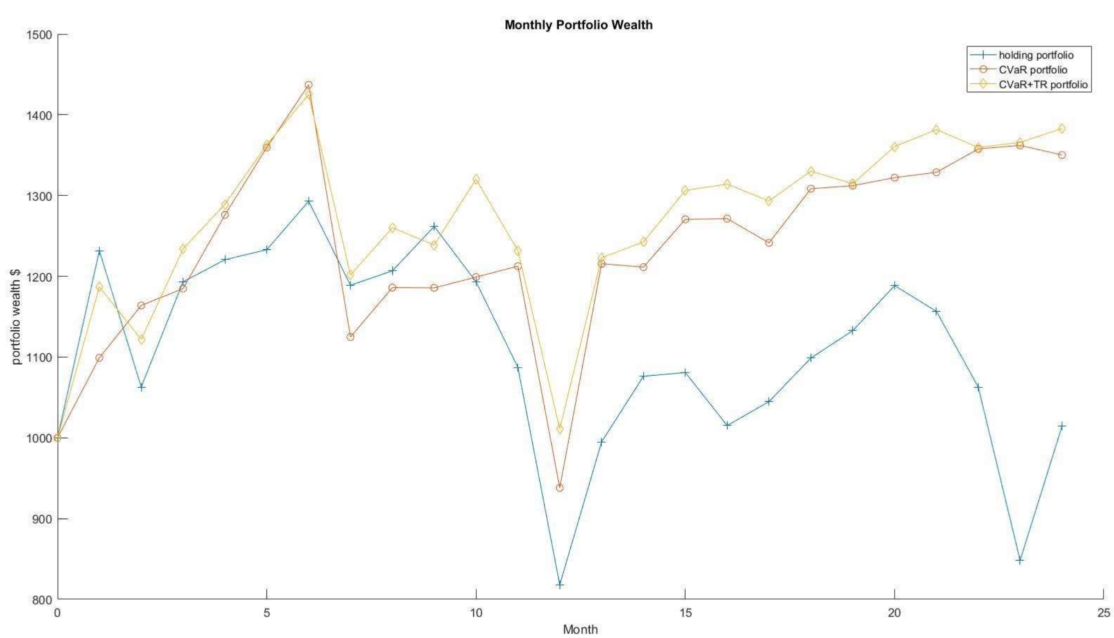


**FIGURE 5.2:** Wealth with $\tau = 1.05$ and V-shape cost $m_1^+ = m_1^- = 0.01$

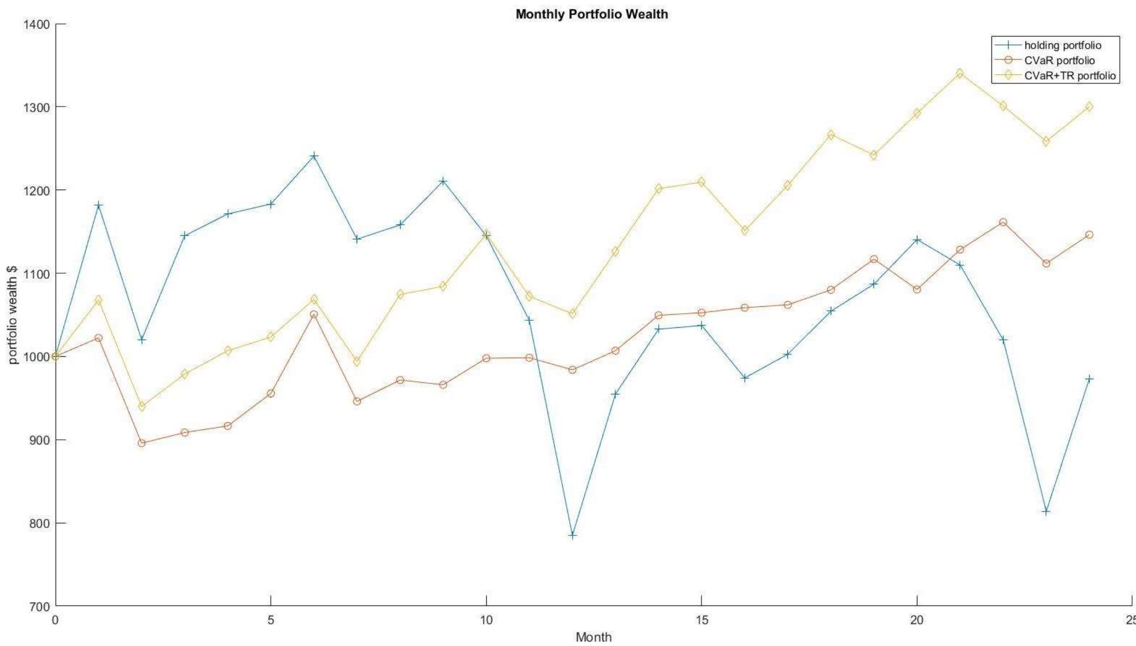


**FIGURE 5.3:** Wealth with $\tau = 1.01$ and V-shape cost $m_1^+ = m_1^- = 0.05$

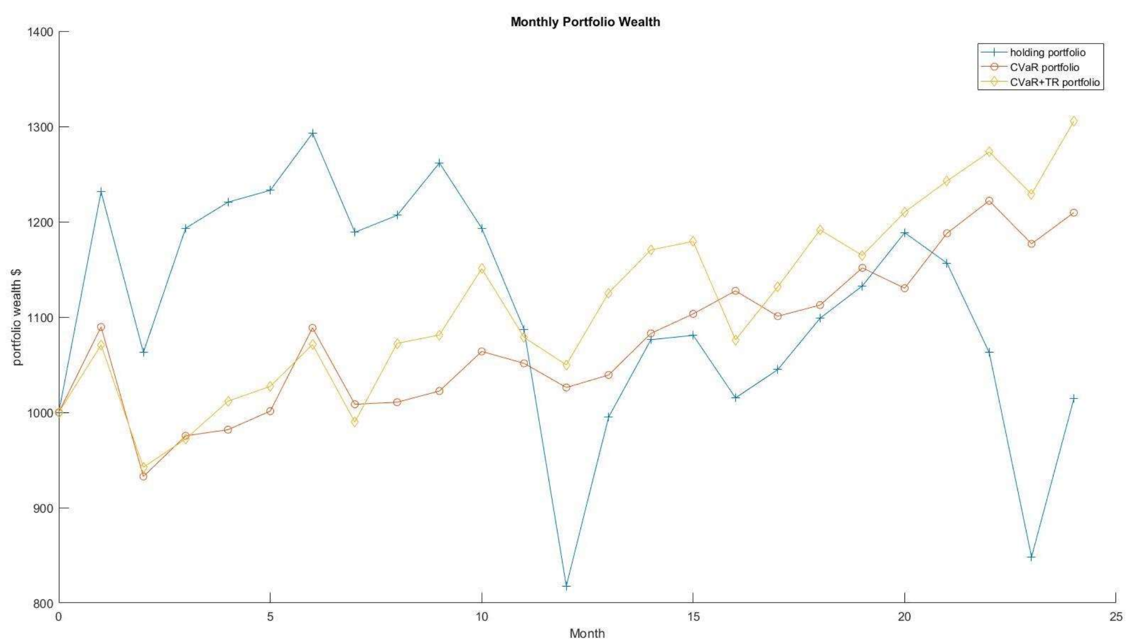


**FIGURE 5.4:** Wealth with $\tau = 1.01$ and V-shape cost $m_1^+ = m_1^- = 0.01$

As we can see from the experimental results above, the fluctuations of curves for ‘CVaR portfolio’ and ‘CVaR+TR portfolio’ are more intense in Figures 5.1 and 5.2 compared with the fluctuations in Figures 5.3 and 5.4. It is reasonable because high-level expected returns are always accompanied by high-level risks which lead to dramatic changes. Comparing the performance of two CVaR optimization strategies, we can observe that the CVaR with costs model performs much better than the CVaR model without cost, especially when the expected return and cost rates are higher. That is because the magnitude of transaction costs is determined by the product of trading amounts and cost rates. As shorting is allowed in our models, a higher expected return requires more buying and selling. Therefore, the higher expected return and cost rates are, the more effective our CVaR portfolio optimization model with transaction costs is. It can be verified by the larger gap between ‘CVaR portfolio’ and ‘CVaR+TR portfolio’ in Figure 5.1 compared with the gap in Figure 5.2.

Secondly, the effect of our CVaR portfolio optimization model with Butterfly-shape transaction cost function are tested as follows. Same as the experiments of V-shape transaction cost functions above, different levels of expected returns and transaction cost rates are used with the same synthetic return data. High-level and low-level expected returns are set as $\tau = 1.05$ and $\tau = 1.01$ respectively. $m_1^+ = m_1^- = 0.05$ and $m_2^+ = m_2^- = 0.005$ represent high-level transaction cost rates while $m_1^+ = m_1^- = 0.01$ and $m_2^+ = m_2^- = 0.005$ represent low-level transaction cost rates. Assuming the kink points are $x^+ = 100$ and $x^- = -100$, the numerical results are shown in Figures 5.5-5.8.

