공학석사학위논문

# An approximation scheme for the probability maximizing combinatorial optimization problem

## 확률최대화 조합최적화 문제에 대한 근사해법

2019 년 8 월

서울대학교 대학원
산업공학과

이 지 선

# Abstract

# An approximation scheme for the probability maximizing combinatorial optimization problem

Jisun Lee
Department of Industrial Engineering
The Graduate School
Seoul National University

In this thesis, we consider a variant of the deterministic combinatorial optimization problem (DCO) where there is uncertainty in the data, the probability maximizing combinatorial optimization problem (PCO). PCO is the problem of maximizing the probability of satisfying the capacity constraint, while guaranteeing the total profit of the selected subset is at least a given value. PCO is closely related to the chance-constrained combinatorial optimization problem (CCO), which is of the form that the objective function and the constraint function of PCO is switched. It search for a subset that maximizes the total profit while guaranteeing the probability of satisfying the capacity constraint is at least a given threshold. Thus, we discuss the relation between the two problems and analyse the complexities of the problems in special cases. In addition, we generate pseudo polynomial time exact algorithms of PCO and CCO that use an exact algorithm of a deterministic constrained combinatorial optimization problem. Further, we propose an approximation scheme of PCO that is fully polynomial time approximation scheme (FPTAS) in some special cases that are $\mathcal{NP}$-hard. An approximation scheme of CCO is also presented which was derived in the process of generating the approximation scheme of PCO.

**Keywords**: Combinatorial Optimization with uncertainty, Probability Maximizing Combinatorial Optimization Problem, Chance-constrained Combinatorial Optimization Problem, Bisection Procedure

**Student Number**: 2017-29219

# Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

## 1.1 Problem Description

In this study, we consider a probability maximizing combinatorial optimization problem (PCO). A (deterministic) combinatorial optimization problem (DCO) is defined with a finite set $N = \{1, \dots, n\}$, weights $a_j$ for $j \in N$, profits $p_j$ for $j \in N$, and a set $\mathcal{F}$ of feasible subsets of $N$. DCO is a problem of finding a feasible set $S \in \mathcal{F}$ such that the sum of weights $\sum_{j \in S} a_j$ is minimized while guaranteeing the total profit $\sum_{j \in S} p_j$ of at least $f$. Here, we assume that $p_j \in \mathbb{Z}$ for all $j \in N$ and let the set of the incidence vectors of $F \in \mathcal{F}$ be $X$. Then, by defining binary variable $x_j$ to be 1 if item $j \in S$ and otherwise 0 for each $j \in N$, we can formulate DCO as the following.

$$
\begin{aligned}
\text{(DCO)} \quad \text{minimize} \quad & \sum_{j \in N} a_j x_j \\
\text{subject to} \quad & \sum_{j \in N} p_j x_j \geq f, \\
& x \in X,
\end{aligned}
$$

where $X \subseteq \mathbb{B}^n$.

In PCO however, weights $a_j$ for $j \in N$ are assumed to be independent normal random variables, that is, $a_j \sim N(\mu_j, \sigma_j^2)$ with $\mu_j \in \mathbb{Z}$ for all $j \in N$. Then, the problem is to find a subset $S \in \mathcal{F}$ that maximizes the probability $Pr\left(\sum_{j\in S} a_j \leq b\right)$ with the sum of the profit $\sum_{j\in S} p_j$ not less than $f$, where $b$ is a given nonnegative integer. We assume that there exists at least one feasible set $S \in \mathcal{F}$ that satisfies $\sum_{j\in S} \mu_j \leq b$, which indicates that the optimal objective value is at least 0.5. By defining $x_j$ for $j \in N$ as same as that of DCO, we can represent PCO as follows.

$$
\begin{aligned}
\text{(PCO)} \quad \text{maximize} \quad & Pr\left(\sum_{j\in N} a_j x_j \leq b\right) \\
\text{subject to} \quad & \sum_{j\in N} p_j x_j \geq f, \qquad (1.1) \\
& x \in X.
\end{aligned}
$$

The knapsack constraint (1.1) may be one of the defining constraints of the feasible set $X$ of a PCO. However, in the above formulation, we separate a knapsack constraint from the defining constraints of the feasible set $X$ for the ease of later exposition. Note that constraint (1.1) may be redundant.

Closely related to PCO from a theoretical point of view is so-called the chance-constrained combinatorial optimization problem (CCO) which is defined as follows.

$$
\begin{aligned}
\text{(CCO)} \quad \text{maximize} \quad & \sum_{j\in N} p_j x_j \\
\text{subject to} \quad & Pr\left(\sum_{j\in N} a_j x_j \leq b\right) \geq \rho, \\
& x \in X,
\end{aligned}
$$

where $\rho$ is a given probability threshold which we assume $0.5 \leq \rho < 1$. CCO is the problem to find a subset $S \subseteq N$ that maximizes $\sum_{j \in S} p_j$, the total profit, while guaranteeing the probability of satisfying the knapsack constraint $\sum_{j \in S} a_j \leq b$ to be at least $\rho$.

There are well-known applications of PCO and CCO in reality, where we consider the variability of the data. First, consider a case of choosing the shortest path from the departure $s$ to the destination $t$. Since there are many uncertainties in each road such as traffic jam or signal, the time required varies. Let $d_j$ be the duration of each road $j \in N$ such that $d_j \sim N(\mu_j, \sigma_j^2)$, and need pay the fee $p_j$ to drive through each road $j \in N$. Suppose that we have the budget $f$ and $\mathcal{X}_{st}$ is the set of the incidence vectors of all the possible $s$-$t$ paths. Then, the problem of choosing the path with maximum probability of arriving $t$ in time $b$ can be formulated as the following.

$$
\begin{aligned}
\text{maximize} \quad & Pr\left(\sum_{j \in N} d_j x_j \leq b\right) \\
\text{subject to} \quad & \sum_{j \in N} p_j x_j \leq f, \\
& x \in \mathcal{X}_{st}.
\end{aligned}
$$

Since we have no assumption on the sign of $p_j$, $j \in N$, the constraint $\sum_{j \in N} p_j x_j \leq f$ can be formulated as $\sum_{j \in N} -p_j x_j \geq -f$, which makes the above problem as a special case of PCO.

In addition, there is an application of the multi-robot teaming introduced in Yang and Chakraborty (2018). There are a finite set of robots $R$ that each robot $r \in R$ has an uncertain travel distance $d_r \sim N(\mu_r, \sigma_r^2)$. Each robot $r$ costs $c_r$ and we want to choose a subset of $R$ that has the probability of covering the total distance $D$ at least the given threshold $\rho$ with the minimum cost. The problem can be formulated as

$$\begin{aligned}
\text{minimize} \quad & \sum_{r \in R} c_r x_r \\
\text{subject to} \quad & Pr\left(\sum_{r \in R} d_r x_r \geq D\right) \geq \rho, \\
& x \in \mathbb{B}^{|R|},
\end{aligned}$$

which is the form of an instance of CCO when we substitute $x_r$ with $y_r$ where $y_r = 1 - x_r$. There are much more applications of PCO and CCO as we specify $X$ and the coefficients of the problems.

Now, we examine the relation between PCO and CCO. Clearly, it can be checked if the optimal objective value of PCO is at least a given threshold $\rho$ by solving CCO. Conversely, if the optimal value of PCO is at least $\rho$, then it means the optimal value of CCO is at least $f$. According to this relation, we can use one of them to solve the other. First, we can solve CCO by solving PCO polynomial times without any assumption. Let the lower and upper bound of the objective value of CCO be $L$ and $U$, respectively. A possible value of the lower and upper bound are $L = \min_{j \in N}\{p_j\}$ and $U = \sum_{j \in N} |p_j|$, respectively. Then we iteratively solve PCO with $f = \lceil \frac{L+U}{2} \rceil$. Then,

- if the optimal value is strictly smaller than $\rho$, reset as $U = \lceil \frac{L+U}{2} \rceil - 1$,
- otherwise, reset as $L = \lceil \frac{L+U}{2} \rceil$.

We continue solving PCO with updated value $f$, until we have $L = U$. Since we assumed that $p_j$ for all $j \in N$ are integer, it is sufficient to conclude and the optimal objective value is $L(=U)$. The number of iterations solving PCO is $\mathcal{O}(\log_2(U-L))$, which is polynomial of $n$.

We now consider the opposite direction, obtaining the optimal value of PCO by iteratively solving CCO. Unfortunately, since $Pr\left(\sum_{j\in S} a_j \leq b\right)$ can have any real value between $[0.5, 1]$, we need an assumption to solve PCO by solving CCO polynomial times. We assume that we know a lower bound $\delta > 0$ for the gap between any two possible value of $\frac{b-\sum_{j\in S}\mu_j}{\sqrt{\sum_{j\in S}\sigma_j^2}}$ that is larger than 0;

$$\delta \leq \left| \frac{b - \sum_{j\in S}\mu_j}{\sqrt{\sum_{j\in S}\sigma_j^2}} - \frac{b - \sum_{j\in T}\mu_j}{\sqrt{\sum_{j\in T}\sigma_j^2}} \right| \tag{1.2}$$

for all $S, T \subseteq N$ such that

$$\frac{b - \sum_{j\in S}\mu_j}{\sqrt{\sum_{j\in S}\sigma_j^2}} \neq \frac{b - \sum_{j\in T}\mu_j}{\sqrt{\sum_{j\in T}\sigma_j^2}}.$$

Then, similar to how we used PCO to solve CCO, we solve CCO with $\rho = \frac{L+U}{2}$, where $L$ and $U$ are a lower and upper bound of the objective value of PCO. Possible values are $L = 0.5$ and $U = 1$, since we have assumed that there exists at least one feasible subset $S \in \mathcal{F}$ that satisfies $\sum_{j\in S}\mu_j \leq b$. Then,

- if the optimal value is strictly smaller than $f$, reset as $U = \frac{L+U}{2}$,
- otherwise, reset as $L = \frac{L+U}{2}$.

We solve CCO with updated $L$ and $U$ until we have $U - L < \delta$. Then we obtain an optimal subset $S$ with $L \leq Pr\left(\sum_{j\in S} a_j \leq b\right)$. To solve PCO, we have to solve CCO $\mathcal{O}(\log_2(\frac{1}{\delta}))$ times. However, the assumption we made may be unrealistic, since there is no algorithm to find such $\delta$ that satisfies (1.2) that is known so far, except to check for all possible subset $S \subseteq N$ and find the minimum positive gap, which takes $\mathcal{O}(2^n)$ computational time.

## 1.2 Literature Review

Combinatorial optimization problems and the algorithms to solve them have been widely studied. Well-known combination optimization problems include knapsack problem, shortest path problem, vehicle routing problem, etc. The knapsack problem and the vehicle routing problem are $\mathcal{NP}$-hard (Pisinger and Toth, 1998; Toth and Vigo, 2002), and the shortest path problem is polynomial time solvable (Dijkstra, 1959). However, when there is uncertainty in the data, problems may be harder than the corresponding deterministic combinatorial problems. Thus, various studies have been done about the combinatorial optimization problems with uncertainty, recently.

From the perspective of the stochastic optimization, there are several popular models that have been considered. Among the models, using the probability of satisfying the capacity constraint with coefficients that follow independent normal distributions are one of the mainly studied models. The probability can be used as the objective function or it is possible to construct a constraint with the probability. The problems considered in this thesis are PCO and CCO as defined in Section 1.1. The complexity studies for the special cases of PCO and CCO have been done. Atamtürk and Narayanan (2009) proved that the submodular function minimization with a cardinality constraint can be solved in a polynomial time. This induces that CCO with unit profit values can be solved in a polynomial time. Additionally, Atamtürk et al. (2013) proved that the minimization of the mean-risk function, which is the function of the capacity constraint in the equivalent deterministic nonlinear form of CCO, over the generalized upper bound (GUB) constraints is $\mathcal{NP}$-hard. This implies the

$\mathcal{NP}$-hardness of CCO over GUB constraints. Nikolova et al. (2006) gave an upper-bound for both PCO and CCO in the case of the shortest path problem as $n^{\mathcal{O}(\log n)}$ by showing the one-to-one correspondence of the extreme points of the shadow of the path polytope dominant and the breakpoints of the parametric shortest path problem. Additionally, they suggested an exact algorithm for the stochastic shortest path problem with the complexity of $n^{\mathcal{O}(\log n)}$.

Many researches on the algorithm of PCO and CCO for general and some special cases also have been actively proceeded. For the stochastic approach of the combinatorial optimization problem with uncertainty, Nikolova (2008) presented two different stochastic optimization problems and their equivalent deterministic nonconvex forms, the threshold and the risk stochastic problem. She proposed approximation schemes of the two problems with an application of the stochastic shortest path problem in Nikolova (2009). In particular, the two problems covered in Nikolova (2008) and Nikolova (2009) are closely related to PCO and CCO. The problems are

$$\begin{aligned} \text{(N-PCO)} \quad \text{maximize} \quad & \frac{b - \sum_{j \in N} \mu_j x_j}{\sqrt{\sum_{j \in N} \sigma_j^2 x_j}} \\ \text{subject to} \quad & x \in \mathcal{F}, \end{aligned}$$

and

$$\begin{aligned} \text{(N-CCO)} \quad \text{minimize} \quad & \sum_{j \in N} \mu_j x_j + \Phi^{-1}(\rho) \sqrt{\sum_{j \in N} \sigma_j^2 x_j} \\ \text{subject to} \quad & x \in \mathcal{F}, \end{aligned}$$

where $\mathcal{F} \subseteq \mathbb{R}^n$. The objective function of N-PCO is to maximize the tail probability, and that of N-CCO is to minimize the possible value of the sum of the mean and the deviation of the weight, without assuming any distribution to the uncertain data. Since the objective functions of N-PCO and N-CCO are different from those of PCO and CCO, the approximation solutions satisfy different conditions. The complexities of the approximation schemes of N-PCO and N-CCO proposed in Nikolova (2009) are

$$\mathcal{O}\left(\log(\frac{s_{\max}}{s_{\min}})\log(\frac{f_u}{f_l})\frac{1}{\epsilon^2}g(w,\mathcal{F})\right) \text{ and } \mathcal{O}\left((1+\frac{1}{\epsilon}\log(\frac{f_u}{f_l}))(1+\frac{\log(\frac{1}{\epsilon^2})}{\log(1+\epsilon)})g(w,\mathcal{F})\right),$$

respectively. Here, $s_{\max}$ and $s_{\min}$ correspond to the maximum and the minimum values of $\sum_{j\in N}\sigma_j^2 x_j$ for $x \in \mathcal{F}$, $f_u$ and $f_l$ are the values of the maximum and the minimum value of $\sum_{j\in N}\mu_j x_j + \Phi^{-1}\sqrt{\sum_{j\in N}\sigma_j^2 x_j}$ for $x \in \mathcal{F}$, and $g(w,\mathcal{F})$ is the computational time of the minimization of the linear function $w^T x$ over $\mathcal{F}$. Other studies also considered PCO and CCO such as Ilyina (2017), which covered the complexity analysis and the solution approaches of the combinatorial optimization under ellipsoidal uncertainty. It studied both the uncorrelated and the correlated cases.

The most common special case of the stochastic combinatorial optimization is the stochastic knapsack problem. Goyal and Ravi (2010) proposed a polynomial time approximation scheme (PTAS) for the chance-constrained knapsack problem with the uncertainty in the item size. Han et al. (2016) also proposed an approximation algorithm of the chance-constrained knapsack problem by approximating the ellipsoidal uncertainty set with a polyhedral set. Klopfenstein and Nace (2008) suggested an robust model of the chance-constrained knapsack problem and devised an approximation algorithm of the chance-constrained knapsack problem that solves the robust model iteratively. Shabtai et al. (2018) proposed a relaxed fully polynomial time approximation scheme (FPTAS) for the chance-constrained knapsack problem of which the item weights follow normal distributions. Here, the relaxed approximation means that the probability of satisfying the capacity constraint is $(1-\epsilon)$ times the given threshold. A heuristic algorithm for the chance-constrained knapsack problem has been presented by Joung and Lee (2018), where they used the submodularity of the mean-risk function.

There are also some algorithmic studies for the special case of the shortest path problem. Ji (2005) proposed three stochastic models for the shortest path problem, which are the expected shortest path model, the most shortest path model, $\alpha$-shortest path model, and suggested a hybrid intelligent algorithm which consist of genetic algorithm and stochastic simulation. Cheng and Lisser (2015) generated an approximation algorithm of the maximization of the probability of the stochastic resource constrained shortest path problem, which maximizes the probability of satisfying all the resources constraints while not exceeding the cost threshold. They used a second-order cone programming approximation to solve the relaxed problem repeatedly. Additionally, Dinh et al. (2018) suggested an exact algorithm that solves the chance-constrained vehicle routing problem with uncertainties in the demand.

Even though there are many studies that take uncertainty into account, there are still areas that need to be studied. We could not find any previous study that suggests an approximation scheme guaranteeing an absolute error of the probability value by the approach of using the probability function itself. Also, complexity studies for special cases of PCO and CCO are not much. Thus in this thesis, we focus on the stochastic model with uncertain values that follow normal distributions. We analyse the complexities of PCO and CCO, and generate solution approaches to solve those problems efficiently.

## 1.3 Research Motivation and Contribution

The research motivations and the main contributions of our thesis are as follows:

(a) We conducted the complexity study of PCO and CCO in the general and the special cases, and discovered that some special cases of PCO and CCO can be $\mathcal{NP}$-hard even though the deterministic combinatorial optimization problems (DCO) in the same condition are polynomial time solvable.

(b) We proposed pseudo polynomial time exact algorithms of PCO and CCO that iteratively solve constrained DCOs.

(c) We devised an approximation scheme of PCO which guarantees that the absolute error of the probability value of the resulting solution is at most $\delta$ to the optimal probability value of PCO.