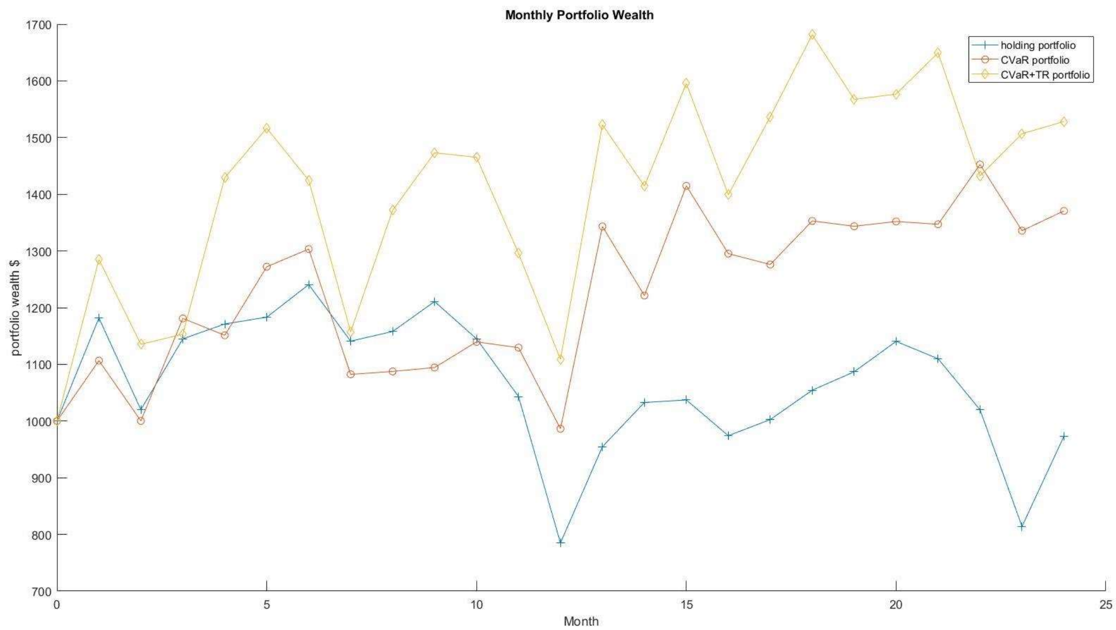


**FIGURE 5.5:** Wealth with $\tau = 1.05$ and B-shape cost $m_1^+ = m_1^- = 0.05, m_2^+ = m_2^- = 0.005$

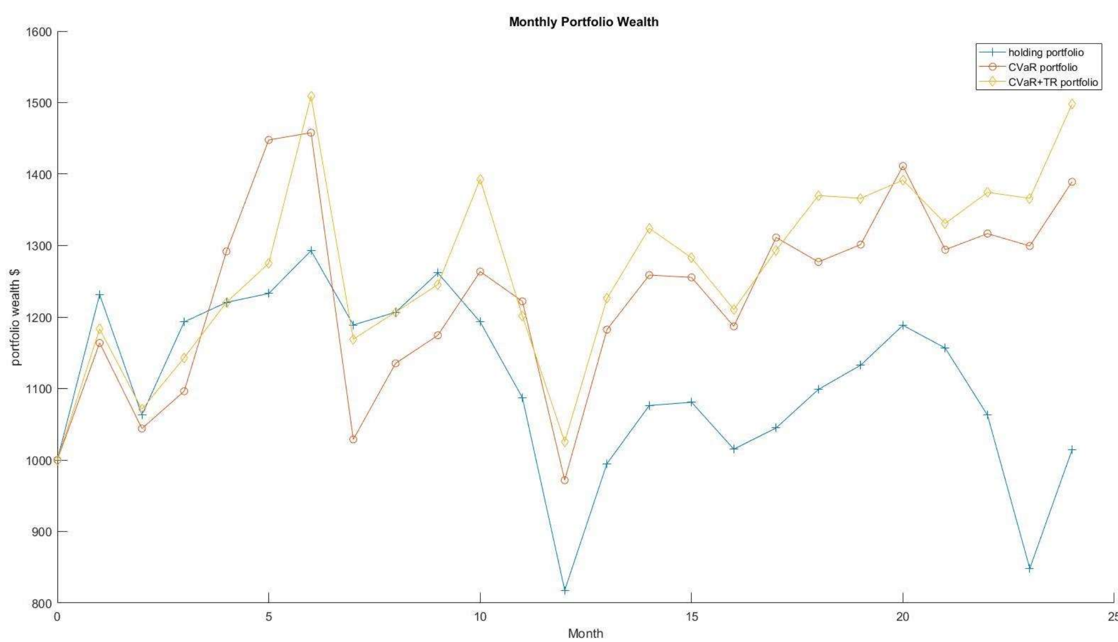


**FIGURE 5.6:** Wealth with $\tau = 1.05$ and B-shape cost $m_1^+ = m_1^- = 0.01, m_2^+ = m_2^- = 0.005$

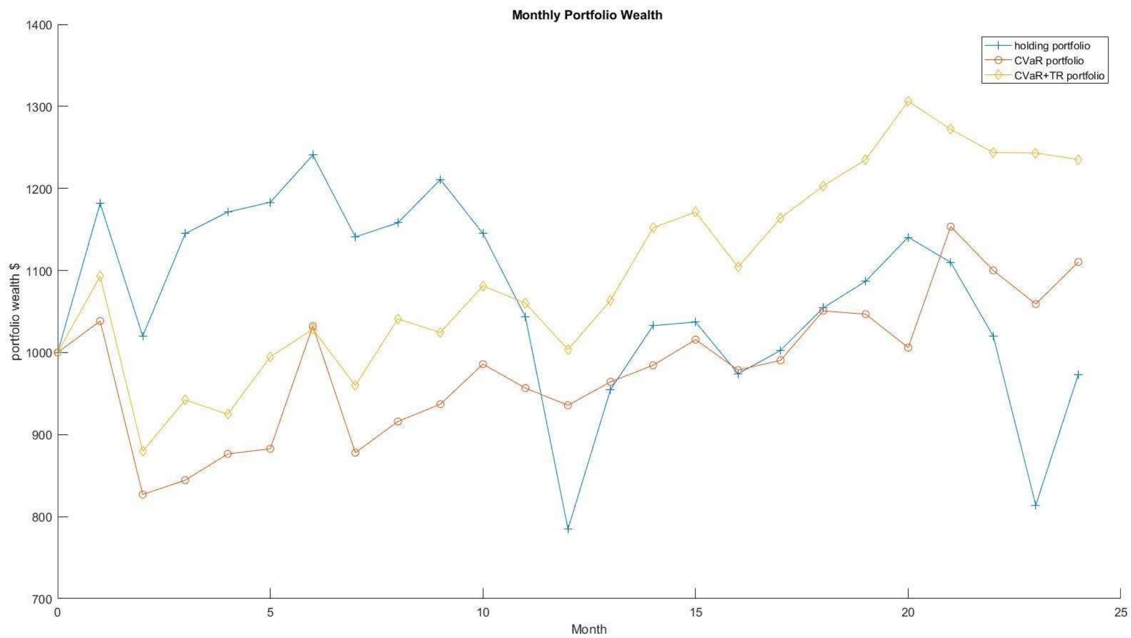


**FIGURE 5.7:** Wealth with $\tau = 1.01$ and B-shape cost $m_1^+ = m_1^- = 0.05$, $m_2^+ = m_2^- = 0.005$

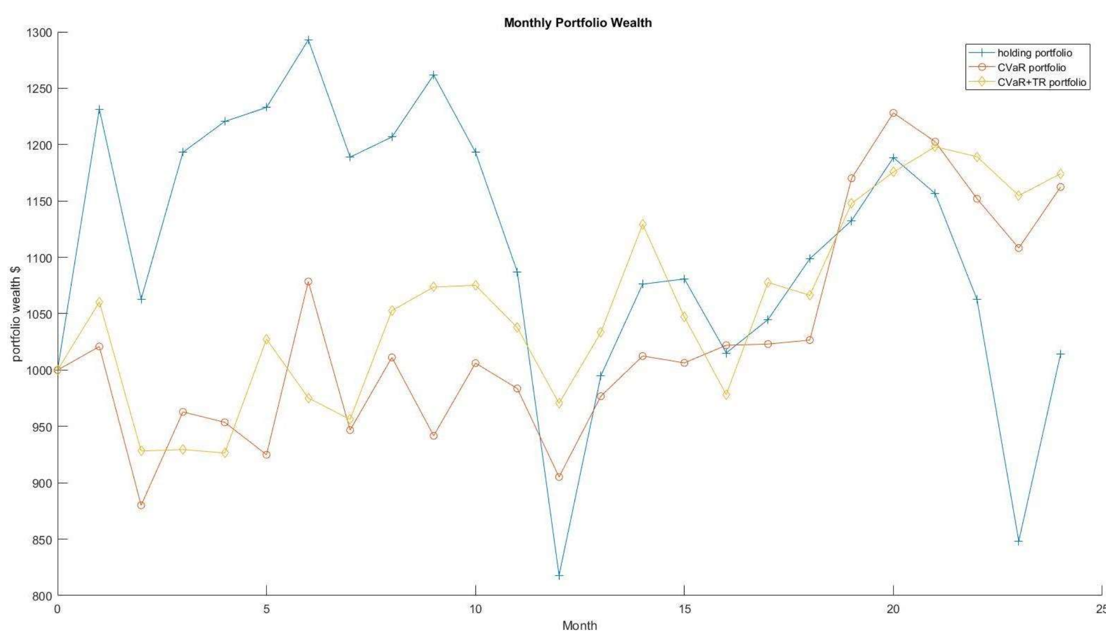


**FIGURE 5.8:** Wealth with $\tau = 1.01$ and B-shape cost $m_1^+ = m_1^- = 0.01$, $m_2^+ = m_2^- = 0.005$

Because the V-shape and B-shape transaction cost functions used in our experiments are similar except for the cost rates when trading amounts are large, the general patterns of Figures for B-shape cost and V-shape costs are similar as well. However, the impact of the B-shape transaction cost function compared with V-shape cost function does exist, especially when the expected returns $\tau$ are high-level. To be specific, the difference is significant in the comparison between Figure 5.1 and Figure 5.5. The large expected return leads to more buying and selling which correspond to smaller transaction cost rates. Therefore, the gap between 'CVaR portfolio' and 'CVaR+TR portfolio' for B-shape cost function in Figure 5.5 is narrower than the gap for V-shape cost function in Figure 5.1. Moreover, unlike the obvious difference between Figure 5.1 and Figure 5.2, the difference between Figure 5.5 and Figure 5.6 is relatively small because large amounts of trading caused by the high-level expected return drive cost rates to the smaller part in B-shape cost functions.

In general, the CVaR portfolio optimization model with transaction costs is effective when the transaction costs are large. Transaction costs are positively related to the expected returns and cost rates in simple V-shape cost functions. However, the analysis of the impact of the B-shape transaction cost function is more complicated because of its varying transaction cost rates. Large expected returns lead to large trading amounts which corresponds to smaller cost rates. In conclusion, it is necessary to consider transaction costs in portfolio optimization problem, especially when the expected returns or transaction cost rates are high-level.

# Chapter 6

# Conclusions

In this essay, a CVaR robust portfolio optimization model with transaction costs and some numerical techniques for solving it are introduced. To improve the CVaR robust portfolio optimization model in terms of numerical convergence stability, we proposed a new smoothing technique (exponential approximation) which is twice continuously differentiable.

Then transaction cost are introduced into CVaR portfolio optimization. Smoothing approaches for cost functions are illustrated by two typical kinds of cost function examples, V-shape and Butterfly-shape functions. After numerical smoothing, the CVaR robust portfolio optimization model with transaction costs has been formulated to help investors find the optimal portfolios. Finally, the effect of the CVaR robust portfolio optimization model with transaction costs is shown by numerical experiments.

Prospective research can include:

- Considering the smoothing accuracy and robustness of the smoothing parameters.
- Using the CVaR type of covariance matrix based on sampled data.
- Investigating more advanced global optimization methods.
- Conducting a computational cost study of different smoothing approaches.
- Finding other possible techniques to get higher order continuous differentiability.