(d) An approximation scheme of CCO is suggested, which guarantees the solution that the probability of satisfying the capacity constraint is at least $\rho - \delta$ for any given $\rho \in [0.5, 1]$.

## 1.4 Organization of the Thesis

This thesis is composed of 5 chapters. In Chapter 2, we analyse the complexities of PCO and CCO in the general and the special cases. In Chapter 3, we propose exact algorithms to solve PCO and CCO, which follow similar procedures of iteratively solving deterministic combinatorial problems. Then, in Chapter 4, we suggest an approximation scheme of PCO that uses an approximation scheme of CCO, which solves DCO repeatedly. Finally, in Chapter 5, the concluding remarks and future works of this study are presented.

# Chapter 2

# Computational Complexity of Probability Maximizing Combinatorial Optimization Problem

In this chapter, we study the complexity of the probability maximizing combinatorial optimization problem (PCO) and the chance-constrained combinatorial optimization problem (CCO) for some special cases, given in the Table 2.1.

Table 2.1: Special cases of PCO and CCO

| Case | $X$ | Profit value |
|---|---|---|
| 1 | $\mathbb{B}^n$ | $p_j \in \mathbb{Z}, \forall j \in N$ |
| 2 | | $p_j = 1, \forall j \in N$ |
| 3 | | $p_j \sim \mathcal{O}(p(n))$,[a] $\forall j \in N$ |
| 4 | $\mathcal{X}_{st}$[b] | $p_j = 0\ (\geq f), \forall j \in A$ |
| 5 | | $p_j \in \mathbb{Z}_-, \forall j \in A$ |
| 6 | | $p_j = -1\ (\geq f), \forall j \in A$ |

[a] $p(n)$ : polynomial function of $n$
[b] $\mathcal{X}_{st}$ : set of incidence vectors of $s$-$t$ paths for a given graph $G = (V, A)$ with $|V| = n$, $|A| = m$

Beforehand, consider the complexity of the deterministic combinatorial optimization problem (DCO) for the six cases. We formulate DCO as

$$
\begin{aligned}
\text{(DCO)} \quad \text{minimize} \quad & \sum_{j \in N} w_j x_j \\
\text{subject to} \quad & \sum_{j \in N} p_j x_j \geq f, \\
& x \in X,
\end{aligned}
\tag{2.1}
$$

where the coefficients $w_j \in \mathbb{R}$ and $p_j \in \mathbb{Z}$ for $j \in N$. It is obvious that DCO is a 0-1 knapsack problem when $X = \mathbb{B}^n$ and is a constrained shortest path problem when $X = \mathcal{X}_{st}$. For the cases of $X = \mathcal{X}_{st}$, assume that we are given a directed graph $G$ of which the cycle set $\mathcal{C}$ of $G$ satisfies $\sum_{j \in C} w_j x_j \geq 0$ for all $C \in \mathcal{C}$ i.e., all cycles are nonnegative cycles. Then, the complexity of each cases are given as the Table 2.2. (See, e.g., Kellerer et al. (2004) for $X = \mathbb{B}^n$, and Bellman (1958) for $X = \mathcal{X}_{st}$.)

Table 2.2: Complexity of DCO for special cases

| Case | $X$ | Profit value | Complexity | Algorithm Complexity |
|---|---|---|---|---|
| 1 | $\mathbb{B}^n$ | $p_j \in \mathbb{Z}, \forall j \in N$ | $\mathcal{NP}$-hard | $\mathcal{O}(nU)$[a] |
| 2 | | $p_j = 1, \forall j \in N$ | $\mathcal{P}$ | $\mathcal{O}(n^2)$ |
| 3 | | $p_j \sim \mathcal{O}(p(n))$,[b] $\forall j \in N$ | $\mathcal{P}$ | $\mathcal{O}(n^2 p(n))$ |
| 4 | $\mathcal{X}_{st}$[c] | $p_j = 0 \ (\geq f), \forall j \in A$ | $\mathcal{P}$ | $\mathcal{O}(nm)$[d] |
| 5 | | $p_j \in \mathbb{Z}_-, \forall j \in A$ | $\mathcal{NP}$-hard | $\mathcal{O}(nmU)$ |
| 6 | | $p_j = -1 \ (\geq f), \forall j \in A$ | $\mathcal{P}$ | $\mathcal{O}(nm^2)$ |

[a] $U$ : upper bound of $\sum_{j \in N} |p_j| x_j$, $x \in X$

[b] $p(n)$ : polynomial function of $n$

[c] $\mathcal{X}_{st}$ : set of incidence vectors of $s$-$t$ paths for a given graph $G = (V, A)$ with $|V| = n$, $|A| = m$

Consider a case of PCO with $\sigma_j = 0$ and $\mu_j = w_j$ for all $j \in N$. Then the problem turns out to be a decision problem

$$\begin{aligned} \text{Instance :} \quad & N = \{1, \dots, n\}, X \subseteq \mathbb{B}^n,\ b, f \in \mathbb{Z},\ p_j \in \mathbb{Z} \text{ and } w_j \in \mathbb{R}, \forall j \in N \\ \text{Question :} \quad & \exists x \in X \text{ such that } \sum_{j \in N} w_j x_j \leq b,\ \sum_{j \in N} p_j x_j \geq f\ ? \end{aligned} \tag{2.2}$$

The optimal value of PCO is 1 if the answer of (2.2) is "yes" and 0, otherwise. In addition, (2.2) is also the decision problem of DCO and can be answered by solving a single DCO. In this chapter, the main goal is to determine the complexity of PCO in special cases in Table 2.1. Thus, in the rest of the chapter, we will first analyse the complexity of CCO for the above six cases, and then use some of the proof to analyse the complexity of PCO for the cases.

## 2.1 Complexity of General Case of PCO and CCO

We first analyse the complexity of PCO and CCO in general case. Since the two problems have equivalent decision problem, the complexities of both problems can be determined by using the same decision problem. Consider the following decision problem

$$\begin{aligned} \text{Instance :}\quad & N = \{1, \ldots, n\}, X \subseteq \mathbb{B}^n,\ b, f \in \mathbb{Z},\ p_j \in \mathbb{Z}, \forall j \in N, \\ & a_j \sim N(\mu_j, \sigma_j^2) \text{ where } \mu_j \in \mathbb{Z}, \sigma_j \in \mathbb{R}, \forall j \in N \\ \text{Question :}\quad & \exists x \in X \text{ such that } \sum_{j \in N} p_j x_j \geq f,\ Pr\left(\sum_{j \in N} a_j x_j \leq b\right) \geq \rho\ ? \end{aligned} \tag{2.3}$$

which is the decision problem of both PCO and CCO.

**Proposition 2.1.** *If (2.3) is $\mathcal{NP}$-complete, then both PCO and CCO is $\mathcal{NP}$-hard.*

*Proof.* Assume the contrary, i.e., PCO or CCO is not $\mathcal{NP}$-hard when (2.3) is $\mathcal{NP}$-complete. Without loss of generality, suppose that CCO is polynomial time solvable. Then for any instance of (2.3), we can answer the question by solving CCO with same instance of $p_j$, $\mu_j$, $\sigma_j$ for all $j \in N$ and $\rho$, and comparing the objective value to $f$. Thus, the assumption is wrong and CCO needs to be $\mathcal{NP}$-hard. Same logic can be applied to PCO. □

Now consider a special case of (2.3). If $X = \mathbb{B}^n$ and $\sigma_j = 0$ for all $j \in N$, (2.3) becomes a decision problem of the knapsack problem which is $\mathcal{NP}$-complete (Kellerer et al., 2004). Thus, (2.3) is $\mathcal{NP}$-complete in general and by Proposition 2.1, PCO and CCO are both $\mathcal{NP}$-hard.

## 2.2 Complexity of CCO in Special Cases

In this section, we consider the complexity of the chance-constrained combinatorial optimization problem of the special cases in Table 2.1.

$$\begin{aligned}
\text{(CCO)} \quad \text{maximize} \quad & \sum_{j \in N} p_j x_j \\
\text{subject to} \quad & Pr\left(\sum_{j \in N} a_j x_j \leq b\right) \geq \rho, \qquad (2.4)\\
& x \in X,
\end{aligned}$$

where $a_j \sim N(\mu_j, \sigma_j^2)$ for $j \in N$ and $X \subseteq \mathbb{B}^n$. Throughout this paper, we assume that the values of $p_j$ and $\mu_j$ are both integer for all $j \in N$.

Prior to the analysis of the complexity, we reformulate CCO to an equivalent deterministic problem by using the cumulative distribution function $\Phi$ of the standard normal distribution. The constraint (2.4) can be reformulated as

$$Pr\left( z \leq \frac{b - \sum_{j \in N} \mu_j x_j}{\sqrt{\sum_{j \in N} \sigma_j^2 x_j^2}} \right) \geq \rho, \qquad (2.5)$$

where $z = \frac{\sum_{j \in N} a_j x_j - \sum_{j \in N} \mu_j x_j}{\sqrt{\sum_{j \in N} \sigma_j^2 x_j^2}}$ and $z \sim N(0, 1^2)$. Since $\Phi(\cdot)$ is nondecreasing, (2.5) can be replaced by

$$\sum_{j \in N} \mu_j x_j + \Phi^{-1}(\rho) \sqrt{\sum_{j \in N} \sigma_j^2 x_j^2} \leq b, \qquad (2.6)$$

where $\Phi^{-1}$ is the inverse cumulative distribution function of standard normal distribution. Additionally, $x_j^2$ can be converted to $x_j$ because $x \in \{0,1\}$ for all $j \in N$. Then, we can reformulate CCO as

$$\begin{aligned} \text{(CCO)} \quad \text{maximize} \quad & \sum_{j\in N} p_j x_j \\ \text{subject to} \quad & \sum_{j\in N} \mu_j x_j + \Phi^{-1}(\rho)\sqrt{\sum_{j\in N} \sigma_j^2 x_j} \le b, \\ & x \in X. \end{aligned} \tag{2.7}$$

Either of the formulation of CCO with the probability lower bound constraint (2.4) or that with nonlinear deterministic inequality constraint (2.7) can be used to analyse the complexity of CCO. Note that we call the first three cases of $X = \mathbb{B}^n$ as the chance-constrained knapsack problem (CKP) and the other three cases of $X = \mathcal{X}_{st}$ as the chance-constrained shortest path problem (CSP).

▷ **Case 1 :** $X = \mathbb{B}^n$**,** $p_j \in \mathbb{Z}$ **for** $j \in N$

Consider an instance of $\sigma_j = 0$ for all $j \in N$, then the second term of the constraint (2.7) can be removed and the problem turns out to be as

$$\begin{aligned} \text{maximize} \quad & \sum_{j\in N} p_j x_j \\ \text{subject to} \quad & \sum_{j\in N} \mu_j x_j \le b, \\ & x \in \mathbb{B}^n, \end{aligned}$$

which is a deterministic 0-1 knapsack problem. This problem is well-known to be $\mathcal{NP}$-hard (Kellerer et al., 2004) and thus, Case 1 with $X = \mathbb{B}^n$ and $p_j \in \mathbb{Z}$ for all $j \in N$ is also $\mathcal{NP}$-hard.

$\triangleright$ **Case 2 : $X = \mathbb{B}^n$, $p_j = 1$ for $j \in N$**

Before we cover the case of unit profit values, i.e., $p_j = 1$ for all $j \in N$, we first introduce a new problem

$$
\begin{aligned}
\text{(Sub-CCO)} \quad \text{minimize} \quad & \sum_{j \in N} \mu_j x_j + \Phi^{-1}(\rho) \sqrt{\sum_{j \in N} \sigma_j^2 x_j} \\
\text{subject to} \quad & \sum_{j \in N} p_j x_j \geq f, \\
& x \in X,
\end{aligned}
$$

which can be obtained by switching the objective function and the constraint (2.7) of the deterministic nonlinear formulation of CCO.

**Proposition 2.2.** *If Sub-CCO is polynomial time solvable, then CCO is also polynomial time solvable.*

*Proof.* Recall the assumption that the profit values $p_j$, $\forall j \in N$, are integer. Thus, the possible candidate values of the objective function of CCO are $0, 1, \ldots, \sum_{j \in N} p_j$. Then, we can obtain an optimal solution of CCO by solving Sub-CCO with $f = \sum_{j \in N} p_j, \sum_{j \in N} p_j - 1, \ldots, 1$ in decreasing order, until we have a solution that has the objective value less than $b$. This cost us the number of the iterations of solving Sub-CCO at most $\sum_{j \in N} p_j$ times. However, we can reduce the number of iterations by doing bisection of $f$ in the range $\left[1, \sum_{j \in N} p_j\right]$.

First, initialize as $LB = 1$ and $UB = \sum_{j \in N} p_j$. In each iteration, solve Sub-CCO with the value $f = \lceil \frac{LB+UB}{2} \rceil$. Then,

- if the objective value is strictly larger than $b$, reset as $UB = \lceil \frac{LB+UB}{2} \rceil - 1$,
- otherwise, reset as $LB = \lceil \frac{LB+UB}{2} \rceil$.

We continue the iteration until we have $LB = UB$, which takes $\mathcal{O}\left(\log_2(\sum_{j \in N} p_j)\right)$ iterations. Thus even if $p_j$ for some $j \in N$ are exponential of $n$, we can guarantee that it is possible to obtain an optimal solution of CCO by solving Sub-CCO polynomial times. This induces to the relation that if Sub-CCO is polynomially solvable, then CCO is also polynomially solvable. □

Now, if we show that a special case of Sub-CCO is polynomial-time solvable, we can guarantee that CCO is also polynomial-time solvable in same case. When $X = \mathbb{B}$ and $p_j = 1$ for all $j \in N$, Sub-CCO is proven to be polynomial-time solvable in Atamtürk and Narayanan (2009). Thus, CCO is also polynomial-time solvable in the same condition.

$\triangleright$ **Case 3 : $X = \mathbb{B}^n$, $p_j \sim \mathcal{O}(p(n))$ for $j \in N$**

Next, consider the case of $p_j$ values that are polynomially bounded by $n$, i.e., $p_j \sim \mathcal{O}(p(n))$, where $p(n)$ is a polynomially bounded function of $n$. We do not yet know whether CKP in this case is $\mathcal{NP}$-hard or not. However, we propose a pseudo-polynomial time dynamic programming algorithm for the problem. The algorithm is given in Section 3.2 and for CKP, we can solve the problem for all $f = 1, \dots, U = \sum_{j \in N} |p_j|$ and $t = 1, \dots, b$ at once using dynamic programming. This reduces the complexity of the algorithm for CKP to $\mathcal{O}(nbU)$.

▷ **Case 4-6 :** $X = \mathcal{X}_{st}$

Now, let $X = \mathcal{X}_{st}$, which is the set of the incidence vectors of $s$-$t$ simple paths for a given directed graph $G = (V, A)$ with the departure node $s$ and the destination node $t$, where $s, t \in V$. For each arc $j \in A$, arc weights are given as $a_j \sim N(\mu_j, \sigma_j^2)$ and the resource consumption $p_j$, $\forall j \in A$ with no negative cycle. Note that $G$ does not need to be simple graph. For all Cases 4,5, and 6, we can show that CSP is $\mathcal{NP}$-hard by proving only for the Case 4, i.e., all zero profits $p_j = 0$ for all $j \in A$. The corresponding decision problem of the Case 4 is as follows.

$$\begin{aligned} \text{Instance :} \quad & G = (V, A) \text{ with } |V| = n,\ |A| = m,\ \mathcal{X}_{st} \subseteq \mathbb{B}^m,\ b \in \mathbb{Z},\ \rho \in [0.5, 1], \\ & a_j \sim N(\mu_j, \sigma_j^2) \text{ where } \mu_j \in \mathbb{Z}, \sigma_j \in \mathbb{R}, \forall j \in N \\ \text{Question :} \quad & \exists x \in \mathcal{X}_{st} \text{ such that } \sum_{j \in A} \mu_j x_j + \Phi^{-1}(\rho) \sqrt{\sum_{j \in A} \sigma_j^2 x_j} \le b\ ? \end{aligned} \tag{2.8}$$

We prove that the above decision problem is $\mathcal{NP}$-complete by showing a special case of $G$ that makes (2.8) $\mathcal{NP}$-complete.

**Proposition 2.3.** *The decision problem (2.8) is $\mathcal{NP}$-complete.*

*Proof.* Consider a directed graph $G = (V, A)$ with $V = \{0, 1, \ldots, q\}$ and set of arcs $A$ that consists of arcs $k$, $q+k$ that go from node $k-1$ to node $k$ for all $k = 1, \ldots, q$.