# APPENDICES

# Appendix A

# Theorem

**Theorem:** If $f(x) \triangleq \sqrt{x^T Hx}$ where $H$ is symmetric positive definite of order $n$, then for $x \neq 0$, the Hessian matrix $\nabla^2 f(x)$ has $n-1$ positive eigenvalues and a single zero eigenvalue corresponding to eigenvector $x$.

**Proof:** Applying the chain rule, we have the gradient of $f(x)$:

$$\nabla f(x) = (x^T Hx)^{-\frac{1}{2}} \cdot Hx$$

Therefore, with an additional application of the chain rule, we get the corresponding Hessian matrix:

$$\begin{aligned}\nabla^2 f(x) &= -(x^T Hx)^{-\frac{3}{2}}(Hx)\ (Hx)^T + (x^T Hx)^{-\frac{1}{2}} H \\ &= \frac{-(Hx)(Hx)^T + (x^T Hx) \cdot H}{(x^T Hx)^{\frac{3}{2}}}\end{aligned}$$

Note that the gradient and Hessian of $f$ exist, under the assumption that $H$ is SPD as long as $x \neq 0$ (i.e., we don't consider the empty portfolio).

But considering the numerator,

$$\begin{aligned}&[-(Hx)(Hx)^T + (x^T Hx) \cdot H]x \\ &= [-(x^T Hx) + (x^T Hx)]Hx \\ &= 0.\end{aligned}$$

If $M$ is any SPD matrix of order $n$, the matrix $M - vv^T$ has at least $n-1$ positive eigenvalues (consider the null-space of $v$). So letting $(x^T Hx)^{-\frac{1}{2}} H$ and $Hx$ play the roles of $M$ and $v$ respectively, the result follows. □

**Corollary:** If vector $x$ satisfies the constraint $e^T x = W > 0$, the reduction of $\nabla^2 f(x)$ onto $null(e^T)$ (denoted by $\bar{\nabla}^2 f(x) = Z^T \nabla^2 f(x) Z$, where the columns of $Z$ form a basis for $null(e^T)$) is SPD.