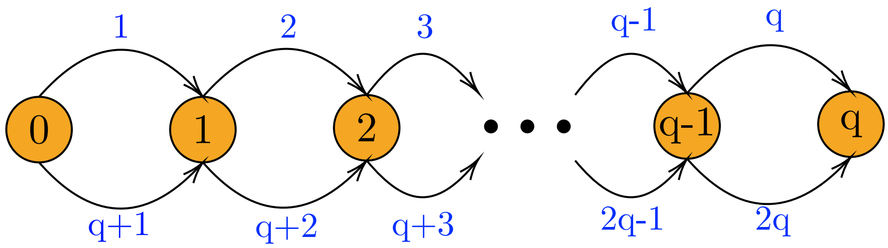


Figure 2.1: Example graph for the proof of $\mathcal{NP}$-hardness

For convenience, index arcs $k \in A = \{1, \ldots, 2q\}$ as in Figure 2.1 and suppose that each arc has a length $a_k \sim N(\mu_k, \sigma_k^2)$ such that $\mu_k \in \mathbb{Z}$ and $\sigma_k \in \mathbb{R}$ for all $k \in A$. The problem is to answer whether there is a path from node 0 to node $q$ that satisfies $\sum_{k \in A} \mu_k x_k + \Phi^{-1}(\rho)\sqrt{\sum_{k \in A} \sigma_k^2 x_k} \leq b$. We can state the decision problem (2.8) for the graph $G$ in Figure 2.1 as the following.

$$
\begin{aligned}
\text{Instance :} \quad & G = (V, A) \text{ with } |V| = q+1,\ |A| = 2q,\ b \in \mathbb{Z},\ \rho \in [0.5, 1], \\
& a_k \sim N(\mu_k, \sigma_k^2) \text{ where } \mu_k \in \mathbb{Z}, \sigma_k \in \mathbb{R}, \forall k \in A \\
\text{Question :} \quad & \exists x \text{ such that } \sum_{k \in A} \mu_k x_k + \Phi^{-1}(\rho)\sqrt{\sum_{k \in A} \sigma_k^2 x_k} \leq b, \\
& x_k + x_{k+q} = 1,\ \forall k = 1, \ldots, q, \quad x_k \in \{0, 1\},\ \forall k = 1, \ldots, 2q \ ?
\end{aligned}
\tag{2.9}
$$

Define a lower bound $l$ and an upper bound $u$ as

$$
\begin{aligned}
l &= \sum_{k \in Q} \min\{\mu_k, \mu_{q+k}\}, \\
u &= \sum_{k \in Q} \max\{\mu_k, \mu_{q+k}\} + \Phi^{-1}\sqrt{\sum_{k \in Q} \max\{\sigma_k^2, \sigma_{q+k}^2\}}
\end{aligned}
$$

of the possible values of

$$
\sum_{k \in Q} \mu_k x_k + \Phi^{-1}(\rho)\sqrt{\sum_{k \in Q} \sigma_k^2 x_k},
$$

where $Q = \{1, \ldots, q\}$ and let $D = u - l$. Then, define another decision problem as

$$\begin{aligned}
&\sum_{k \in Q} \mu_k x_k + \Phi^{-1}(\rho) \sqrt{\sum_{k \in Q} \sigma_k^2 x_k} + s = b, \\
&x_k + x_{q+k} = 1, \forall k \in \{1, \ldots, q\}, \\
&x \in \mathbb{B}^{2q}, \\
&0 \le s \le D,
\end{aligned} \tag{2.10}$$

where $s$ is a slack variable and since $u - l = D$, $s$ can be any real value between 0 and $D$. We can prove that if (2.10) is $\mathcal{NP}$-complete, (2.9) is also $\mathcal{NP}$-complete. For the decision problem (2.10), the answer depends on the value of $b$.

- If $b < l$ : answer "no" (no item chosen with $s = b$).
- If $b > u = l + D$ : answer "yes" (there exists a solution such that $\exists j \in N$ with $x_j = 1$ since $\sum_{k \in Q} \mu_k x_k + \Phi^{-1}(\rho) \sqrt{\sum_{k \in Q} \sigma_k^2 x_k} > l$).
- If $l \le b \le u$ : the answer is same as the answer of (2.9).

Thus, when (2.10) is $\mathcal{NP}$-complete, it corresponds to the case of $l \le b \le u$ implying that (2.9) is also $\mathcal{NP}$-complete.

Now, it is sufficient to prove that (2.10) is $\mathcal{NP}$-complete. Here we prove it using the Two-Partition Problem which is $\mathcal{NP}$-complete (Karp, 1972).

**Definition 2.4.** *Two-Partition Problem : Given a set of positive integers $W = \{w_1, \ldots, w_q\}$, is it possible to construct two sets $W_1$ and $W_2$ that have equal values without an intersection, i.e., $\sum_{k \in W_1} w_k = \sum_{k \in W_2} w_k = \frac{1}{2} \sum_{k \in W} w_k = C$, $W_1 \cap W_2 = \emptyset$?*

The two-partition problem can be reduced to (2.10) by setting the instance values of (2.10) as the following. Let $\mu_k := 2Dw_k$ and $\sigma_k := 0$ for $k = 1, \ldots, q$, and $\mu_{q+k} := 0$ and $\sigma_{q+k}^2 := w_k$ for $k = 1, \ldots, q$. Then by assigning $b := 2DC + \sqrt{C}$ and $\Phi^{-1}(\rho) := 1$, (2.10) turns out to be

$$\sum_{k=1}^{q} 2Dw_k x_k + \sqrt{\sum_{k=1}^{q} w_k x_{q+k}} + s = 2DC + \sqrt{C},$$
$$x_k + x_{q+k} = 1, \forall k \in \{1, \dots, q\},$$
$$x \in \mathbb{B}^{2q},$$
$$0 \le s \le D.$$

This problem is proven to be $\mathcal{NP}$-complete in Atamtürk et al. (2013). Therefore, (2.10) is $\mathcal{NP}$-complete and thus, the decision problem (2.9) is $\mathcal{NP}$-complete which is an instance of Case 4. Consequently, (2.8) is $\mathcal{NP}$-complete. □

Even though, the $\mathcal{NP}$-hardness of the PCO and CCO with $X = \mathcal{X}_{st}$ may be inferred by the $\mathcal{NP}$-hardness of constrained shortest path problem (Warburton, 1987), the above proof implies that it is still $\mathcal{NP}$-hard for a very simple graph like Figure 2.1. This naturally leads to the $\mathcal{NP}$-hardness of Case 5 and 6, since the Case 4 is a special case of both Case 5 and 6. Thus, CSP is $\mathcal{NP}$-hard with any profit values $p_{ij}$ for all $(i, j) \in A$.

We summarize the complexity of CCO for the six special cases as Table 2.3.

Table 2.3: Complexity of CCO for special cases

| Case | $X$ | Profit value | Complexity |
|---|---|---|---|
| 1 | $\mathbb{B}^n$ | $p_j \in \mathbb{Z}, \forall j \in N$ | $\mathcal{NP}$-hard |
| 2 | | $p_j = 1, \forall j \in N$ | $\mathcal{P}$ |
| 3 | | $p_j \sim \mathcal{O}(p(n))$[a], $\forall j \in N$ | at most $\mathcal{O}(nbU)$[b] |
| 4 | $\mathcal{X}_{st}$[c] | $p_j = 0\ (\ge f), \forall j \in A$ | $\mathcal{NP}$-hard |
| 5 | | $p_j \in \mathbb{Z}_-, \forall j \in A$ | $\mathcal{NP}$-hard |
| 6 | | $p_j = -1\ (\ge f), \forall j \in A$ | $\mathcal{NP}$-hard |

[a] $p(n)$ : polynomial function of $n$
[b] $U$ : upper bound of $\sum_{j \in N} |p_j| x_j$, $x \in X$
[c] $\mathcal{X}_{st}$ : set of incidence vectors of $s$-$t$ paths for a given graph $G = (V, A)$ with $|V| = n$, $|A| = m$

## 2.3 Complexity of PCO in Special Cases

Finally, consider PCO which can be formulated as

$$\text{(PCO)} \quad \text{maximize} \quad Pr\left(\sum_{j\in N} a_j x_j \le b\right) \tag{2.11}$$

$$\text{subject to} \quad \sum_{j\in N} p_j x_j \ge f,$$

$$x \in X,$$

where $a_j \sim N(\mu_j, \sigma_j^2)$ for $j \in N$, $p_j$ and $\mu_j$ are both integer for all $j \in N$ and $X \subseteq \mathbb{B}^n$. Before analysing the complexity, we reformulate PCO to an equivalent deterministic problem using the cumulative distribution function $\Phi$ of the standard normal distribution. The objective function (2.11) can be reformulated as

$$Pr\left( z \le \frac{b - \sum\limits_{j\in N} \mu_j x_j}{\sqrt{\sum\limits_{j\in N} \sigma_j^2 x_j^2}} \right),$$

where $z = \frac{\sum_{j\in N} a_j x_j - \sum_{j\in N} \mu_j x_j}{\sqrt{\sum_{j\in N} \sigma_j^2 x_j^2}}$ and $z \sim N(0, 1^2)$. Since $\Phi^{-1}$, the inverse function of $\Phi$, is a nondecreasing function of $\rho$, we can obtain an optimal solution of PCO by maximizing the following fractional function

$$\frac{b - \sum\limits_{j\in N} \mu_j x_j}{\sqrt{\sum\limits_{j\in N} \sigma_j^2 x_j^2}}.$$

We can substitute $x_j^2$ with $x_j$, since $x_j \in \{0,1\}$ for all $j \in N$. Thus, solving PCO is equivalent to solving the following problem

$$\begin{aligned}
\text{(PCO)} \quad \text{maximize} \quad & \frac{b - \sum\limits_{j\in N} \mu_j x_j}{\sqrt{\sum\limits_{j\in N} \sigma_j^2 x_j}} && (2.12)\\
\text{subject to} \quad & \sum_{j\in N} p_j x_j \geq f, \\
& x \in X.
\end{aligned}$$

We use either of the formulation with the objective function of the form (2.11) or (2.12) to analyse the complexity of PCO of the special cases.

**▷ Case 1,4-6 : CCO is $\mathcal{NP}$-hard**

Here, we first propose a proposition about the relation between PCO and CCO.

**Proposition 2.5.** *If CCO is $\mathcal{NP}$-hard in a certain condition, then PCO is also $\mathcal{NP}$-hard in the same condition.*

*Proof.* Recall the assumption that all values of $p_j$, $j \in N$ are integer. Then the objective function of CCO can only have integer values. Also remind that the formulation of PCO can be obtained by switching the objective function and the constraint (2.4) of CCO. Then, we can obtain an optimal solution of CCO by solving PCO with $f = \sum\limits_{j\in N} p_j, \sum\limits_{j\in N} p_j - 1, \ldots, 1$ in decreasing order, until we have a solution with the objective value larger than $\rho$. By applying a binary search of $f$ in range $\left[1, \sum\limits_{j\in N} p_j\right]$, same as the procedure in the proof of Proposition 2.2, the number of the iterations is at most $\mathcal{O}(\log_2(\sum_{j\in N} p_j))$. Thus, we can obtain an optimal solution of CCO by solving PCO polynomial times. Suppose that PCO is polynomial-time solvable.

Then, by solving PCO polynomial time, we obtain an optimal solution of CCO in polynomial time. Thus, as the contraposition, if CCO is $\mathcal{NP}$-hard, then PCO is also $\mathcal{NP}$-hard. $\square$

By Proposition 2.5, we can convince that PCO in the cases 1,4,5, and 6 is $\mathcal{NP}$-hard. We then only have to analyse the complexity of PCO for the case 2 and 3, which we call it probability maximizing knapsack problem (PKP) since $X = \mathbb{B}^n$.

$\triangleright$ **Case 2 : $X = \mathbb{B}^n$, $p_j = 1$ for $j \in N$**

Suppose that $p_j = 1$ for all $j \in N$. Then, we solve

$$\begin{aligned} \text{maximize} \quad & \frac{b - \sum_{j \in N} \mu_j x_j}{\sqrt{\sum_{j \in N} \sigma_j^2 x_j}} && (2.13) \\ \text{subject to} \quad & \sum_{j \in N} x_j \geq f, && (2.14) \\ & x \in \mathbb{B}^n. \end{aligned}$$

Since the objective function (2.13) is a nonincreasing function of variable $x_j$, $\forall j \in N$, we still obtain the same optimal objective value even if we change the constraint (2.14) to the equation form, that is

$$\begin{aligned} \text{maximize} \quad & \frac{b - \sum_{j \in N} \mu_j x_j}{\sqrt{\sum_{j \in N} \sigma_j^2 x_j}} && (2.15) \\ \text{subject to} \quad & \sum_{j \in N} x_j = f, \\ & x \in \mathbb{B}^n. \end{aligned}$$

Let $a(T) = \sum_{j \in T} \mu_j$ and $c(T) = \sum_{j \in T} \sigma_j^2$ for any subset $T \subseteq N$. Then, the above problem can be represented as the following problem

$$\nu_f = \max_{T \subseteq N} \left\{ \frac{b - a(T)}{\sqrt{(c(T))}} : |T| = f \right\}. \tag{2.16}$$

Let

$$\mathcal{T}_f = \{T \subseteq N : |T| = f\},$$
$$\mathcal{Y}_f = conv\{(a(T), c(T)) : T \in \mathcal{T}_f\}.$$

Now consider the problem

$$\max \left\{ \frac{b - z_1}{\sqrt{z_2}} : (z_1, z_2) \in \mathcal{Y}_f \right\}. \tag{2.17}$$

Since the objective function of (2.17) is quasi-convex on $\mathcal{Y}_f$ (Nikolova, 2008) and $\mathcal{Y}_f$ is a compact convex set, we have an optimal solution that is an extreme point of $\mathcal{Y}_f$ (Nikolova et al., 2006) and thus (2.17) gives equivalent solution with (2.16). Due to the nonincreasing objective function (2.15) on $\mathbb{B}^n$, the candidate extreme points of $\mathcal{Y}_f$ can be enumerated efficiently by parametric linear programming. We solve

$$\min \ z_1 + \lambda z_2 \ : \ (z_1, z_2) \in \mathcal{Y}_f \quad \text{for } \forall \lambda \geq 0 \tag{2.18}$$

and a single optimization

$$\min \ z_2 \ : \ (z_1, z_2) \in \mathcal{Y}_f. \tag{2.19}$$

For fixed $\lambda$, the optimal solutions for (2.19) is the sum of $f$ smallest $\sigma_j^2$, $j \in N$. The optimal solutions of (2.18) for each $\lambda \geq 0$ are $f$ smallest $\mu_j + \lambda\sigma_j^2$, $j \in N$. Since the order of $(\mu_j + \lambda\sigma_j^2)$ changes at most $\binom{n}{2}$ times as $\lambda$ ranges over $[0, +\infty)$, there are at most $\binom{n}{2}$ extreme points to consider. These can be enumerated by solving (2.18) for each $\lambda = \lambda_{ij}$, where $\lambda_{ij}$ satisfies

$$\mu_i + \lambda_{ij}\sigma_i^2 = \mu_j + \lambda_{ij}\sigma_j^2, \quad \forall i, j \in N, i \neq j.$$

Due to the cardinality constraint $\sum\limits_{j \in N} x_j = f$, only the order change of the $f$th and $(f+1)$th smallest items matters. In addition, an exchange of every pair $\{i, j\}$ is of interest of at most one value of $f$ (Atamtürk and Narayanan, 2009). With the proof in Atamtürk and Narayanan (2009), we can conclude that PKP with unit profit values for each items can be solved in $\mathcal{O}(n^3)$.

▷ **Case 3 :** $X = \mathbb{B}^n$**,** $p_j \sim \mathcal{O}(p(n))$ **for** $j \in N$

Lastly, we consider the case of PKP with $p_j \sim \mathcal{O}(p(n))$ for all $j \in N$, where $p(n)$ is a polynomial function of $n$. Same as CKP, the complexity of this case have not been proven. However, we provide an pseudo-polynomial dynamic programming algorithm in Section 3.1, of which the complexity can be reduced by specifying that $X = \mathbb{B}^n$. The complexity of the algorithm is $\mathcal{O}(nbU)$, where $U = \sum\limits_{j \in N} |p_j|$.

The summary of the complexity of PCO for the special cases is given in Table 2.4.

Table 2.4: Complexity of PCO for special cases

| Case | $X$ | Profit value | Complexity |
|---|---|---|---|
| 1 | $\mathbb{B}^n$ | $p_j \in \mathbb{Z}, \forall j \in N$ | $\mathcal{NP}$-hard |
| 2 | | $p_j = 1, \forall j \in N$ | $\mathcal{P}$ |
| 3 | | $p_j \sim \mathcal{O}(p(n))$,[a] $\forall j \in N$ | at most $\mathcal{O}(nbU)$[b] |
| 4 | $\mathcal{X}_{st}$[c] | $p_j = 0\ (\geq f), \forall j \in A$ | $\mathcal{NP}$-hard |
| 5 | | $p_j \in \mathbb{Z}_-, \forall j \in A$ | $\mathcal{NP}$-hard |
| 6 | | $p_j = -1\ (\geq f), \forall j \in A$ | $\mathcal{NP}$-hard |

[a] $p(n)$ : polynomial function of $n$

[b] $U$ : upper bound of $\sum_{j \in N} |p_j| x_j$, $x \in X$

[c] $\mathcal{X}_{st}$ : set of incidence vectors of $s$-$t$ paths for a given graph $G = (V, A)$ with $|V| = n$, $|A| = m$

# Chapter 3

# Exact Algorithms

In this chapter, we talk about the exact algorithm for the probability maximizing combinatorial optimization problem (PCO) and the chance-constrained combinatorial optimization problem (CCO). Though our study is focused on PCO, we also cover the exact algorithm of CCO since it has very similar form with that of PCO.

Prior to the construction of the exact algorithms of PCO and CCO, we assume that there is an exact algorithm for the constrained deterministic combinatorial optimization problem (C-DCO).

$$
\begin{aligned}
\text{(C-DCO)} \quad \text{minimize} \quad & \sum_{j \in N} w_j x_j \\
\text{subject to} \quad & \sum_{j \in N} p_j x_j \geq f, \\
& \sum_{j \in N} \mu_j x_j \leq t, \\
& x \in X,
\end{aligned}
$$

of which the complexity is $f(n, t, U)$. Here, $U$ is the upper bound of $\sum_{j \in N} p_j x_j$ for $x \in X$. In this chapter, we derive exact algorithms of PCO and CCO that use the exact algorithm of C-DCO.

## 3.1 Exact Algorithm of PCO

We first derive an exact algorithm of PCO. Consider the deterministic nonlinear formulation of PCO

$$
\begin{aligned}
\text{(PCO)} \quad & \text{maximize} \quad && \frac{b - \sum_{j \in N} \mu_j x_j}{\sqrt{\sum_{j \in N} \sigma_j^2 x_j}} && (3.1)\\
& \text{subject to} \quad && \sum_{j \in N} p_j x_j \geq f, &&\\
& && x \in X. &&
\end{aligned}
$$

Since the objective function (3.1) is a fractional function whose numerator is a linear and denominator is a nonlinear function of $x$, we split the objective function. We define a subproblem of PCO by adding a constraint which specifies the lower bound of the numerator as $k$ and substituting the objective function by the function in the root of the denominator. Then the subproblem has the form of

$$
\begin{aligned}
\text{(Sub-PCO}_k\text{)} \quad & \text{minimize} \quad && \sum_{j \in N} \sigma_j^2 x_j && (3.2)\\
& \text{subject to} \quad && \sum_{j \in N} \mu_j x_j \leq b - k, && (3.3)\\
& && \sum_{j \in N} p_j x_j \geq f, &&\\
& && x \in X, &&
\end{aligned}
$$

with $k = 1, \ldots, b-1$. Note that (3.3) is equivalent to $b - \sum_{j \in N} \mu_j x_j \geq k$. Then, we can obtain an optimal solution of PCO by solving Sub-PCO$_k$ for all $k = 1, \ldots, b-1$

and an additional optimization problem

$$\max\left\{\frac{k}{\sqrt{z_k}} : k = 1, \ldots, b-1\right\}, \tag{3.4}$$

where $z_k$ is the optimal value of each Sub-PCO$_k$ for $k = 1, \ldots, b-1$. Since Sub-PCO$_k$ has the form of C-DCO and $b - k \leq b$ for all $k = 1, \ldots, b-1$, we can solve each of Sub-PCO$_k$ in $f(n, b, U)$ and (3.4) can be calculated in $\mathcal{O}(b)$. Thus, the complexity of the exact algorithm of PCO is $\mathcal{O}(b \cdot f(n, b, U))$.

**Algorithm 1** Exact Algorithm of PCO

```
1: procedure ALGORITHM
2:     for k = 1, ..., b − 1 do
3:         Solve Sub-PCO_k and let z_k be the optimal objective value.
4:     return z* = max { k/√z_k : k = 1, ..., b }
```

Further, in the special cases of $X = \mathbb{B}^n$ or $X = \mathcal{X}_{st}$, we can reduce the complexity by $\mathcal{O}(f(n, b, U))$. For convenience, we call the problem as probability maximizing knapsack problem (PKP) and probability maximizing shortest path problem (PSP) when $X = \mathbb{B}^n$ and $X = \mathcal{X}_{st}$, respectively. Also we call their subproblems as Sub-PKP and Sub-PSP, and C-DCO in each case as C-DKP and C-DSP.

When $X = \mathbb{B}^n$, i.e., PKP, Sub-PKP$_k$ turns out to be a knapsack problem with additional constraint (3.3) of which the coefficients $\mu_j$ for $j \in N$ are all integer. Since this has identical form with C-DKP, we can use the dynamic programming algorithm with complexity $\mathcal{O}(nbU)$ (Kellerer et al., 2004). Since the DP algorithm contains states for every value of $\sum_{j \in S} p_j$ for all $S \subseteq N$, i.e., any $t'$ such that $0 \leq t' \leq t$, we can solve Sub-PCO$_k$ for all $k = 1, \ldots, b-1$ at once by solving C-DKP with $t = b - 1$. Therefore, the algorithm complexity can be reduced to $\mathcal{O}(f(n, b, U)) = \mathcal{O}(nbU)$.

**Algorithm 2** Exact Algorithm of PKP

**Define :** $\Pi(j,k,p)$ is the minimum value of $\sum_{j\in S}\sigma_j^2$ of any subset $S \subseteq J = \{1,\dots,j\}$ of which $b - \sum_{j\in S}\mu_j \geq k$ and $\sum_{j\in S} p_j \geq p$, for $j = 0,\dots,n$, $k = 0,\dots,b-1$, and $p = 0,\dots,f$.

```
procedure DP ALGORITHM
    for k = b − 1, ..., 0 do
        Π(0, k, 0) = 0
        for p = 1, ..., f do
            Π(0, k, p) = ∞
    for j = 1, ..., n do
        for p = 0, ..., p_j − 1 do
            for k = b − μ_j, ..., 0 do
                Π(j, k, p) = min{Π(j − 1, k, p), Π(j − 1, k + μ_j, 0) + σ_j^2}
            for k = b − 1, ..., b − μ_j + 1 do
                Π(j, k, p) = Π(j − 1, k, p)
        for p = p_j, ..., f − 1 do
            for k = b − μ_j, ..., 0 do
                Π(j, k, p) = min{Π(j − 1, k, p), Π(j − 1, k + μ_j, p − p_j) + σ_j^2}
            for k = b − 1, ..., b − μ_j + 1 do
                Π(j, k, p) = Π(j − 1, k, p)
        for p = f, ..., U do
            for k = b − μ_j, ..., 0 do
                Π(j, k, f) = min{Π(j − 1, k, f), Π(j − 1, k + μ_j, min{p − p_j, f}) + σ_j^2}
            for k = b − 1, ..., b − μ_j + 1 do
                Π(j, k, f) = Π(j − 1, k, f)
    for k = 1, ..., b − 1 do
        z_k = Π(n, k, f)
    return z* = min{ k/√(z_k) : k = 1, ..., b − 1 }
```

Similar logic can be adopted to PSP, where $X = \mathcal{X}_{st}$, since there exists a DP algorithm for multi-resource constrained shortest path problem in the form of C-DSP. Given a directed graph $G = (V, A)$ with the arc weight $a_j \sim N(\mu_j, \sigma_j^2)$ and the resource consumption $p_j$ for $j \in A$. Assume there is no negative cycle in $G$ and $\mu_j, p_j \in \mathbb{Z}$ for all $j \in N$. The DP algorithm runs in $\mathcal{O}(nmbU)$, where $n = |V|$ and $m = |A|$ (Ziegelmann and Mehlhorn, 2001). Then, the complexity of the exact algorithm of PSP can be reduced to $\mathcal{O}(nmbU)$. Note that since $m = \mathcal{O}(n^2)$, the complexity can be represented as $\mathcal{O}(n^3bU)$.

**Algorithm 3** Exact Algorithm of PSP

**Define :** $\Pi(l, v, k, p)$ is the minimum value of $\sum_{j \in \mathcal{P}} \sigma_j^2$ for a simple path $\mathcal{P}$ from 0 to $v$ with at most $l$ arcs that satisfies $b - \sum_{j \in \mathcal{P}} \mu_j \geq k$ and $\sum_{j \in \mathcal{P}} p_j \geq p$, for $l = 0, \ldots, n-1$,$v = 0, \ldots, n$, $k = 0, \ldots, b-1$, and $p = 0, \ldots, U$. Note that for $j \in A$ such that $j = (u, v)$ for some $u, v \in V$, $p_j$, $\mu_j$, $\sigma_j$ can be written as $p_{uv}$, $\mu_{uv}$, $\sigma_{uv}$, respectively.

**procedure** DP Algorithm
- **for** $k = 0, \ldots, b-1$ **do**
  - **for** $v = 0, \ldots, n$ **do**
    - **for** $p = 0, \ldots, U$ **do**
      - $\Pi(0, v, \mu, p) = \infty$
  - $\Pi(0, 0, k, 0) = 0$
- **for** $l = 1, \ldots, n-1$ **do**
  - **for** $v = 1, \ldots, n-1$ **do**
    - **for** $p = 1, \ldots, f$ **do**
      - **for** $k = b-1, \ldots, 1$ **do**
        - $\Xi(l, v, k, p) = \min_{u \in V^-(v) \,:\, \mu_{uv} \leq b-k} \left[\Pi(l-1, u, k + \mu_{uv}, \max\{p - p_{uv}, 0\}) + \sigma_{uv}^2\right]$
        - $\Pi(l, v, k, p) = \min\{\Pi(l-1, v, k, p), \Xi(l, v, k, p)\}$
- **for** $k = 1, \ldots, b-1$ **do**
  - **for** $v = 0, \ldots, n-1$ **do**
    - $d_k(v) = \min\{\Pi(n-1, v, \mu, p) : p \geq f - p_{vn}, \mu \geq k + \mu_{vn}\}$
  - $z_k = \min_{v \in V^-(n)} \{d_k(v) + \sigma_{vn}^2\}$
- **return** $z^* = \min\left\{\frac{k}{\sqrt{z_k}} : k = 1, \ldots, b-1\right\}$

## 3.2 Exact Algorithm of CCO

The exact algorithm of CCO follows a procedure similar to the exact algorithm of PCO. Given the deterministic nonlinear formulation of CCO

$$\begin{aligned} \text{(CCO)} \quad \text{maximize} \quad & \sum_{j \in N} p_j x_j \\ \text{subject to} \quad & \sum_{j \in N} \mu_j x_j + \Phi^{-1}(\rho) \sqrt{\sum_{j \in N} \sigma_j^2 x_j} \leq b, \qquad (3.5) \\ & x \in X. \end{aligned}$$

The constraint (3.5) is nonlinear and we can substitute it with the following three inequalities.

$$\begin{aligned} & \sum_{j \in N} \mu_j x_j \leq t, \\ & \sum_{j \in N} \sigma_j^2 x_j \leq \left( \frac{b-t}{\Phi^{-1}(\rho)} \right)^2, \qquad (3.6) \\ & t \leq b. \end{aligned}$$

Due to the integrality of $\mu_j$ for all $j \in N$, we only need to consider integer values of $t$. Since the coefficients of the constraint (3.6) are not integer, we define a subproblem of CCO by switching (3.6) with the objective function of CCO. Then the subproblem can be constructed as

$$
\begin{aligned}
(\text{Sub-CCO}_{(f,t)}) \quad & \text{minimize} \quad && \sum_{j\in N} \sigma_j^2 x_j \\
& \text{subject to} \quad && \sum_{j\in N} \mu_j x_j \leq t, \\
& && \sum_{j\in N} p_j x_j \geq f, \\
& && x \in X,
\end{aligned}
$$

for $f = 1, \ldots, U = \sum_{j\in N} |p_j|$ and $t = 1, \ldots, b-1$. Then, we can obtain an optimal solution of CCO by solving Sub-CCO$_{(f,t)}$ for $f = U, U-1, \ldots$ in decreasing order for all $t = 1, \ldots, b-1$ until we find a value $f$ that satisfies

$$
\left\{ t \ : \ t + \Phi^{-1}(\rho)\sqrt{z_{f,t}} \leq b, \ t = 1, \ldots, b-1 \right\} \neq \emptyset, \tag{3.7}
$$

where $z_{f,t}$ is the optimal value of Sub-CCO$_{(f,t)}$ for $f = 1, \ldots, U$ and $t = 1, \ldots, b-1$. Since Sub-CCO$_{(f,t)}$ has the form of C-DCO and $t \leq b$, we can solve each Sub-CCO$_{(f,t)}$ in $f(n, b, U)$ and (3.7) can be checked in $\mathcal{O}(b)$ for each $f$. Thus, the complexity of the exact algorithm of CCO is $\mathcal{O}(bU \cdot f(n, b, U))$.

**Algorithm 4** Exact Algorithm of CCO

```
procedure ALGORITHM
    for f = U, U − 1, . . . , 1 (in decreasing order) do
        for t = 1, . . . , b − 1 do
            Solve Sub-CCO_(f,t) and let z_f,t be the optimal objective value.
            if t + Φ^{-1}(ρ) sqrt(z_f,t) ≤ b then
                Stop. It is optimal solution.
                return f
            else
                Continue.
```

Identically with PCO, we denote CCO with $X = \mathbb{B}^n$ by chance-constrained knapsack problem (CKP) and CCO with $X = \mathcal{X}_{st}$ as chance-constrained shortest path problem (CSP). Then, the corresponding subproblems are called as Sub-CKP and Sub-CSP, and C-DCO as C-DKP and C-DSP, respectively. For CKP, the complexity can be reduced to $\mathcal{O}(f(n, b, U))$ similarly with the case of PKP. That is, we can solve Sub-CKP$_{(f,t)}$ for all $f = 1, \ldots, U$, $t = 1, \ldots, b$ simultaneously, by using DP algorithm. As mentioned, there exists a DP algorithm that can solve C-DKP in $\mathcal{O}(nbU)$ and thus, we can solve CKP in $\mathcal{O}(nbU)$.

**Algorithm 5** Exact Algorithm of CKP

**Define :** $\Pi(j, k, p)$ is the minimum value of $\sum_{j \in S} \sigma_j^2$ of subset $S \subseteq \{1, \ldots, j\}$ of which $b - \sum_{j \in S} \mu_j \geq k$ and $\sum_{j \in S} p_j \geq p$, for $j = 0, \ldots, n$, $k = 0, \ldots, b-1$, and $p = 0, \ldots, U$.

```
procedure DP ALGORITHM
    Set f = U and do the line 2 to 21 of Algorithm 2
    for f = U, U − 1, ..., 1 (in decreasing order) do
        for k = 1, ..., b − 1 do
            Set t = b − k
            if t + Φ^{-1}(ρ) √Π(n, t, p) ≤ b then
                Stop. It is optimal solution.
                return f
            else
                Continue.
```

Also for the case of $X = \mathcal{X}_{st}$, we can solve CSP in $\mathcal{O}(nmbU)$ adopting the DP algorithm for C-DSP.

**Algorithm 6** Exact Algorithm of CSP

**Define :** $\Pi(l, v, k, p)$ is the minimum value of $\sum_{j \in \mathcal{P}} \sigma_j^2$ for a simple path $\mathcal{P}$ from 0 to $v$ with at most $l$ arcs that satisfies $b - \sum_{j \in \mathcal{P}} \mu_j \geq k$ and $\sum_{j \in \mathcal{P}} p_j \geq p$, for $l = 0, \dots, n-1$,$v = 0, \dots, n$, $k = 0, \dots, b-1$, and $p = 0, \dots, U$. Note that for $j \in A$ such that $j = (u, v)$ for some $u, v \in V$, $p_j$, $\mu_j$, $\sigma_j$ can be written as $p_{uv}$, $\mu_{uv}$, $\sigma_{uv}$, respectively.

**procedure** DP ALGORITHM
    Set $f = U$ and do the line 2 to 12 of Algorithm 3
    **for** $k = 1, \dots, b-1$ **do**
        **for** $p = 1, \dots, U$ **do**
            $Pi(n, n, k, p) = \min_{\substack{v \in V^-(n) \\ : \mu_{vn} \leq b-k}} [\Pi(n-1, v, k + \mu_{vn}, \max\{p - p_{vn}, 0\}) + \sigma_{vn}^2]$
    **for** $p = U, U-1, \dots, 1$ (in decreasing order) **do**
        **for** $k = 1, \dots, b-1$ **do**
            Set $t = b - k$
            **if** $t + \Phi^{-1}(\rho)\sqrt{\Pi(n, n, t, p)} \leq b$ **then**
                Stop. It is optimal solution.
                **return** $p$
            **else**
                Continue.

# Chapter 4

# Approximation Scheme for Probability Maximizing Combinatorial Optimization Problem

This chapter suggests an approximation scheme for probability maximizing combinatorial optimization problem (PCO). For the approximation schemes, we use the relation between PCO and the chance-constrained combinatorial optimization problem (CCO). Consider the decision problem of PCO

$$\begin{aligned} \text{Instance :} \quad & N=\{1,\ldots,n\},\ X\subseteq\mathbb{B}^n,\ b,f\in\mathbb{Z},\ \rho\in[0.5,1],\ p_j\in\mathbb{Z},\forall j\in N \\ & a_j\sim N(\mu_j,\sigma_j^2) \text{ where } \mu_j\in\mathbb{Z},\sigma_j\in\mathbb{R},\forall j\in N \\ \text{Question :} \quad & \exists x\in X\ :\ \sum_{j\in N}p_jx_j\geq f,\ Pr\left(\sum_{j\in N}a_jx_j\leq b\right)\geq\rho\ ? \end{aligned} \tag{4.1}$$

We assume that $\exists S\subseteq N$ such that $\sum_{j\in S}\mu_j\leq b$ and $\sum_{j\in S}p_j\geq f$, so that $\rho\in[0.5,1]$. We can answer to this decision problem by solving CCO, and compare the optimal objective value to $f$. The lower and upper bound of the optimal objective value $\rho^*$ of PCO can be updated to tighten the range depending on the answer of (4.1). Thus, we can obtain an optimal solution of PCO by iteratively solving CCO with $\rho\in[0.5,1]$. However, since $\rho$ is a real number and there is no other algorithm known so far that gives an lower bound of the gap between the possible values of $Pr\left(\sum_{j\in S}a_j\leq b\right)$

for all $S \subseteq N$, we have to check all the values of $\rho$ that corresponds to $2^n$ subset $S \subseteq N$ to obtain an exact solution. Therefore, we attempt to attain an approximate solution.

Before we explain the approximation scheme, we first define the $\delta$-approximation scheme of CCO and the $\delta$-approximation scheme of PCO.

**Definition 4.1.** *A $\delta$-approximation scheme of CCO is the algorithm that gives a solution that satisfies*

$$x \in X,\ Pr\left(\sum_{j \in N} a_j x_j \leq b\right) \geq \rho - \delta,$$

*for a given value $\delta \in (0, \rho)$.*

*A $\delta$-approximation scheme of PCO is the algorithm that gives a solution that guarantees the absolute error of the resulting objective value $\rho$ to the optimal value $\rho^*$ is at most $\delta$;*

$$\rho^* - \delta \leq \rho,$$

*for any given value $\delta \in (0, \rho)$.*

Now, suppose that the optimal value of PCO is $\rho^*$. Then, we suggest a $\delta$-approximation scheme that guarantees to find a solution with the objective value $\rho$ that has at most $\delta > 0$ absolute error from $\rho^*$; $\rho^* - \rho \leq \delta$. This can be also interpreted as $(1 - 2\delta)$-approximation as the ratio since $\rho^* \geq 0.5$ and thus,

$$\rho^* - \delta = \rho^*(1 - \frac{\delta}{\rho^*}) \geq \rho^*(1 - 2\delta).$$

The overall structure of the approximation scheme is as follows.