**Proof:** Clearly $x \notin null(e^T)$ since $e^T x = W > 0$. It follows that $\bar{\nabla}^2 f(x)$ is SPD. □

# Appendix B

# Data

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1.1613 | 1.1840 | 1.1490 | 1.4688 | 1.6717 | 1.4467 | 1.0987 | 1.2252 | 1.1832 | 1.1232 | 1.2492 | 1.1331 | 1.1717 | 1.1858 | 1.2082 |
| 0.9459 | 0.9414 | 0.9870 | 0.9590 | 0.9559 | 1.0934 | 1.1104 | 1.0557 | 1.1043 | 0.9707 | 0.8770 | 0.9819 | 1.0348 | 1.0248 | 1.0598 |
| 1.0291 | 1.0221 | 1.0544 | 1.0072 | 1.0741 | 0.8671 | 1.0102 | 1.0117 | 1.0275 | 1.0591 | 0.9683 | 1.0646 | 1.0029 | 0.9869 | 1.0382 |
| 1.1053 | 1.0094 | 1.0220 | 0.8758 | 0.9468 | 1.1423 | 0.9866 | 0.9888 | 0.9858 | 0.9729 | 0.8762 | 0.9538 | 0.9957 | 1.0272 | 1.0330 |
| 0.9360 | 1.0200 | 1.0000 | 1.1819 | 1.0649 | 1.1882 | 1.0495 | 1.0563 | 1.0728 | 1.1408 | 1.2133 | 1.0970 | 0.9816 | 1.0510 | 0.9840 |
| 1.1012 | 1.0596 | 1.1201 | 0.9703 | 1.1160 | 1.1017 | 1.0557 | 1.0282 | 1.0126 | 1.0076 | 1.0681 | 0.9972 | 1.0743 | 0.9815 | 1.0765 |
| 1.0005 | 0.9986 | 0.9990 | 1.0347 | 1.0178 | 1.1564 | 1.0376 | 1.0242 | 1.0232 | 1.0671 | 1.0370 | 1.1551 | 0.9981 | 1.0366 | 1.0816 |
| 1.0711 | 1.0285 | 1.0523 | 0.9369 | 0.8966 | 0.9279 | 1.0484 | 1.0265 | 1.0439 | 1.0642 | 0.9854 | 1.1071 | 1.0429 | 1.0026 | 1.1094 |
| 1.1084 | 1.0312 | 1.0459 | 1.0793 | 1.2031 | 0.9313 | 1.0565 | 1.0454 | 1.0625 | 1.0025 | 0.9457 | 1.0778 | 1.0592 | 1.0131 | 1.0433 |
| 1.0272 | 1.0898 | 1.0734 | 0.8428 | 0.7949 | 0.8640 | 0.9490 | 1.0319 | 0.9322 | 1.0026 | 0.8156 | 0.9323 | 1.0243 | 0.9532 | 0.9359 |
| 0.9276 | 0.9245 | 0.9700 | 0.9996 | 0.8457 | 0.8220 | 0.8792 | 0.8450 | 0.8548 | 0.9526 | 0.9435 | 0.9462 | 0.9020 | 0.9052 | 0.9353 |
| 0.8732 | 0.8301 | 0.9255 | 0.5659 | 0.4014 | 0.5478 | 0.7297 | 0.7254 | 0.6130 | 0.7924 | 0.5628 | 0.8243 | 0.9475 | 0.9413 | 0.9275 |
| 1.1613 | 1.1840 | 1.1490 | 1.4688 | 1.6717 | 1.4467 | 1.0987 | 1.2252 | 1.1832 | 1.1232 | 1.2492 | 1.1331 | 1.1717 | 1.1858 | 1.2082 |
| 1.0012 | 1.0320 | 1.0303 | 1.0372 | 1.1347 | 1.1395 | 1.0260 | 1.0320 | 0.9918 | 1.0034 | 0.9226 | 0.9568 | 1.0006 | 1.0025 | 0.9656 |
| 0.9824 | 1.1090 | 1.1005 | 1.0542 | 1.0046 | 0.9545 | 1.0756 | 1.0325 | 1.0423 | 1.0280 | 0.8538 | 1.0050 | 0.9847 | 1.0063 | 1.0644 |
| 1.0680 | 0.9529 | 0.9916 | 0.8970 | 0.9957 | 0.8294 | 0.9975 | 0.9867 | 0.9646 | 1.0095 | 1.0655 | 0.9825 | 0.9388 | 0.9866 | 0.9594 |
| 0.9333 | 0.9232 | 0.9625 | 1.1076 | 1.0066 | 1.0149 | 0.9522 | 0.9568 | 0.9050 | 0.9813 | 0.9007 | 0.9772 | 0.9900 | 0.9806 | 0.9679 |
| 0.9986 | 0.9910 | 1.0303 | 1.1266 | 1.0126 | 1.1431 | 0.9863 | 0.9453 | 0.9987 | 0.9386 | 1.2598 | 0.9636 | 0.9883 | 1.0045 | 1.0849 |
| 1.1297 | 1.0845 | 1.0651 | 1.0719 | 0.9440 | 1.1098 | 0.9990 | 0.9616 | 0.9890 | 1.0588 | 0.8729 | 1.1294 | 0.9530 | 1.0191 | 1.0284 |
| 1.0711 | 1.0285 | 1.0523 | 0.9369 | 0.8966 | 0.9279 | 1.0484 | 1.0265 | 1.0439 | 1.0642 | 0.9854 | 1.1071 | 1.0429 | 1.0026 | 1.1094 |
| 1.1084 | 1.0312 | 1.0459 | 1.0793 | 1.2031 | 0.9313 | 1.0565 | 1.0454 | 1.0625 | 1.0025 | 0.9457 | 1.0778 | 1.0592 | 1.0131 | 1.0433 |
| 1.0272 | 1.0898 | 1.0734 | 0.8428 | 0.7949 | 0.8640 | 0.9490 | 1.0319 | 0.9322 | 1.0026 | 0.8156 | 0.9323 | 1.0243 | 0.9532 | 0.9359 |
| 0.9276 | 0.9245 | 0.9700 | 0.9996 | 0.8457 | 0.8220 | 0.8792 | 0.8450 | 0.8548 | 0.9526 | 0.9435 | 0.9462 | 0.9020 | 0.9052 | 0.9353 |