1. Do bisection of the possible interval of the value of $\rho$ to choose a value $\rho$.

2. Solve CCO with the chosen $\rho$.

3. Update the interval and repeat until the length of the interval is small enough, i.e., less than $\delta$.

The following sections in this chapter are composed of the steps of the approximation scheme. In Section 4.1, we first cover the bisection procedure of $\rho$ and how to update the interval, which are step 1 and 3 of the approximation scheme. Then, in Section 4.2, we provide an approximation scheme of CCO for step 2. In Section 4.3, variation of the bisection procedure to reduce the practical computational time of the approximation scheme is presented. Lastly, in Section 4.4, we compare our approximation scheme to the approximation scheme of Nikolova (2009).

## 4.1 Bisection Procedure of $\rho$

In the approximation scheme, we set $\rho$ as a specific value considering the range of the possible value and solve CCO. Then we adjust the range according to the objective value obtained by solving CCO. To validate the bisection procedure, we first check the monotone condition of $\rho$.

**Proposition 4.2.** *For $0.5 \leq \rho_1, \rho_2 < 1$, if $\rho_1 > \rho_2$, then*

$$\left\{x \in X : \rho_1 \leq Pr\left(\sum_{j \in N} a_j x_j \leq b\right)\right\} \subseteq \left\{x \in X : \rho_2 \leq Pr\left(\sum_{j \in N} a_j x_j \leq b\right)\right\}.$$

Due to Proposition 4.2, we can do bisection on the range of $\rho$ and update lower and upper bound depending on the answer of the decision problem (4.1). By the assumption that $\rho^* \in [0.5, 1]$, we initially set the lower bound $LB_0 = 0.5$ and the upper bound $UB_0 = 1$.

Suppose that we have an exact algorithm to solve CCO. In each iteration $k$, we are given the lower bound $LB_{k-1}$ and the upper bound $UB_{k-1}$ of the possible value of $\rho$. Then we solve CCO with $\rho_k = \frac{LB_{k-1}+UB_{k-1}}{2}$ and update the interval.

- If $z_k \geq f$, then $LB_k = \rho_k$, $UB_k = UB_{k-1}$
- else if $z_k < f$, then $LB_k = LB_{k-1}$, $UB_k = \rho_k$

where $z_k$ is the optimal value of CCO with $\rho_k$. Then we repeat the iteration until we have $UB_m - LB_m \leq \delta$. This guarantees a solution that has at most $\delta$ absolute error from the optimal value $\rho^*$. Since the length of the interval is reduced by half in each iteration, the total number of iterations $m$ satisfies

$$(1-0.5)\cdot\left(\frac{1}{2}\right)^m \leq \delta \quad \Rightarrow \quad m \geq \log_2\left(\frac{1}{\delta}\right) - 1.$$

Thus, the total number of the iterations is $\mathcal{O}(\log(\frac{1}{\delta}))$.

However, if we use the exact algorithm of CCO given in Section 3.2, the complexity of the approximation scheme is $\mathcal{O}(bU\log(\frac{1}{\delta})f(n,b,U))$, which is worse than the complexity of the exact algorithm of PCO given in Section 3.1. Thus, we generate a new approximation scheme that in each iteration we solve CCO approximately, rather than solving it exactly.

Consider the case of which we have an $\alpha$-approximation scheme of CCO that gives the solution that guarantees the following bound

$$x \in X \ : \ Pr\left(\sum_{j\in N} a_j x_j \leq b\right) \geq \rho - \alpha, \tag{4.2}$$

for any $\alpha \in (0,\rho)$. In each iteration $k$, apply the approximation scheme of CCO with $\rho_k \in [LB_{k-1}, UB_{k-1}]$ and update the interval as the following.

- If $z_k \geq f$, then $LB_k = \rho_k - \alpha$, $UB_k = UB_{k-1}$
- else if $z_k < f$, then $LB_k = LB_{k-1}$, $UB_k = \rho_k$,

where $z_k$ is the objective value of the $\alpha$-approximate solution of CCO. We stop the iteration when we have $UB_m - LB_m \leq \delta$.

Now, we have to derive the method to select the value of $\rho_k$ from the range $[LB_{k-1}, UB_{k-1}]$ to reduce the range efficiently. Suppose that in each iteration $k$ we can reduce the length of the interval to be at most $\beta_k (< 1)$ times the length prior to the iteration. Then, the following two conditions should be satisfied.

$$(\rho_k - LB_{k-1}) \leq \beta_k \cdot (UB_{k-1} - LB_{k-1}) \;\Rightarrow\; \rho_k \leq \beta_k \cdot UB_{k-1} + (1-\beta_k) \cdot LB_{k-1}, \tag{4.3}$$

$$(UB_{k-1} - \rho_k + \alpha) \leq \beta_k \cdot (UB_{k-1} - LB_{k-1}) \;\Rightarrow\; \rho_k \geq (1-\beta_k) \cdot UB_{k-1} + \beta_k \cdot LB_{k-1} + \alpha. \tag{4.4}$$

Since the upper bound and the lower bound of $\rho_k$ given in (4.3) and (4.4), respectively, have to satisfy

$$(1-\beta_k) \cdot UB_{k-1} + \beta_k \cdot LB_{k-1} + \alpha \leq \beta_k \cdot UB_{k-1} + (1-\beta_k) \cdot LB_{k-1}$$

to guarantee the existence of such $\rho_k$, we can generate an additional condition

$$(2\beta_k - 1) \cdot (UB_{k-1} - LB_{k-1}) \geq \alpha. \tag{4.5}$$

Thus, while (4.5) is satisfied, we can reduce the range of possible $\rho$ values by the factor of $\beta_k$ if we set $\rho_k$ as the value that satisfied (4.3) and (4.4). We let $\beta_k = \frac{1}{2} + \frac{\alpha}{2(UB_{k-1} - LB_{k-1})}$, and it induces

$$\rho_k = \frac{UB_{k-1} + LB_{k-1} + \alpha}{2}. \tag{4.6}$$

**Proposition 4.3.** *If we have an $\alpha$-approximation scheme of CCO, then we can generate a $t\alpha$-approximation scheme of PCO that runs the $\alpha$-approximation scheme of CCO $\mathcal{O}(\log(\frac{1}{\alpha}))$ times, for any fixed $t > 1$.*

*Proof.* In each iteration, set $\rho_k$ as (4.6) and apply the $\alpha$-approximation of CCO. If the objective value of the solution $z_k \geq f$, then let $LB_k = \rho_k - \alpha$, $UB_k = UB_{k-1}$ and otherwise, let $LB_k = LB_{k-1}$, $UB_k = \rho_k$. Then, in either of the case, we have

$$UB_k - LB_k = \beta_k \cdot (UB_{k-1} - LB_{k-1}),$$

where $\beta_k = \frac{1}{2} + \frac{\alpha}{2(UB_{k-1} - LB_{k-1})}$.
To guarantee $t\alpha$-approximation of PCO, we do the iteration until we have

$$UB_m - LB_m \leq t\alpha. \tag{4.7}$$

Note that the reduction ratio $\beta_k$ of the length of the interval $[LB_{k-1}, UB_{k-1}]$ is inverse proportional to the length. Thus, we have

$$\beta_k \leq \beta_{k+1}, \ \forall k \geq 1. \tag{4.8}$$

Let $m$ be the total number of the iterations, then we have

$$UB_{m-1} - LB_{m-1} > t\alpha, \quad UB_m - LB_m \leq t\alpha.$$

Thus,

$$\beta_m = \frac{1}{2} + \frac{\alpha}{2(UB_{m-1} - LB_{m-1})} < \frac{1}{2} + \frac{1}{2t} = \frac{t+1}{2t},$$

and due to (4.8),

$$\beta_k < \frac{t+1}{2t}, \quad \forall k = 1, \ldots, m. \tag{4.9}$$

Then for any fixed $t > 1$, we have the following by (4.9).

$$(1 - 0.5) \cdot \left(\frac{t+1}{2t}\right)^m \leq t\alpha \quad \Rightarrow \quad m \geq \log_{\frac{2t}{t+1}} \left(\frac{1}{2t\alpha}\right),$$

that is $m = \mathcal{O}(\log(\frac{1}{\alpha}))$. □

Thus, we can generate a $t\alpha$-approximation scheme of PCO that requires the $\alpha$-approximation scheme of CCO $\mathcal{O}(\log(\frac{1}{\alpha}))$ times, for any fixed $t > 1$. Since our goal is to generate a $\delta$-approximation of PCO, we need an $\alpha$-approximation scheme of CCO that satisfies $t\alpha = \delta$ for a fixed $t > 1$. Suppose that the corresponding $\alpha$-approximation scheme of CCO has the complexity of $h(n, b, U)$. Then the $\delta$-approximation scheme of PCO has the complexity of $\mathcal{O}(\log(\frac{1}{\delta}) \cdot h(n, b, U))$.

**Algorithm 7** $\delta$-Approximation Scheme of PCO

**procedure** BISECTION
  Initialize $LB_0 = 0.5$ and $UB_0 = 1$
  **while** $UB_{k-1} - LB_{k-1} \geq \delta$ **do**
    Set $\rho_k = \frac{UB_{k-1}+LB_{k-1}+\alpha}{2}$.
    Apply the $\alpha$-approximation scheme to CCO with $\rho_k$ and let the value of $Pr\left(z \leq \frac{b - \sum_{j \in N} \mu_j x_j}{\sqrt{\sum_{j \in N} \sigma_j^2 x)j}}\right)$ of the optimal solution be $\rho'$.
    **if** $z_k \geq f$ **then**
      Save $\rho'$ as current optimal value $\rho^*$ of PCO.
      Update $LB_k = \rho_k - \alpha$ and $UB_k = UB_{k-1}$.
    **else**
      Update $LB_k = LB_{k-1}$ and $UB_k = \rho_k$.
  **return** $\rho^*$

## 4.2 Approximation Scheme of CCO

In this section, we suggest an $\alpha$-approximation scheme of CCO, which is an extension of the approximation scheme of CKP in Han et al. (2016) to the general combinatorial optimization problem with $X \subseteq \mathbb{B}^n$, CCO. We first define a deterministic combinatorial optimization problem (DCO) as

$$
\begin{aligned}
\text{(DCO)} \quad \text{minimize} \quad & \sum_{j \in N} w_j x_j \\
\text{subject to} \quad & \sum_{j \in N} p_j x_j \geq f, \\
& x \in X.
\end{aligned}
$$

It is obvious that DCO is a special case of C-DCO of which the second constraint of C-DCO, i.e., $\sum_{j \in N} \mu_j x_j \leq t$, is redundant. We assume that there is an exact algorithm of DCO with the complexity $g(n, b, U)$, where $b$ is an upper bound of $\sum_{j \in N} w_j x_j$ and $U$ is an upper bound of $\sum_{j \in N} p_j x_j$ for $x \in X$. For the case of $X = \mathbb{B}^n$, which we denote by the deterministic knapsack problem (DKP), the algorithm can be a DP algorithm that runs in $\mathcal{O}(nU)$. Note that by extending the DP algorithm, C-DKP can be solved in $\mathcal{O}(nbU)$. We propose an approximation scheme of CCO, which iteratively uses the exact algorithm of DCO.

Remind the deterministic nonlinear formulation of CCO

$$
\begin{aligned}
\text{(CCO)} \quad & \text{maximize} \quad && \sum_{j \in N} p_j x_j \\
& \text{subject to} \quad && \sum_{j \in N} \mu_j x_j + \Phi^{-1}(\rho) \sqrt{\sum_{j \in N} \sigma_j^2 x_j} \leq b, \\
& && x \in X.
\end{aligned}
$$

The feasible set of CCO

$$
\left\{ x \in X : \sum_{j \in N} \mu_j x_j + \Phi^{-1}(\rho) \sqrt{\sum_{j \in N} \sigma_j^2 x_j} \leq b \right\}
$$

can be reformulated as

$$
\left\{ x \in X : \sum_{j \in N} a_j x_j \leq b, \forall a \in \mathcal{U} \right\}, \tag{4.10}
$$

where

$$
\mathcal{U} = \left\{ \mu + \Phi^{-1}(\rho) \Sigma^{1/2} \epsilon : \sum_{j \in N} \epsilon_j^2 \leq (\Phi^{-1}(\rho))^2 \right\}, \; \mu = \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_n \end{bmatrix}, \Sigma = \begin{bmatrix} \sigma_1^2 & & \\ & \ddots & \\ & & \sigma_n^2 \end{bmatrix}.
$$

Then, define a loss function $f(\epsilon_j) = \epsilon_j^2$ and approximate it by a piecewise linear function as shown in Figure 4.1.

The interval $[0, \left(\Phi^{-1}(\rho)\right)^2]$ along the vertical axis is divided into $m$ segments of equal length, $[\pi_j^0 = 0, \pi_j^1], [\pi_j^1, \pi_j^2], \ldots, [\pi_j^{m-1}, \pi_j^m = \left(\Phi^{-1}(\rho)\right)^2]$, and every two successive points are connected by a straight line. Then, define a piecewise linear

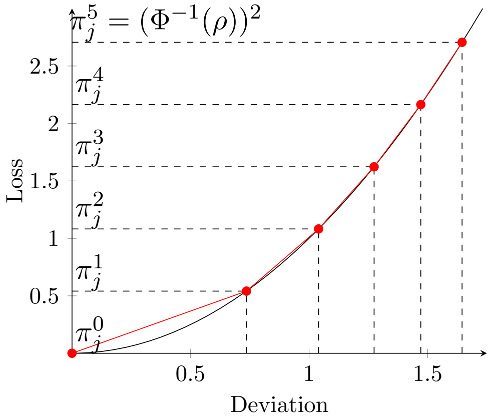


Figure 4.1: Approximation of quadratic loss function when $\rho = 0.95$ and $m = 5$

loss function $l_m(\cdot)$ over $[0, \Phi^{-1}(\rho)]$ as

$$l_m(\epsilon_j) = \frac{\pi_j^m}{m}\left(k - 1 + \frac{\epsilon_j - \sqrt{\pi_j^{k-1}}}{\sqrt{\pi_j^k} - \sqrt{\pi_j^{k-1}}}\right), \text{ if } \sqrt{\pi_j^{k-1}} \le \epsilon_j \le \sqrt{\pi_j^k} \text{ for } k \in M,$$

where $M = \{1, \dots, m\}$ is a set of the linear segments. Then the ellipsoidal uncertainty set $\mathcal{U}$ can be approximated by $\mathcal{U}_m$ :

$$\mathcal{U}_m := \left\{\mu + \Sigma^{1/2}\epsilon : \sum_{j \in N} l_m(\epsilon_j) \le \left(\Phi^{-1}(\rho)\right)^2\right\}.$$

Since the loss function $f(\cdot)$ is convex, the piecewise linear approximation function $l_m(\cdot)$ is also a convex function and thus $\mathcal{U}_m$ is a bounded convex polyhedron. Furthermore, it satisfies

$$\mathcal{U}_m = \left\{\mu + \Sigma^{1/2}\epsilon : \epsilon \in conv(\varepsilon)\right\}, \tag{4.11}$$

where

$$\varepsilon = \left\{ \epsilon \in \mathbb{R}^n : \sum_{j \in N} l_m(\epsilon_j) \leq \left(\Phi^{-1}(\rho)\right)^2, \epsilon_j \in \left\{0, \sqrt{\pi_j^1}, \dots, \sqrt{\pi_j^m}\right\}, \forall j \in N \right\}.$$

Now, define $\mathrm{RCO}_m$, the robust approximation of CCO, which is a robust combinatorial optimization problem with a polyhedral uncertainty set $\mathcal{U}_m$ as (4.11).

$$\begin{aligned} (\mathrm{RCO}_m) \quad \text{maximize} \quad & \sum_{j \in N} p_j x_j \\ \text{subject to} \quad & \sum_{j \in N} a_j x_j \leq b, \qquad \forall a \in \mathcal{U}_m, \\ & x \in X. \end{aligned} \tag{4.12}$$

Let $z_{\mathrm{CCO}}$ and $z_{\mathrm{RCO}_m}$ be the optimal values of CCO and $\mathrm{RCO}_m$, respectively, for a given threshold probability $\rho$ and the number of linear segment $m$. Then by Han et al. (2016), we have

$$z_{\mathrm{CCO}} \leq z_{\mathrm{RCO}_m}.$$

Moreover, let $F_m$ be the feasible solution set to $\mathrm{RCO}_m$ and

$$\rho_m := \inf_{x \in F_m} P\left\{ \sum_{j \in N} a_j x_j \leq b \right\}.$$

Then, it is proven in Han et al. (2016) that $\lim_{m \to \infty} \rho_m = \rho$ and

$$\rho_m \geq \rho - \frac{1}{\sqrt{2\pi}} \Phi^{-1}(\rho) \left(1 - \sqrt{1 - \frac{n}{4m}}\right) e^{-\left(\Phi^{-1}(\rho)\right)^2 \left(1 - \frac{n}{4m}\right)/2}. \tag{4.13}$$

Thus by solving $\text{RCO}_m$, we can obtain an approximation solution that guarantees

$$Pr\left(\sum_{j\in N} a_j x_j \leq b\right) \geq \rho_m,\ x \in X,$$

with $\rho_m$ that satisfies (4.13). Suppose that $\alpha_m = \rho - \rho_m$, where $m$ is the number of linear segments in $\mathcal{U}_m$. In Han et al. (2016), they provided an upper bound of $\alpha_m$ as

$$\alpha_m \leq \frac{1}{\sqrt{2\pi}}\Phi^{-1}(\rho)\left(1-\sqrt{1-\frac{n}{4m}}\right),$$

which is dependent on $\rho$. However, we propose an upper bound that is independent of the value of $\rho$.