| 0.8732 | 0.8301 | 0.9255 | 0.5659 | 0.4014 | 0.5478 | 0.7297 | 0.7254 | 0.6130 | 0.7924 | 0.5628 | 0.8243 | 0.9475 | 0.9413 | 0.9275 |
| 1.1613 | 1.1840 | 1.1490 | 1.4688 | 1.6717 | 1.4467 | 1.0987 | 1.2252 | 1.1832 | 1.1232 | 1.2492 | 1.1331 | 1.1717 | 1.1858 | 1.2082 |
| 0.9778 | 0.9997 | 0.9881 | 1.0857 | 1.0879 | 0.9599 | 0.9574 | 0.9621 | 0.9810 | 0.9687 | 1.0501 | 0.9833 | 1.0080 | 0.9928 | 1.0052 |
| 1.0833 | 1.1174 | 1.0808 | 0.9833 | 1.0046 | 1.0713 | 1.1160 | 1.0853 | 1.0950 | 1.0582 | 1.0045 | 1.0073 | 1.0911 | 1.1008 | 1.0453 |
| 1.1470 | 0.9807 | 1.0455 | 1.0411 | 0.9291 | 0.9763 | 0.9878 | 0.9969 | 0.9792 | 1.0853 | 1.0566 | 1.2554 | 1.0174 | 1.0467 | 1.0465 |
| 0.9980 | 0.9964 | 1.0352 | 1.0661 | 1.1089 | 1.0588 | 0.9915 | 0.9515 | 1.0176 | 1.0041 | 1.1105 | 0.8752 | 1.0265 | 1.0615 | 0.9972 |
| 0.9610 | 0.8966 | 0.9329 | 0.8055 | 0.8328 | 1.0597 | 0.9669 | 0.9980 | 0.9149 | 0.9812 | 0.7516 | 0.9254 | 1.0131 | 0.9811 | 0.9838 |
| 0.8421 | 1.0440 | 1.0556 | 0.9694 | 1.0091 | 0.8992 | 1.0252 | 0.8593 | 0.9992 | 1.0281 | 0.9523 | 0.8839 | 1.0393 | 1.0734 | 1.0764 |
| 1.0711 | 1.0285 | 1.0523 | 0.9369 | 0.8966 | 0.9279 | 1.0484 | 1.0265 | 1.0439 | 1.0642 | 0.9854 | 1.1071 | 1.0429 | 1.0026 | 1.1094 |
| 1.1084 | 1.0312 | 1.0459 | 1.0793 | 1.2031 | 0.9313 | 1.0565 | 1.0454 | 1.0625 | 1.0025 | 0.9457 | 1.0778 | 1.0592 | 1.0131 | 1.0433 |
| 1.0272 | 1.0898 | 1.0734 | 0.8428 | 0.7949 | 0.8640 | 0.9490 | 1.0319 | 0.9322 | 1.0026 | 0.8156 | 0.9323 | 1.0243 | 0.9532 | 0.9359 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0.9276** | **0.9245** | **0.9700** | **0.9996** | **0.8457** | **0.8220** | **0.8792** | **0.8450** | **0.8548** | **0.9526** | **0.9435** | **0.9462** | **0.9020** | **0.9052** | **0.9353** |
| **0.8732** | **0.8301** | **0.9255** | **0.5659** | **0.4014** | **0.5478** | **0.7297** | **0.7254** | **0.6130** | **0.7924** | **0.5628** | **0.8243** | **0.9475** | **0.9413** | **0.9275** |
| **1.1613** | **1.1840** | **1.1490** | **1.4688** | **1.6717** | **1.4467** | **1.0987** | **1.2252** | **1.1832** | **1.1232** | **1.2492** | **1.1331** | **1.1717** | **1.1858** | **1.2082** |
| **0.8397** | **0.9051** | **0.8810** | **0.6999** | **0.7963** | **0.8770** | **0.8538** | **0.8471** | **0.7674** | **0.8592** | **0.9984** | **1.0075** | **0.8802** | **0.9299** | **0.8657** |
| **1.0779** | **1.0943** | **1.0425** | **1.3897** | **1.1408** | **1.1934** | **1.0839** | **1.2049** | **1.2678** | **1.0638** | **1.1203** | **1.0298** | **1.0460** | **0.9942** | **1.0958** |
| **1.0438** | **1.0112** | **1.0878** | **1.0670** | **0.9982** | **0.9943** | **1.0106** | **1.0038** | **1.0082** | **1.0042** | **1.1098** | **1.0481** | **1.0239** | **1.0173** | **0.9072** |
| **1.0996** | **1.0526** | **1.0537** | **1.0484** | **0.9970** | **1.0371** | **0.9719** | **0.9761** | **0.9702** | **0.9707** | **0.9905** | **0.9566** | **1.0202** | **0.9764** | **1.0253** |
| **1.0952** | **1.0033** | **1.0974** | **1.0540** | **1.0271** | **1.0936** | **1.1328** | **1.0639** | **1.1246** | **1.0232** | **1.0706** | **1.0247** | **1.0069** | **0.9533** | **0.9334** |
| **0.8367** | **0.8968** | **0.9488** | **0.8745** | **0.9340** | **0.9355** | **0.9143** | **0.8803** | **0.8786** | **0.9575** | **0.8891** | **0.9235** | **0.9330** | **1.0256** | **1.0361** |
| **1.0711** | **1.0285** | **1.0523** | **0.9369** | **0.8966** | **0.9279** | **1.0484** | **1.0265** | **1.0439** | **1.0642** | **0.9854** | **1.1071** | **1.0429** | **1.0026** | **1.1094** |
| **1.1084** | **1.0312** | **1.0459** | **1.0793** | **1.2031** | **0.9313** | **1.0565** | **1.0454** | **1.0625** | **1.0025** | **0.9457** | **1.0778** | **1.0592** | **1.0131** | **1.0433** |
| **1.0272** | **1.0898** | **1.0734** | **0.8428** | **0.7949** | **0.8640** | **0.9490** | **1.0319** | **0.9322** | **1.0026** | **0.8156** | **0.9323** | **1.0243** | **0.9532** | **0.9359** |
| **0.9276** | **0.9245** | **0.9700** | **0.9996** | **0.8457** | **0.8220** | **0.8792** | **0.8450** | **0.8548** | **0.9526** | **0.9435** | **0.9462** | **0.9020** | **0.9052** | **0.9353** |
| **0.8732** | **0.8301** | **0.9255** | **0.5659** | **0.4014** | **0.5478** | **0.7297** | **0.7254** | **0.6130** | **0.7924** | **0.5628** | **0.8243** | **0.9475** | **0.9413** | **0.9275** |
| **1.1613** | **1.1840** | **1.1490** | **1.4688** | **1.6717** | **1.4467** | **1.0987** | **1.2252** | **1.1832** | **1.1232** | **1.2492** | **1.1331** | **1.1717** | **1.1858** | **1.2082** |
| **1.1570** | **1.0305** | **1.1032** | **1.1604** | **1.1250** | **1.1841** | **1.0702** | **1.1447** | **1.1411** | **1.0521** | **0.9438** | **0.9940** | **1.0662** | **1.0452** | **1.0185** |
| **1.0528** | **1.1057** | **1.0027** | **0.9682** | **0.8978** | **1.0446** | **1.0185** | **1.0546** | **0.9973** | **1.0388** | **1.0702** | **0.9863** | **0.9291** | **0.8805** | **1.0132** |
| **0.9651** | **0.9695** | **0.9972** | **0.9700** | **0.9558** | **0.7440** | **0.9449** | **0.9168** | **0.9056** | **0.9039** | **0.9490** | **0.9444** | **0.9826** | **0.9295** | **0.9300** |
| **1.0903** | **1.0357** | **1.0223** | **0.9971** | **0.9335** | **1.1313** | **1.0864** | **1.0294** | **1.0870** | **1.0617** | **1.0572** | **0.9624** | **1.0159** | **1.0090** | **0.9458** |
| **1.1087** | **1.0377** | **1.0329** | **1.0914** | **0.9729** | **1.0563** | **1.0659** | **1.0399** | **1.0385** | **0.9974** | **0.9941** | **1.0214** | **1.0156** | **1.0768** | **1.1588** |
| **1.0711** | **1.0285** | **1.0523** | **0.9369** | **0.8966** | **0.9279** | **1.0484** | **1.0265** | **1.0439** | **1.0642** | **0.9854** | **1.1071** | **1.0429** | **1.0026** | **1.1094** |
| **1.1084** | **1.0312** | **1.0459** | **1.0793** | **1.2031** | **0.9313** | **1.0565** | **1.0454** | **1.0625** | **1.0025** | **0.9457** | **1.0778** | **1.0592** | **1.0131** | **1.0433** |
| **1.0272** | **1.0898** | **1.0734** | **0.8428** | **0.7949** | **0.8640** | **0.9490** | **1.0319** | **0.9322** | **1.0026** | **0.8156** | **0.9323** | **1.0243** | **0.9532** | **0.9359** |
| **0.9276** | **0.9245** | **0.9700** | **0.9996** | **0.8457** | **0.8220** | **0.8792** | **0.8450** | **0.8548** | **0.9526** | **0.9435** | **0.9462** | **0.9020** | **0.9052** | **0.9353** |
| **0.8732** | **0.8301** | **0.9255** | **0.5659** | **0.4014** | **0.5478** | **0.7297** | **0.7254** | **0.6130** | **0.7924** | **0.5628** | **0.8243** | **0.9475** | **0.9413** | **0.9275** |
| **1.1613** | **1.1840** | **1.1490** | **1.4688** | **1.6717** | **1.4467** | **1.0987** | **1.2252** | **1.1832** | **1.1232** | **1.2492** | **1.1331** | **1.1717** | **1.1858** | **1.2082** |