**Proposition 4.4.** *For $m$, the number of linear segments, we can guarantee*

$$\rho - \rho_m = \alpha_m \leq \frac{n}{\sqrt{2\pi e}(4m-n)}. \tag{4.14}$$

*Proof.* Let $C(\rho) = \frac{1}{\sqrt{2\pi}}\Phi^{-1}(\rho)$, then (4.13) is

$$\alpha_m \leq C(\rho)\left(1-\sqrt{1-\frac{n}{4m}}\right)e^{-(\Phi^{-1}(\rho))^2(1-\frac{n}{4m})/2}.$$

Since $\ln(\cdot)$ is a nondecreasing function,

$$\begin{aligned}
\ln \alpha_m &\leq \ln\left(C(\rho)\right) + \ln\left(1-\sqrt{1-\frac{n}{4m}}\right) - \left(\Phi^{-1}(\rho)\right)^2\left(1-\frac{n}{4m}\right)/2 \\
&= \ln\left(C(\rho)\right) + \ln\left(1-\sqrt{1-\frac{n}{4m}}\right) - \pi\left(C(\rho)\right)^2\left(1-\frac{n}{4m}\right).
\end{aligned}$$

Let $t = 1 - \frac{n}{4m}$ and $u = C(\rho)$, and then we obtain following inequality

$$\ln \alpha_m \leq \ln u + \ln\left(1 - \sqrt{t}\right) - \pi t u^2. \tag{4.15}$$

Define a function $f(u)$ as

$$f(u) = \ln u + \ln\left(1 - \sqrt{t}\right) - \pi t u^2.$$

Then we have

$$\begin{aligned} f'(u) &= \frac{1}{u} - 2\pi t u, \\ f''(u) &= -\frac{1}{u^2} - 2\pi t < 0. \end{aligned}$$

Therefore, $f(u)$ is an concave function that has maximum value at $u = \frac{1}{\sqrt{2\pi t}}$. Since $u = C(\rho) \in [0, \infty)$ when $\rho \in [0, 1]$, there can be a value of $\rho$ with $u = \frac{1}{\sqrt{2\pi t}}$. Applying this to (4.15), we have

$$\begin{aligned} \ln \alpha_m &\leq \ln u + \ln\left(1 - \sqrt{t}\right) - \pi t u^2 \\ &\leq \ln\left(\frac{1}{\sqrt{2\pi t}}\right) - \pi t \frac{1}{2\pi t} + \ln\left(1 - \sqrt{t}\right) \\ &= \ln\left(\frac{1 - \sqrt{t}}{\sqrt{2\pi t e}}\right). \end{aligned}$$

By converting $t$ back to $1 - \frac{n}{4m}$, we have

$$\begin{aligned} &\ln \alpha_m \leq \ln\left(\frac{1 - \sqrt{1 - \frac{n}{4m}}}{\sqrt{2\pi e\left(1 - \frac{n}{4m}\right)}}\right) \\ \Leftrightarrow \quad &\alpha_m \leq \frac{1 - \sqrt{1 - \frac{n}{4m}}}{\sqrt{2\pi e\left(1 - \frac{n}{4m}\right)}}. \end{aligned}$$

Since

$$\frac{1-\sqrt{1-\frac{n}{4m}}}{\sqrt{2\pi e\left(1-\frac{n}{4m}\right)}} = \frac{\frac{n}{4m}}{\sqrt{2\pi e\left(1-\frac{n}{4m}\right)}\left(1+\sqrt{1-\frac{n}{4m}}\right)}$$
$$\leq \frac{\frac{n}{4m}}{\sqrt{2\pi e\left(1-\frac{n}{4m}\right)}}$$
$$\leq \frac{\frac{n}{4m}}{\sqrt{2\pi e}\left(1-\frac{n}{4m}\right)}$$
$$= \frac{n}{\sqrt{2\pi e}(4m-n)}.$$

Thus, we have the following final inequality

$$\alpha_m \leq \frac{n}{\sqrt{2\pi e}(4m-n)}.$$

□

**Corollary 4.5.** *The number of linear segments $m$ is $\mathcal{O}(\frac{n}{\alpha})$.*

*Proof.* By (4.14) of Proposition 4.4, it is sufficient to set $m$ that satisfies the condition

$$\alpha \geq \frac{n}{\sqrt{2\pi e}(4m-n)}$$

to guarantee the absolute error less than $\alpha$. Thus,

$$m \geq \frac{\sqrt{2\pi e}+\frac{1}{\alpha}}{4\sqrt{2\pi e}}n,$$

which is $m = \mathcal{O}(\frac{n}{\alpha})$. □

Now, we describe how to solve $\mathrm{RCO}_m$. The procedure is the extension of that in

Han et al. (2016) from $X = \mathbb{B}^n$ to the general set $X \subseteq \mathbb{B}^n$. First, reformulate $\mathcal{U}_m$ as

$$\mathcal{U}_m = \left\{ a \in \mathbb{R}^n : a_j = \mu_j + \sum_{k \in M} d_j^k z_j^k, \forall j \in N, z \in conv(\mathcal{Z}) \right\}, \tag{4.16}$$

where

$$\mathcal{Z} := \{ z \in \mathbb{B}^{mn} : \sum_{j \in N} \sum_{k \in M} z_j^k \leq m, z_j^k \leq z_j^{k-1}, \forall j \in N, k \in M \backslash \{1\} \} \tag{4.17}$$

and $d_j^k = \sigma_j \left( \sqrt{\pi_j^k} - \sqrt{\pi_j^{k-1}} \right)$ for all $j \in N$ and $k \in M$. Then, using (4.16), $\mathrm{RCO}_m$ can be restated a mixed integer linear program with the number of variables and constraints polynomially bounded by $m$ and $n$.

$$\begin{aligned} (\mathrm{RCO}_m) \quad \text{maximize} \quad & \sum_{j \in N} p_j x_j \\ \text{subject to} \quad & \sum_{j \in N} \mu_j x_j + \beta(x, m) \leq b, \\ & x \in X, \end{aligned}$$

where

$$\begin{aligned} \beta(x, m) = \ \text{maximize} \quad & \sum_{j \in N} \sum_{k \in M} d_j^k x_j z_j^k \\ \text{subject to} \quad & \sum_{j \in N} \sum_{k \in M} z_j^k \leq m, \\ & z_j^k \leq z_j^{k-1}, && j \in N, \ k \in M \backslash \{1\}, \\ & z_j^k \in \{0, 1\}, && \forall j \in N, \ k \in M. \end{aligned}$$

Since $d_j^1 \geq \ldots \geq d_j^m$ for all $j \in N$, optimal value of $\beta(x,m)$ for a nonnegative $x$ is equal to that of its linear relaxation. Thus, by using the dual of the linear relaxation of $\beta(x,m)$, we can reformulate RCO$_m$ with $2mn+1$ variables and $nm+1$ constraints.

$$
\begin{aligned}
(\text{RCO}_m) \quad \text{maximize} \quad & \sum_{j\in N} p_j x_j \\
\text{subject to} \quad & \sum_{j\in N} \mu_j x_j + my + \sum_{j\in N}\sum_{k\in M} v_j^k \leq b, \\
& y - w_j^{k+1} + v_j^k \geq d_j^k x_j, && \forall j \in N, k = 1, \\
& y + w_j^k - w_j^{k+1} + v_j^k \geq d_j^k x_j, && \forall j \in N, k = 2, \ldots, m-1, \\
& y + w_j^k + v_j^k \geq d_j^k x_j, && \forall j \in N, k = m, \\
& w_j^k \geq 0, && \forall j \in N, k \in M \backslash \{1\}, \\
& v_j^k \geq 0, && \forall j \in N, k \in M, \\
& y \geq 0, \\
& x \in X.
\end{aligned}
$$

The above formulation cannot be solved in polynomial time and has weak theoretical lower bounds. Thus, we decompose RCO$_m$ so that we can rather solve $(nm - m + 1)$ DCOs. Let $D = \{(j,k) : j \in N, k \in M\}$ be the set of all linear segments, and $D^+ = D \cup \{(n+1,1)\}$, with an artificial segment $(n+1,1)$ such that $d_{n+1}^1 = 0$. For $S \subseteq N$, define $D(S) = \{(j,k) \in D : j \in S\}$, then $|D(S)| \geq m$ if $S \neq \emptyset$. For each $(j,k) \in D^+$, let $r_j^k = |\{(p,q) \in D^+ : d_p^q > d_j^k\}| + |\{(p,q) \in D^+ : d_p^q = d_j^k, p \leq j\}|$.

Then we have

$$1 \leq r_j^k \leq nm + 1, \tag{4.18}$$

$$(j,k) \neq (p,q) \;\Rightarrow\; r_j^k \neq r_p^q. \tag{4.19}$$

For $l \in \{1, \ldots, nm+1\}$, let $h_l = d_j^k$ such that $l = r_j^k$ and let $D_l = \{(j,k) \in D : d_j^k > h_l\}$. Then, we have the following proposition.

**Proposition 4.6.** *Let $T$ be the set of feasible solutions to $RCO_m$, then*

$$T = \cup_{l \in \{m, m+1, \ldots, nm-1, nm+1\}} T_l, \tag{4.20}$$

*where $T_l = \{x \in X : \sum_{j \in N} \mu_j x_j + \sum_{(j,k) \in D_l} (d_j^k - h_l) x_j \leq b - m h_l\}$.*

*Proof.* Let $V$ be the set of feasible solutions to $\mathrm{RCO}_m$ when $X = \mathbb{B}^n$ and let $V_l = \{x \in \mathbb{B}^n : \sum_{j \in N} \mu_j x_j + \sum_{(j,k) \in D_l} (d_j^k - h_l) x_j \leq b - m h_l\}$. Then, we have

$$T = V \cap X, \quad T_l = V_l \cap X, \ \forall l \in \{m, m+1, \ldots, nm-1, nm+1\}. \tag{4.21}$$

By Han et al. (2016), it is proven that

$$V = \cup_{l \in \{m, m+1, \ldots, nm-1, nm+1\}} V_l,$$

and thus,

$$
\begin{aligned}
T &= V \cap X \\
&= \big(\cup_{l \in \{m, m+1, \ldots, nm-1, nm+1\}} V_l\big) \cap X \\
&= \cup_{l \in \{m, m+1, \ldots, nm-1, nm+1\}} (V_l \cap X) \\
&= \cup_{l \in \{m, m+1, \ldots, nm-1, nm+1\}} T_l
\end{aligned}
$$

$\square$

Proposition 4.6 induces that $\mathrm{RCO}_m$ can be solved by solving $(nm - m + 1)$ combinatorial optimization problem $\mathrm{CO}_l$, $l = m, m+1, \ldots, nm-1, nm+1$, defined as the following.

$$
\begin{aligned}
(\mathrm{CO}_l) \quad & \text{maximize} \quad && \sum_{j \in N} p_j x_j && (4.22) \\
& \text{subject to} \quad && \sum_{j \in N} \mu_j x_j + \sum_{(j,k) \in D_l} (d_j^k - h_l) x_j \leq b - m h_l, && (4.23) \\
& && x \in X.
\end{aligned}
$$

Finally, by switching the objective function (4.22) and the constraint (4.23), we obtain the formulation of $\text{Sub-CO}_{(l,f)}$ as

$$
\begin{aligned}
(\text{Sub-CO}_{(l,f)}) \quad & \text{minimize} \quad && \sum_{j \in N} \mu_j x_j + \sum_{(j,k) \in D_l} (d_j^k - h_l) x_j \\
& \text{subject to} \quad && \sum_{j \in N} p_j x_j \geq f, \\
& && x \in X,
\end{aligned}
$$

for $l = m, m+1, \ldots, nm-1, nm+1$ and $f = 1, \ldots, U = \sum_{j \in N} |p_j|$. Thus, by solving Sub-CO$_{(l,f)}$ for $f = U, U-1, \ldots$ in decreasing order until we have a solution with the objective value $z_{(l,f)} \le b - mh_l$ for each $l$, we can attain an optimal solution of CO$_l$. Note that Sub-CO$_{(l,f)}$ has the form of DCO that was defined earlier. Since we assumed that there exists an exact algorithm that solves each DCO in $g(n, b, U)$, we can solve a single Sub-CO$_{(l,f)}$ in $g(n, b, U)$.

To organize,

- We can obtain an $\alpha$-approximation solution of CCO by solving RCO$_m$.
- The solution of RCO$_m$ can be attained by solving $(nm - m + 1)$ CO$_l$ for $l = m, m+1, \ldots, nm-1, nm+1$.
- Each CO$_l$ for a fixed $l \in \{m, m+1, \ldots, nm-1, nm+1\}$ can be solved by solving Sub-CO$_{(l,f)}$ for $f = U, U-1, \ldots, 1$ until we have a solution that satisfies (4.23), which is at most $U$ times.
- Each Sub-CO$_{(l,f)}$ has the form of DCO which has an algorithm with of the complexity $g(n, b, U)$.

Consequently, we have an $\alpha$-approximation scheme of CCO that solves Sub-CO$_{(l,f)}$ at most $(nm - m + 1)U$ times, with complexity of $\mathcal{O}(nmU \cdot g(n, b, U))$.

In addition, due to Corollary 4.5, the number of the linear segments $m$ defined to approximate the uncertainty set can be represented as $\mathcal{O}(\frac{n}{\alpha})$. Thus, we have the complexity of the $\alpha$-approximation scheme of CCO as $\mathcal{O}(n^2 U \frac{1}{\alpha} \cdot g(n, b, U))$.

**Algorithm 8** $\alpha$-Approximation scheme of CCO

**procedure** ALGORITHM<br>
    **for** $j = 1, \ldots, n$ **do**<br>
        **for** $k = 1, \ldots, m$ **do**<br>
            Calculate $d_j^k = \sigma_j(\sqrt{\pi_j^k} - \sqrt{\pi_j^{k-1}})$, where $\pi_j^k = \frac{k}{m}(\Phi^{-1}(\rho))^2$.<br>
    Define $D$ and $D^+$.<br>
    **for** $(j,k) \in D^+$ **do**<br>
        Calculate $r_j^k = |\{(p,q) \in D^+ : d_p^q > d_j^k\}| + |\{(p,q) \in D^+ : d_p^q = d_j^k, p \leq j\}|$.<br>
    **for** $l \in \{m, \ldots, nm-1, nm+1\}$ **do**<br>
        Let $h_l = d_j^k$ for $l = r_j^k$<br>
        Define $D_l = \{(j,k) \in D : d_j^k > h_l\}$.<br>
        **for** $f = U, U-1, \ldots$ (in decreasing order) **do**<br>
            Formulate and solve Sub-CO$_{(l,f)}$. Let the optimal value be $z_{(l,f)}$.<br>
            **if** $z_{(l,f)} \leq b - mh_l$ **then**<br>
                Stop. Set $f_l = f$.<br>
            **else**<br>
                Continue.<br>
    **return** $z^* = \max\{f_l : l = m, m+1, \ldots, nm-1, nm+1\}$.

Combining Algorithm 7 and Algorithm 8, we have a $\delta$-approximation scheme of PCO with complexity $\mathcal{O}(n^2 U \frac{1}{\delta} \log(\frac{1}{\delta}) \cdot g(n, b, U))$, where $\delta = t\alpha$ for a fixed $t > 1$.

## 4.3 Variation of the Bisection Procedure of $\rho$

The approximation scheme of PCO takes much computational time for several reasons. We propose a variation of the bisection procedure to reduce the practical computational time.

Remind the bound of the number of linear segments $m$ to guarantee the absolute error of the probability $\rho$ of CCO is at most $\alpha$, given in Section 4.1.

$$m \geq \frac{\sqrt{2\pi e} + \frac{1}{\alpha}}{4\sqrt{2\pi e}} n = \mathcal{O}(n/\alpha), \tag{4.24}$$

which is independent of $\rho$. Since it is inversely proportional to $\alpha$, the smaller $\alpha$ is, the larger $m$ is needed to guarantee the error less than $\alpha$. This implies that we need to solve more combinatorial optimization problems to approximate CCO, which can be $\alpha$-approximated by solving $(nm - m + 1)$ ordinary combinatorial optimization problems.

Therefore, to reduce the time of the approximation scheme of CCO, and that of the approximation scheme of PCO as well, we set the value of $\alpha$ according to the length of the interval in each iteration, rather than using fixed value which is very small. We previously checked that the reduction ratio in iteration $k$ is

$$\beta_k = \frac{1}{2} + \frac{\alpha}{2(UB_{k-1} - LB_{k-1})} \tag{4.25}$$

and it increases as $k$ becomes larger, since the interval $UB_{k-1} - LB_{k-1}$ decrease. Note that the reduction ratio in last iteration $m$ is at most $\frac{t+1}{2t}$, where $t\alpha = \delta$ with a fixed $t > 1$, and we have

$$\beta_1 \leq \beta_2 \leq \cdots \leq \beta_m \leq \frac{t+1}{2t}.$$

In the variation of the bisection procedure of $\rho$, we fix the ratio of the reduction of the interval by $\frac{t+1}{2t}$ and define $\alpha_k$ as a function of $UB_{k-1} - LB_{k-1}$ in each iteration. We choose $\rho_k$ by the same method as in Section 4.1, except that $\alpha$ is not fixed. We have

$$\rho_k = \frac{LB_{k-1} + UB_{k-1} + \alpha_k}{2}$$

and update the value of $LB_k$ and $UB_k$ as

- if $obj_k \geq f$ : $LB_k = \rho_k - \alpha_k$, $UB_k = UB_{k-1}$,
- if $obj_k < f$ : $LB_k = LB_{k-1}$, $UB_k = \rho_k$.

Then in the either case, the interval length changes as

$$UB_{k-1} - LB_{k-1} \quad \rightarrow \quad \frac{UB_{k-1} - LB_{k-1} + \alpha_k}{2}.$$

Thus, to have the fixed reduction ratio $\frac{t+1}{2t}$, the following equation has to be satisfied.

$$\frac{UB_{k-1} - LB_{k-1} + \alpha_k}{2} = \frac{t+1}{2t}(UB_{k-1} - LB_{k-1}),$$

which gives the function of $\alpha_k$ as

$$\alpha_k = \frac{1}{t}(UB_{k-1} - LB_{k-1}).$$

Then, we have

$$\rho_k = \frac{(t+1)UB_{k-1} + (t-1)LB_{k-1}}{2t}.$$

Figure 4.2 shows how we select $\rho_k$ and update the lower and upper bound of $\rho$ in each iteration $k$, for the case of $t = 2$.

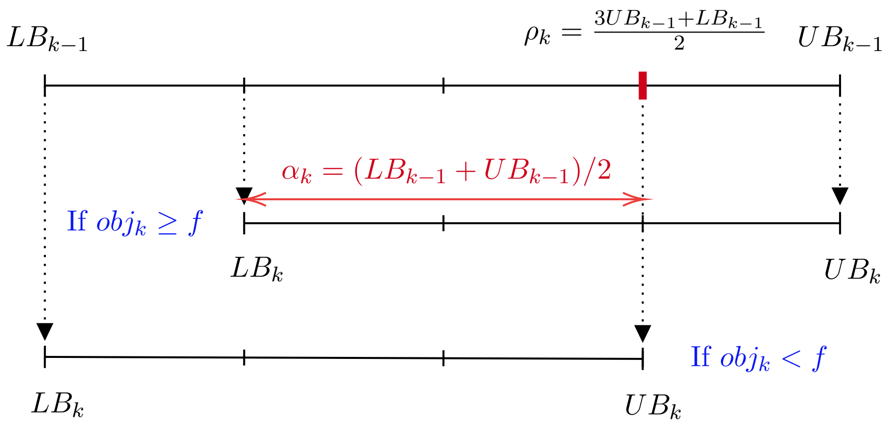


Figure 4.2: Bisection with the fixed reduction ratio $\frac{t+1}{2t}$, where $t = 2$

Since the initial lower and upper bounds are $LB_0 = 0.5$ and $UB_0 = 1$, we have

$$\rho_1 = \frac{(t+1)\cdot 1 + (t-1)\cdot 0.5}{2t} = \frac{3t+1}{4t}, \quad \alpha_1 = \frac{1-0.5}{t} = \frac{1}{2t}.$$

The interval of possible $\rho$ value is reduced by the ratio of $\frac{t+1}{2t}$ and thus, the value of $\alpha_k$ decreases by the factor of $\frac{t+1}{2t}$.

$$\alpha_1 = \frac{1}{2t},\ \alpha_2 = \frac{t+1}{(2t)^2},\ \alpha_3 = \frac{(t+1)^2}{(2t)^3},\ \ldots,\ \alpha_l = \frac{(t+1)^{l-1}}{(2t)^l} \geq \delta.$$

We have the number of iteration $l \leq \log_{\frac{2t}{t+1}}(\frac{1}{t+1}) + \log_{\frac{2t}{t+1}}(\frac{1}{\delta}) = \mathcal{O}(\log(\frac{1}{\delta}))$, for any fixed $t > 1$.

By the inequality (4.24), we have the number of linear segments $m_k$ in each iteration $k$ as

$$m_k \geq \frac{n}{4} + \frac{n}{4\sqrt{2\pi e}} \cdot \frac{1}{\alpha_k}.$$

That is

$$m_1 = \lceil \frac{n}{4} + \frac{tn}{2\sqrt{2\pi e}} \rceil,\ m_2 = \lceil \frac{n}{4} + \frac{2t}{t+1} \cdot \frac{tn}{2\sqrt{2\pi e}} \rceil,\ \ldots,\ m_l = \lceil \frac{n}{4} + (\frac{2t}{t+1})^{l-1} \cdot \frac{tn}{2\sqrt{2\pi e}} \rceil.$$

Now we compare the two bisection procedures. To compare the two bisection procedures, we clarify the number of the iterations and the number of the segments in each iteration of the two procedures. We call the two procedures as the original and the variation, and indicate as $O$ and $V$, respectively.

We first calculate the exact number of the iterations needed to guarantee the absolute error at most $\delta$ of PCO, where $\delta = t\alpha$ for a fixed $t > 1$ with the $\alpha$-approximation scheme of CCO.

Following the original procedure, in iteration $k$, we set $\rho_k$ as

$$\rho_k = \frac{UB_{k-1} + LB_{k-1} + \alpha}{2},$$

and this guarantees the reduction ratio of the length of the interval as

$$\beta_k = \frac{UB_{k-1} - LB_{k-1} + \alpha}{2(UB_{k-1} - LB_{k-1})}.$$

Thus, we have the length of the interval $l_k$ of the possible value of $\rho$ as

$$\begin{aligned}
l_0 &= UB_0 - LB_0, \\
l_1 &= (UB_0 - LB_0) \times \beta_1 = \frac{UB_0 - LB_0 + \alpha}{2}, \\
l_2 &= (UB_1 - LB_1) \times \beta_2 = \frac{\frac{UB_0 - LB_0 + \alpha}{2} + \alpha}{2} = \frac{UB_0 - LB_0 + (2^2 - 1)\alpha}{2^2}, \\
l_3 &= (UB_2 - LB_2) \times \beta_3 = \frac{\frac{UB_2 - LB_2 + \alpha}{2} + \alpha}{2} = \frac{UB_0 - LB_0 + (2^3 - 1)\alpha}{2^3}, \\
&\vdots \\
l_{I_O - 1} &= (UB_{I_O - 2} - LB_{I_O - 2}) \times \beta_{I_O - 1} = \frac{UB_0 - LB_0 + (2^{I_O - 1} - 1)\alpha}{2^{I_O - 1}} \geq \delta, \\
l_{I_O} &= (UB_{I_O - 1} - LB_{I_O - 1}) \times \beta_{I_O} = \frac{UB_0 - LB_0 + (2^{I_O} - 1)\alpha}{2^{I_O}} < \delta,
\end{aligned}$$

where $I_O$ is the total number of the iterations in the original procedure. Since we initialize as $LB_0 = 0.5$ and $UB_0 = 1$, $I_O$ can be obtained by the following.

$$\begin{aligned}
\frac{0.5 + (2^{I_O} - 1)\alpha}{2^{I_O}} < \delta \quad &\Leftrightarrow \quad 0.5 + (2^{I_O} - 1)\alpha < 2^{I_O} t\alpha \\
&\Leftrightarrow \quad 0.5 - \alpha < (t - 1)2^{I_O}\alpha \\
&\Leftrightarrow \quad \frac{1}{t - 1}\left(\frac{1}{2\alpha} - 1\right) < 2^{I_O} \\
&\Leftrightarrow \quad I_O > \log_2\left(\frac{1}{t - 1}\left(\frac{1}{2\alpha} - 1\right)\right).
\end{aligned}$$

Thus, the total number of the iterations in the original bisection procedure is

$$I_O = \left\lceil \log_2\left(\frac{1}{t - 1}\left(\frac{1}{2\alpha} - 1\right)\right)\right\rceil. \tag{4.26}$$

Now, we check the total number of the iterations required in the variation procedure to assure that the absolute error is at most $\delta$ for the PCO. In the iteration $k$, the $\rho_k$ is given as

$$\rho_k = \frac{(t+1)UB_{k-1} + (t-1)LB_{k-1}}{2t},$$

and the corresponding value of $\alpha_k$ is

$$\alpha_k = \frac{UB_{k-1} - LB_{k-1}}{t}. \tag{4.27}$$

In each iteration, we checked that the length of the interval is reduced by the factor of $\frac{t+1}{2t}$ than the length prior to the iteration. Thus, the length of the interval $l_k$ is

$$l_k = (UB_0 - LB_0) \times \left(\frac{t+1}{2t}\right)^k, \quad \forall k = 0, \dots, I_V \tag{4.28}$$

where $I_V$ is the number of the iterations in the variation procedure. Since the following two inequalities should be satisfied due to the ending criterion of the bisection procedure,

$$l_{I_V-1} = (UB_0 - LB_0) \times \left(\frac{t+1}{2t}\right)^{I_V-1} \geq \delta, \quad l_{I_V} = (UB_0 - LB_0) \times \left(\frac{t+1}{2t}\right)^{I_V} < \delta$$

and $LB_0 = 0.5$ and $UB_0 = 1$, we have

$$0.5 \times \left(\frac{t+1}{2t}\right)^{I_V} < t\alpha \quad \Leftrightarrow \quad I_V > \log_{\frac{2t}{t+1}}\left(\frac{1}{2t\alpha}\right).$$

Thus, we set the number of the iterations as

$$I_V = \left\lceil \log_{\frac{2t}{t+1}} \left( \frac{1}{2t\alpha} \right) \right\rceil. \tag{4.29}$$

Next, we compare the total number of segments, which affects the total number of DCOs to solve, that is necessary in the $\delta$-approximation of PCO. As mentioned, the number of the segments $m$ required for $\alpha$-approximation of CCO satisfies

$$m \geq \frac{\sqrt{2\pi e} + \frac{1}{\alpha}}{4\sqrt{2\pi e}} n.$$

Thus, we choose to have the number of segments as

$$m = \left\lceil \frac{\sqrt{2\pi e} + \frac{1}{\alpha}}{4\sqrt{2\pi e}} n \right\rceil.$$

Then, we can generate the number of segments in each iterations for the both of the bisection procedures.

The original procedure uses fixed value $\alpha$, and thus we have the number of the segments in each iteration $k$ as

$$m_k^O = \left\lceil \frac{\sqrt{2\pi e} + \frac{1}{\alpha}}{4\sqrt{2\pi e}} n \right\rceil, \quad \forall k = 1, \ldots, I_O. \tag{4.30}$$

On the other hand, the variation procedure uses different $\alpha_k$ given as (4.27). Since we know the length of the interval in each iteration is as (4.28), we have

$$\alpha_k = \frac{UB_{k-1} - LB_{k-1}}{t} = \frac{1}{t} \cdot (UB_0 - LB_0) \times \left( \frac{t+1}{2t} \right)^{k-1} = \frac{1}{2t} \times \left( \frac{t+1}{2t} \right)^{k-1}. \tag{4.31}$$

By applying (4.31), the number of the segments in each iteration $k$ is

$$m_k^V = \left\lceil \frac{\sqrt{2\pi e} + \frac{1}{\alpha_k}}{4\sqrt{2\pi e}} n \right\rceil = \left\lceil \frac{n}{4} + \frac{nt}{2\sqrt{2\pi e}} \times \left(\frac{2t}{t+1}\right)^{k-1} \right\rceil, \quad \forall k = 1, \dots, I_V. \quad (4.32)$$

Finally, we calculate the total number of segments required to obtain a $\delta$-approximation solution of PCO. Using the number of the iterations $I_O$ and $I_V$ given as (4.26) and (4.29), and the number of the segments in each iteration $k$, $m_k^O$ and $m_k^V$ as (4.30) and (4.32), respectively, the following is attained.

$$\begin{aligned} M_O = \sum_{k=1}^{I_O} m_k^O &= I_O \times \left\lceil \frac{\sqrt{2\pi e} + \frac{1}{\alpha}}{4\sqrt{2\pi e}} n \right\rceil \\ &\approx I_O \times \left( \frac{\sqrt{2\pi e} + \frac{1}{\alpha}}{4\sqrt{2\pi e}} n \right) = \left( \frac{I_O}{4} + \frac{I_O}{4\sqrt{2\pi e}} \cdot \frac{1}{\alpha} \right) \cdot n, \end{aligned}$$

and

$$\begin{aligned} M_V = \sum_{k=1}^{I_V} m_k^V &= \sum_{k=1}^{I_V} \left\lceil \frac{n}{4} + \frac{nt}{2\sqrt{2\pi e}} \times \left(\frac{2t}{t+1}\right)^{k-1} \right\rceil \\ &\approx \sum_{k=1}^{I_V} \left( \frac{n}{4} + \frac{nt}{2\sqrt{2\pi e}} \times \left(\frac{2t}{t+1}\right)^{k-1} \right) = \left( \frac{I_V}{4} + \frac{t}{2\sqrt{2\pi e}} \cdot \frac{\left(\frac{2t}{t+1}\right)^{I_V} - 1}{\frac{2t}{t+1} - 1} \right) \cdot n. \end{aligned}$$

We calculated the number of the iterations and the corresponding number of the total segments for some values of $t$ and $\alpha$. The results is given in Table 4.1 and Table 4.2. We can discover that the variation rule becomes better than the original rule as $t$ and $\alpha$ have larger values.

Table 4.1: Number of Iterations of the Bisection Procedures

| $t$ | 1.1 | | 1.3 | | 1.5 | | 2 | | 2.5 | | 3 | | 4 | | 5 | | 7 | | 10 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\alpha$ | $O$ | $V$ | $O$ | $V$ | $O$ | $V$ | $O$ | $V$ | $O$ | $V$ | $O$ | $V$ | $O$ | $V$ | $O$ | $V$ | $O$ | $V$ | $O$ | $V$ |
| 0.0001 | 16 | 182 | 15 | 68 | 14 | 45 | 13 | 28 | 12 | 22 | 12 | 19 | 11 | 16 | 11 | 14 | 10 | 12 | 10 | 11 |
| 0.0003 | 15 | 158 | 13 | 59 | 12 | 39 | 11 | 24 | 11 | 19 | 10 | 16 | 10 | 13 | 9 | 12 | 9 | 10 | 8 | 9 |
| 0.0005 | 14 | 147 | 12 | 55 | 11 | 36 | 10 | 22 | 10 | 17 | 9 | 15 | 9 | 12 | 8 | 11 | 8 | 9 | 7 | 8 |
| 0.001 | 13 | 132 | 11 | 49 | 10 | 32 | 9 | 20 | 9 | 15 | 8 | 13 | 8 | 11 | 7 | 10 | 7 | 8 | 6 | 7 |
| 0.005 | 10 | 97 | 9 | 36 | 8 | 24 | 7 | 14 | 7 | 11 | 6 | 9 | 6 | 7 | 5 | 6 | 5 | 5 | 4 | 4 |
| 0.01 | 9 | 83 | 8 | 30 | 7 | 20 | 6 | 12 | 6 | 9 | 5 | 7 | 5 | 6 | 4 | 5 | 4 | 4 | 3 | 3 |

Table 4.2: Number of Total Segments of the Bisection Procedures ($\times n$)

| $t$ | 1.1 | | 1.3 | | 1.5 | | 2 | | 2.5 | | 3 | | 4 | | 5 | | 7 | | 10 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\alpha$ | $O$ | $V$ | $O$ | $V$ | $O$ | $V$ | $O$ | $V$ | $O$ | $V$ | $O$ | $V$ | $O$ | $V$ | $O$ | $V$ | $O$ | $V$ | $O$ | $V$ |
| 0.0001 | 9683 | 13328 | 9078 | 5051 | 8473 | 3329 | 7868 | 2293 | 7263 | 1810 | 7263 | 1614 | 6657 | 1492 | 6657 | 1161 | 6052 | 934 | 6052 | 1063 |
| 0.0003 | 3029 | 4387 | 2625 | 1684 | 2423 | 1121 | 2221 | 729 | 2221 | 624 | 2019 | 481 | 2019 | 366 | 1818 | 419 | 1818 | 306 | 1616 | 322 |
| 0.0005 | 1698 | 2642 | 1455 | 1036 | 1334 | 652 | 1213 | 412 | 1213 | 307 | 1092 | 321 | 1092 | 230 | 970 | 252 | 970 | 175 | 849 | 178 |
| 0.001 | 790 | 1328 | 669 | 502 | 608 | 318 | 547 | 234 | 547 | 152 | 486 | 144 | 486 | 144 | 426 | 152 | 426 | 101 | 365 | 98 |
| 0.005 | 124 | 277 | 112 | 108 | 99 | 78 | 87 | 44 | 87 | 38 | 75 | 30 | 75 | 23 | 62 | 21 | 62 | 19 | 50 | 16 |
| 0.01 | 57 | 151 | 51 | 55 | 45 | 39 | 38 | 26 | 38 | 20 | 32 | 14 | 32 | 15 | 26 | 13 | 26 | 11 | 19 | 9 |

## 4.4 Comparison to the Approximation Scheme of Nikolova

Nikolova (2009) approximated the two problems that are closely related to PCO and CCO under a similar condition of ours. Recall the two problems that Nikolova (2009) suggested the approximation schemes of.

$$\begin{aligned} &\text{(N-PCO)} \quad \text{maximize} && \frac{b - \sum_{j \in N} \mu_j x_j}{\sqrt{\sum_{j \in N} \sigma_j^2 x_j}} \\ &\qquad\qquad\quad \text{subject to} && x \in \mathcal{F}, \end{aligned}$$

and

$$\begin{aligned} &\text{(N-CCO)} \quad \text{minimize} && \sum_{j \in N} \mu_j x_j + \Phi^{-1}(\rho) \sqrt{\sum_{j \in N} \sigma_j^2 x_j} \\ &\qquad\qquad\quad \text{subject to} && x \in \mathcal{F}, \end{aligned}$$

where $\mathcal{F} \subseteq \mathbb{R}^n$. By setting $\mathcal{F}$ as

$$\mathcal{F} = \left\{ \sum_{j \in N} p_j x_j \geq f, x \in X \right\},$$

the approximation solution of N-PCO satisfies

$$(1-\epsilon) z^* \leq \frac{b - \sum_{j \in N} \mu_j x_j}{\sqrt{\sum_{j \in N} \sigma_j^2 x_j}}, \ x \in X \quad \Rightarrow \quad Pr \left( z \leq (1-\epsilon) \frac{b - \sum_{j \in N} \mu_j x_j}{\sqrt{\sum_{j \in N} \sigma_j^2 x_j}} \right) \geq \rho^*, \ x \in X,$$

where $z^*$ is the objective value of the given solution of N-PCO and $\rho^*$ is the corresponding probability value of $Pr\left(z \leq z^*\right)$ with $z \sim N(0,1)$. Similarly, the approximation solution of N-CCO satisfies

$$
\begin{aligned}
& (1+\epsilon)z^* \geq \sum_{j \in N} \mu_j x_j + \Phi^{-1}(\rho)\sqrt{\sum_{j \in N} \sigma_j^2 x_j},\ x \in X \\
\Rightarrow \quad & Pr\left(\sum_{j \in N} a_j x_j \leq (1+\epsilon)b\right) \geq \rho^*,\ x \in X,
\end{aligned}
$$

where $z^*$ is the objective value of N-CCO and $\rho^*$ is the corresponding probability value of $Pr\left(\sum_{j \in N} a_j x_j \leq z^*\right)$.