# References

[1] Chen, Li & He, Simai & Zhang, Shuzhong. (2011). Tight Bounds for Some Risk Measures with Applications to Robust Portfolio Selection. Operations Research. 59.

[2] Cornuéjols, Gérard & Tutuncu, Reha. (2007). Optimization Methods in Finance. Cambridge University Press, Cambridge, UK.

[3] Goldfarb, Donald & Iyengar, Garud. (2003). Robust Portfolio Selection Problems. Math. Oper. Res. 28. 1-38.

[4] Markowitz, H. (1952). Portfolio selection. Journal of Finance 7, 77–91.

[5] Zhu, Lei & Coleman, Thomas & li, Yuying. (2009). Min-Max robust and CVaR robust mean-variance portfolios. Journal of Risk. 11.

[6] Alexander, S. & Coleman, T.F. & li, Yuying. (2006). Minimizing CVaR and VaR for a Portfolio of Derivatives. Journal of Banking & Finance. 30.

[7] Rockafellar, R & Uryasev, Stan. (1999). Optimization of Conditional Value-At-Risk. The Journal of Risk. 2.

[8] Quaranta, Anna & Zaffaroni, Alberto. (2008). Robust optimization of conditional value at risk and portfolio selection. Journal of Banking & Finance. 32.

[9] Kang, Zhilin & Li, Xun & Li, Zhongfei & Zhu, Shushang. (2018). Data-driven robust mean-CVaR portfolio selection under distribution ambiguity. Quantitative Finance. 1-17.

[10] Yam, Sheung & Yang, Hailiang & Yuen, Fei. (2015). Optimal asset allocation: Risk and information uncertainty. European Journal of Operational Research. 251.

[11] Garlappi, Lorenzo & Uppal, Raman & Wang, Tan. (2007). Portfolio Selection with Parameter and Model Uncertainty: A Multi-Prior Approach. Review of Financial Studies. 20. 41-81.

[12] Sun, Yufei & Aw, Ee & Li, Bin & Teo, Kok & Sun, Jie. (2017). CVaR-based robust models for portfolio selection. Journal of Industrial & Management Optimization. 13. 1-11.

[13] Meghwani, Suraj & Thakur, Manoj. (2017). Multi-objective heuristic algorithms for practical portfolio optimization and rebalancing with transaction cost. Applied Soft Computing. 67.

[14] Best, Michael & Hlouskova, Jaroslava. (2005). An Algorithm for Portfolio Optimization with Transaction Costs. Management Science. 51. 1676-1688.

[15] Beraldi, Patriza & Violi, Antonio & Ferrara, Massimiliano & Ciancio, Claudio & Pansera, Bruno. (2019). Dealing with complex transaction costs in portfolio management. Annals of Operations Research.

[16] Muthuraman, Kumar & Kumar, Sunil. (2004). Multi-Dimensional Portfolio Optimization with Proportional Transaction Costs. Mathematical Finance. 16.

[17] Kolm, Petter & Tütüncü, Reha & Fabozzi, Frank. (2014). 60 Years of portfolio optimization: Practical challenges and current trends. European Journal of Operational Research. 234. 356–371.