On the other hand, both approximation solutions of our approximation schemes of PCO and CCO satisfy

$$
Pr\left(\sum_{j \in N} a_j x_j \leq b\right) \geq \rho - \delta,\ x \in X,
$$

which implies that our solution guarantees the absolute error of the probability of the solution.

The comparison between the solution of CCO attained by our approximation scheme and the solution of N-CCO attained by that of Nikolova is not suitable. However, we can compare the performances of the approximation schemes of PCO and N-PCO by adjusting the value of $\delta$ and $\epsilon$, the approximation parameter of our scheme and that of Nikolova respectively, for some values of $\rho$. First consider the complexities of the two approximation schemes. Assume $p_j \in \mathbb{Z}$ and $w_j \in \mathbb{R}$ for all $j \in N$, and $U = \sum_{j \in N} |p_j|$ and $b = \sum_{j \in N} |w_j|$. Suppose that there is an algorithm of solving

$$
\begin{aligned}
\min \quad & \sum_{j\in N} w_j x_j \\
\text{s.t.} \quad & \sum_{j\in N} p_j x_j \geq f, \\
& x \in X,
\end{aligned}
$$

with the algorithm complexity $\mathcal{O}(g(n,b,U))$. Then the complexities of the approximation schemes of CCO and PCO for our study and Nikolova (2009) are given as the following.

Table 4.3: Comparison of the Complexities of the Approximation Schemes

| Complexity | Ours | Nikolova |
|---|---|---|
| (N-)CCO | $\mathcal{O}\left(n^2\frac{1}{\delta}Ug(n,b,U)\right)$ | $\mathcal{O}\left((1+\frac{1}{\epsilon}\log(\frac{f_u}{f_l}))(1+\frac{\log(\frac{1}{\epsilon^2})}{\log(1+\epsilon)})g(n,b,U)\right)^{\text{a}}$ |
| (N-)PCO | $\mathcal{O}\left(n^2\frac{1}{\delta}\log(\frac{1}{\delta})Ug(n,b,U)\right)$ | $\mathcal{O}\left(\log(\frac{s_{\max}}{s_{\min}})\log(\frac{f_u}{f_l})\frac{1}{\epsilon^2}g(n,b,U)\right)^{\text{b}}$ |

[a] $s_{\max}$, $s_{\min}$ : the maximum and minimum values of $\sum_{j\in N}\sigma_j^2 x_j$ for $x\in\mathcal{F}$

[b] $f_u$, $f_l$ : the maximum and minimum values of $\sum_{j\in N}\mu_j x_j + \Phi^{-1}\sqrt{\sum_{j\in N}\sigma_j^2 x_j}$ for $x\in\mathcal{F}$

For some fixed values of $\rho$, we can calculate the value of $\epsilon$ that gives the equivalent approximation solution to the value of each $\delta$. By this procedure, we can compare the computation time of two approximation schemes. Tabel 4.4 shows the complexities of two approximation schemes, of which the logarithmic terms of the complexity of the scheme of Nikolova (2009) are ignored.

Table 4.4: Computation Times of the Approximation Schemes for values of $\rho$

| $\rho$ | $z$ | $\delta$ | $\epsilon$ | Ours | Nikolova |
|---|---|---|---|---|---|
| 0.55 | 0.1257 | $10^{-2}$ | 0.2013 | $\mathcal{O}\left(100n^2U \cdot g(n,b,U)\right)$ | $\mathcal{O}\left(25 \cdot g(n,b,U)\right)$ |
| | | $10^{-3}$ | 0.0207 | $\mathcal{O}\left(1000n^2U \cdot g(n,b,U)\right)$ | $\mathcal{O}\left(2337 \cdot g(n,b,U)\right)$ |
| | | $10^{-4}$ | 0.0024 | $\mathcal{O}\left(10000n^2U \cdot g(n,b,U)\right)$ | $\mathcal{O}\left(175561 \cdot g(n,b,U)\right)$ |
| 0.65 | 0.3853 | $10^{-2}$ | 0.0696 | $\mathcal{O}\left(100n^2U \cdot g(n,b,U)\right)$ | $\mathcal{O}\left(207 \cdot g(n,b,U)\right)$ |
| | | $10^{-3}$ | 0.0070 | $\mathcal{O}\left(1000n^2U \cdot g(n,b,U)\right)$ | $\mathcal{O}\left(20364 \cdot g(n,b,U)\right)$ |
| | | $10^{-4}$ | 0.0005 | $\mathcal{O}\left(10000n^2U \cdot g(n,b,U)\right)$ | $\mathcal{O}\left(3711402 \cdot g(n,b,U)\right)$ |
| 0.75 | 0.6745 | $10^{-2}$ | 0.0463 | $\mathcal{O}\left(100n^2U \cdot g(n,b,U)\right)$ | $\mathcal{O}\left(467 \cdot g(n,b,U)\right)$ |
| | | $10^{-3}$ | 0.0047 | $\mathcal{O}\left(1000n^2U \cdot g(n,b,U)\right)$ | $\mathcal{O}\left(44429 \cdot g(n,b,U)\right)$ |
| | | $10^{-4}$ | 0.0004 | $\mathcal{O}\left(10000n^2U \cdot g(n,b,U)\right)$ | $\mathcal{O}\left(5055003 \cdot g(n,b,U)\right)$ |
| 0.85 | 1.0364 | $10^{-2}$ | 0.0404 | $\mathcal{O}\left(100n^2U \cdot g(n,b,U)\right)$ | $\mathcal{O}\left(611 \cdot g(n,b,U)\right)$ |
| | | $10^{-3}$ | 0.0041 | $\mathcal{O}\left(1000n^2U \cdot g(n,b,U)\right)$ | $\mathcal{O}\left(60891 \cdot g(n,b,U)\right)$ |
| | | $10^{-4}$ | 0.0004 | $\mathcal{O}\left(10000n^2U \cdot g(n,b,U)\right)$ | $\mathcal{O}\left(6713281 \cdot g(n,b,U)\right)$ |
| 0.95 | 1.6449 | $10^{-2}$ | 0.0548 | $\mathcal{O}\left(100n^2U \cdot g(n,b,U)\right)$ | $\mathcal{O}\left(333 \cdot g(n,b,U)\right)$ |
| | | $10^{-3}$ | 0.0059 | $\mathcal{O}\left(1000n^2U \cdot g(n,b,U)\right)$ | $\mathcal{O}\left(28756 \cdot g(n,b,U)\right)$ |
| | | $10^{-4}$ | 0.0006 | $\mathcal{O}\left(10000n^2U \cdot g(n,b,U)\right)$ | $\mathcal{O}\left(2705696 \cdot g(n,b,U)\right)$ |

We can see that the tendency of the relative complexity of our scheme to that of Nikolova (2009) gets better as the $\rho$ value increases and the delta decreases. Additionally, we propose the following proposition of the relation between $\epsilon$ and $\delta$.

**Proposition 4.7.** *As $\epsilon \to 0$, $\epsilon$ is nearly linear to $\delta$ which gives the equivalent approximation solution.*

*Proof.* Suppose that $t > 0$ is given. Then

$$\begin{aligned}
\delta &= Pr\left((1-\epsilon)t \le z \le t\right) \\
&= \int_{(1-\epsilon)t}^{t} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2} dx \\
&= \frac{1}{\sqrt{2\pi}} \left( x - \frac{x^3}{6} + \frac{x^5}{40} - \frac{x^7}{336} + \cdots \right) |_{(1-\epsilon)t}^{t} \quad \text{(Taylor expansion)} \\
&\approx \frac{1}{\sqrt{2\pi}} \epsilon t, \quad \text{if } \epsilon \ll 1.
\end{aligned}$$

□

# Chapter 5

# Conclusion

## 5.1 Concluding Remarks

In this thesis, we studied the combinatorial optimization problems with uncertainty from the stochastic optimization point of view. Specifically, we assumed the uncertain data follows normal distributions without correlation. We defined two general form of the stochastic optimization problem using the probability of satisfying the constraint of the deterministic combinatorial problem (DCO), which are the probability maximizing combinatorial optimization (PCO) and the chance-constrained combinatorial optimization (CCO). We proved that both PCO and CCO are $\mathcal{NP}$-hard in general, and additionally analyzed the complexities of them in the special cases. In particular, we found out that PCO and CCO can be $\mathcal{NP}$-hard, even in the condition that DCO is polynomial time solvable. We also proposed exact algorithms of PCO and CCO that attain the exact solutions by iteratively solving DCO with an additional capacity constraint. Moreover, we derived an approximation scheme of PCO, and also an approximation scheme of CCO that is required in the process of the approximation of PCO. Both the approximation schemes repeatedly solve DCO and obtain a solution that guarantees the absolute error of the probability to be less than a given threshold. Especially, the approximation schemes are fully polynomial time approximation scheme (FPTAS) for both PCO and CCO in some special cases that we proved to be $\mathcal{NP}$-hard in Chapter 2. Furthermore, variation of the bisection procedure in the approximation scheme of PCO has been provided, with tables comparing the approximated number of total iterations and the number of times solving DCO of the two bisection procedures. Finally, we analyzed the difference between the approximation scheme of Nikolova (2009) and our scheme, and compared the computational times of the equivalent approximation for some $\rho$ values.

## 5.2 Future Works

As the future works, the experimental study is required to compare the practical computation times of the algorithms. Since the algorithmic complexities of the exact algorithm and the approximation scheme depend on different parameters, various types of instances should be tested. The computational time of the approximation scheme with the two bisection procedures also need to be compared. Additionally, the complexity analysis for special cases of PCO and CCO other than the knapsack problem or the shortest path problem has not been done much yet. Thus, not much of the complexity of the special cases are known and more researches are required. Furthermore, the complexities of PCO and CCO in the case of $X = \mathbb{B}^n$ with values of $p_j$ for $j \in N$ polynomially bounded by $n$ is unknown. Thus, the complexity of this case is an open question. Moreover, there is a possibility of other algorithms that solve PCO and CCO exactly or approximately with better algorithm complexity. There can be faster algorithms that deal with the special cases of PCO and CCO with some assumptions of the problems. By confining the problem to the knapsack problem, shortest path problem, or other well known combinatorial optimization problems, there can be algorithms with better performance.

Further extensions of the research include the variations of the probability distributions of the data with uncertainty. We assumed that $a_j$ for all $j \in N$ follows a normal distribution that is independent to the other values. However, we can assume that there are correlations between the values of $a_j$, $j \in N$, which is at least as hard as the problem without correlation. Else, other probability distribution can be assumed to the uncertain data rather than the normal distribution. Lastly, the study of combinatorial optimization problems that has more than one type of data with uncertainty can be a future extension direction.

# Bibliography


Atamtürk, A., Muller, L. F., and Pisinger, D. (2013). Separation and extension of cover inequalities for conic quadratic knapsack constraints with generalized upper bounds. *INFORMS Journal on Computing*, 25(3):420–431.

Atamtürk, A. and Narayanan, V. (2009). The submodular knapsack polytope. *Discrete Optimization*, 6(4):333–344.

Bellman, R. (1958). On a routing problem. *Quarterly of Applied Mathematics*, 16(1):87–90.

Ben-Tal, A., El Ghaoui, L., and Nemirovski, A. (2009). *Robust optimization*, volume 28. Princeton University Press.

Bertsimas, D. and Sim, M. (2003). Robust discrete optimization and network flows. *Mathematical Programming*, 98(1-3):49–71.

Birge, J. R. and Louveaux, F. (2011). *Introduction to stochastic programming*. Springer Science & Business Media.

Cheng, J. and Lisser, A. (2015). Maximum probability shortest path problem. *Discrete Applied Mathematics*, 192:40–48.

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271.

Dinh, T., Fukasawa, R., and Luedtke, J. (2018). Exact algorithms for the chance-constrained vehicle routing problem. *Mathematical Programming*, 172(1-2):105–138.

Gabrel, V. and Minoux, M. (2002). A scheme for exact separation of extended cover inequalities and application to multidimensional knapsack problems. *Operations Research Letters*, 30(4):252–264.

Goyal, V. and Ravi, R. (2010). A ptas for the chance-constrained knapsack problem with random item sizes. *Operations Research Letters*, 38(3):161–164.

Han, J., Lee, K., Lee, C., Choi, K.-S., and Park, S. (2016). Robust optimization approach for a chance-constrained binary knapsack problem. *Mathematical Programming*, 157(1):277–296.

Holzhauser, M., Krumke, S. O., and Thielen, C. (2016). Budget-constrained minimum cost flows. *Journal of Combinatorial Optimization*, 31(4):1720–1745.

Ilyina, A. (2017). *Combinatorial optimization under ellipsoidal uncertainty*. PhD thesis, Technische Universität Dortmund.

Isbell, J. and Marlow, W. (1956). Attrition games. *Naval Research Logistics (NRL)*, 3(1-2):71–94.

Ji, X. (2005). Models and algorithm for stochastic shortest path problem. *Applied Mathematics and Computation*, 170(1):503–514.

Joung, S. and Lee, K. (2018). Robust optimization-based heuristic algorithm for the chance-constrained knapsack problem using submodularity. *arXiv preprint arXiv:1811.01569*.

Karp, R. M. (1972). Reducibility among combinatorial problems. In *Complexity of computer computations*, pages 85–103. Springer.

Kasperski, A. and Zieliński, P. (2016). Robust discrete optimization under discrete and interval uncertainty: A survey. In *Robustness analysis in decision aiding, optimization, and analytics*, pages 113–143. Springer.

Kellerer, H., Pferschy, U., and Pisinger, D. (2004). Multidimensional knapsack problems. In *Knapsack problems*, pages 235–283. Springer.

Klopfenstein, O. and Nace, D. (2008). A robust approach to the chance-constrained knapsack problem. *Operations Research Letters*, 36(5):628–632.

Lee, C., Lee, K., and Park, S. (2012). Robust vehicle routing problem with deadlines and travel time/demand uncertainty. *Journal of the Operational Research Society*, 63(9):1294–1306.

Monaci, M., Pferschy, U., and Serafini, P. (2013). Exact solution of the robust knapsack problem. *Computers & Operations Research*, 40(11):2625–2631.

Nikolova, E. (2008). Stochastic combinatorial optimization with risk. Technical report, Computer Science and Artificial Intelligence Laboratory, MIT.

Nikolova, E., Kelner, J. A., Brand, M., and Mitzenmacher, M. (2006). Stochastic shortest paths via quasi-convex maximization. In *European Symposium on Algorithms*, pages 552–563. Springer.

Nikolova, E. V. (2009). *Strategic algorithms*. PhD thesis, Massachusetts Institute of Technology.

Pisinger, D. (2005). Where are the hard knapsack problems? *Computers & Operations Research*, 32(9):2271–2284.

Pisinger, D. and Toth, P. (1998). Knapsack problems. In *Handbook of combinatorial optimization*, pages 299–428. Springer.

Poss, M. (2018). Robust combinatorial optimization with knapsack uncertainty. *Discrete Optimization*, 27:88–102.

Shabtai, G., Raz, D., and Shavitt, Y. (2018). A relaxed fptas for chance-constrained knapsack. In *29th International Symposium on Algorithms and Computation (ISAAC 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Shapiro, A., Dentcheva, D., and Ruszczyński, A. (2009). *Lectures on stochastic programming: modeling and theory*. SIAM.

Toth, P. and Vigo, D. (2002). *The vehicle routing problem*. SIAM.

Warburton, A. (1987). Approximation of pareto optima in multiple-objective, shortest-path problems. *Operations research*, 35(1):70–79.

Wolsey, L. A. and Nemhauser, G. L. (2014). *Integer and combinatorial optimization*. John Wiley & Sons.

Yang, F. and Chakraborty, N. (2018). Algorithm for optimal chance constrained knapsack problem with applications to multi-robot teaming. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1043–1049. IEEE.

Ziegelmann, M. and Mehlhorn, K. (2001). *Constrained Shortest Paths and Related Problems*. PhD thesis, Universität des Saarlandes Saarbrücken.

# 국문초록


본 논문에서는 일반적인 조합 최적화 문제(deterministic combinatorial optimization problem : DCO)에서 데이터의 불확실성이 존재할 때를 다루는 문제로, 총 수익을 주어진 상수 이상으로 보장하면서 용량 제약을 만족시킬 확률을 최대화하는 확률 최대화 조합 최적화 문제(probability maximizing combinatorial optimization problem : PCO)을 다룬다. PCO와 매우 밀접한 관계가 있는 문제로, 총 수익을 최대화하면서 용량 제약을 만족시킬 확률이 일정 값 이상이 되도록 보장하는 확률 제약 조합 최적화 문제(chance-constrained combinatorial optimization problem : CCO)가 있다. 우리는 두 문제의 관계에 대하여 논의하고 특정 조건 하에서 두 문제의 복잡도를 분석하였다. 또한, 제약식이 하나 추가된 DCO를 반복적으로 풀어 PCO와 CCO의 최적해를 구하는 유사 다항시간 알고리즘을 제안하였다. 더 나아가, PCO가 $\mathcal{NP}$-hard인 특별한 인스턴스들에 대해서 완전 다항시간 근사해법(FPTAS)가 되는 근사해법을 제안하였다. 이 근사해법을 유도하는 과정에서 CCO의 근사해법 또한 고안하였다.



**주요어**: 불확실성을 고려한 조합 최적화, 확률 최대화 조합 최적화 문제, 확률 제약 조합 최적화 문제, 이분법

**학번**: 2017-29219