Robust Multicovers: Algorithms and Complexity

Eva Maria Schmidt

Vom Fachbereich Mathematik
der Technischen Universität Kaiserslautern
zur Verleihung des akademischen Grades
Doktor der Naturwissenschaften
(Doctor rerum naturalium, Dr. rer. nat.)
genehmigte Dissertation

Gutachter/innen:	Prof. Dr. Sven O. Krumke
	Prof. Dr. Frauke Liers
Datum der Disputation:	03. September 2021

D 386

TECHNISCHE UNIVERSITÄT
KAISERSLAUTERN

Danksagung

Zuallererst möchte ich mich bei meinem Betreuer Prof. Dr. Sven O. Krumke bedanken, insbesondere für seine Unterstützung und Motivation sowie das jederzeit offene Ohr. Ohne seine Kreativität und Ideen wäre diese Arbeit in ihrer aktuellen Form nicht möglich gewesen.

Weiterhin danke ich Prof. Dr. Frauke Liers für ihre Bereitschaft und Mühe als Zweitgutachterin dieser Arbeit. Ein besonderer Dank geht auch an Manuel Streicher für die tolle gemeinsame Arbeit in den Projekten *HealthFaCT* und *One Plan*, die letztendlich zu vielen Fragestellungen dieser Arbeit geführt hat. Neben Sven O. Krumke danke ich Lena Leiß für unsere Zusammenarbeit im Projekt *GraThO*, die uns allen noch lange in Erinnerung bleiben wird.

Ein weiterer Dank gilt den Co-Autorinnen und Co-Autoren der Veröffentlichungen, die im Zusammenhang mit dieser Arbeit entstanden sind: Christina Büsing, Martin Comis, Sven O. Krumke und Manuel Streicher.

Ebenso bedanke ich mich bei allen Korrekturleserinnen und -lesern für ihre Anmerkungen und Vorschläge, von denen diese Arbeit entscheidend profitiert hat: Oliver Bachtler, Tim Bergner, Till Heller, Lena Leiß, Sebastian Johann, Sven O. Krumke und Manuel Streicher. Auch danke ich Andrea Maier für ihre konstruktiven Vorschläge, die in diese Arbeit eingeflossen sind.

Ich bin sehr froh ein Teil dieser tollen Arbeitsgruppe sein zu dürfen und bin dankbar für die vielen hilfreichen Beiträge und Diskussionen – insbesondere im Doktorandentreffen.

Nicht zuletzt möchte ich meiner Familie und meinen Freundinnen und Freunden Danke sagen, die mich auch außerhalb dieser Arbeit jederzeit unterstützt haben und auf die ich mich immer verlassen konnte.

Die Forschung zu dieser Arbeit wurde in Teilen finanziert durch das Bundesministerium für Bildung und Forschung im Rahmen des Projekts *HealthFaCT* (Förderkennzeichen 05M16UKC), das Ministerium des Innern und für Sport Rheinland-Pfalz im Rahmen des Projekts *One Plan* und den Deutschen Akademischen Austauschdienst im Rahmen des Projekts *GraThO*.

Zusammenfassung

Das große Interesse an robusten Überdeckungsproblemen ist vielfältig, besonders durch die Vielzahl realer Anwendungsmöglichkeiten und die zusätzliche Betrachtung von Unsicherheiten, die vielen praktischen Problemen innewohnen.

In dieser Arbeit stellen wir ein neues robustes Überdeckungsproblem vor, das wir *Robust Min q-Multiset Multicover* nennen, wobei q eine fixe natürliche positive Zahl darstellt. Dieses und weitere verwandte Probleme werden sorgfältig ausgearbeitet. Die gemeinsame Idee dieser Probleme ist es, bei gegebener Auswahl an Teilmengen einer Grundmenge, die Auswahlhäufigkeit jeder Teilmenge so zu bestimmen, dass die unsichere Nachfrage aller vorkommenden Elemente erfüllt ist. Im Unterschied zu allgemeinen Überdeckungsproblemen darf hier jede Teilmenge höchstens q ihrer Elemente überdecken. Durch Variation der Eigenschaften der vorkommenden Elemente entstehen so vier interessante robuste Überdeckungsprobleme, die untersucht werden.

Wir analysieren ausführlich die Komplexität dieser Probleme. Dabei betrachten wir auch Einschränkungen in spezielle Klassen von Unsicherheitsmengen. Für ein gegebenes Problem geben wir entweder einen Polynomialzeit-Algorithmus an oder zeigen, dass es, solange nicht $\mathrm{P} = \mathrm{NP}$ gilt, einen solchen Algorithmus nicht geben kann. Weiterhin beweisen wir für den Großteil der Fälle sogar, dass aller Voraussicht nach auch ein polynomielles Approximationsschema für die schweren Varianten nicht möglich ist.

Außerdem streben wir nach Approximationen und Approximationsalgorithmen für diese schweren Varianten. Hier fokussieren wir uns auf Robust Min q-Multiset Multicover. Für eine umfangreiche Klasse von Unsicherheitsmengen präsentieren wir den ersten Polynomialzeit--Approximationsalgorithmus für Robust Min q-Multiset Multicover mit beweisbarer Güte.

Abstract

The great interest in robust covering problems is manifold, especially due to the plenitude of real world applications and the additional incorporation of uncertainties which are inherent in many practical issues.

In this thesis, for a fixed positive integer q, we introduce and elaborate on a new robust covering problem, called *Robust Min q-Multiset Multicover*, and related problems. The common idea of these problems is, given a collection of subsets of a ground set, to decide on the frequency of choosing each subset so as to satisfy the uncertain demand of each overall occurring element. Yet, in contrast to general covering problems, the subsets may only cover at most q of their elements. Varying the properties of the occurring elements leads to a selection of four interesting robust covering problems which are investigated.

We extensively analyze the complexity of the arising problems, also for various restrictions to particular classes of uncertainty sets. For a given problem, we either provide a polynomial time algorithm or show that, unless $\mathrm{P} = \mathrm{NP}$, such an algorithm cannot exist. Furthermore, in the majority of cases, we even give evidence that a polynomial time approximation scheme is most likely not possible for the hard problem variants.

Moreover, we aim for approximations and approximation algorithms for these hard variants, where we focus on Robust Min q-Multiset Multicover. For a wide class of uncertainty sets, we present the first known polynomial time approximation algorithm for Robust Min q-Multiset Multicover having a provable worst-case performance guarantee.

Contents

List of Figures

List of Tables

1. Introduction

This thesis studies particular robust covering problems. In this chapter, we motivate and summarize its main content.

Motivation

Informally, an instance of a covering problem comprises a collection of subsets of a finite ground set and a coverage requirement for every element of the ground set. The task is to choose a selection of subsets in order to fulfill all requirements. Problems of this type arise in plenty of real world applications and, hence, the need for efficient algorithms and well-performing approximations is ubiquitous. Therefore, there has been a lot of research regarding this area of optimization and the studies are still on-going. Most commonly, the developed algorithms exploit the specific structure of the considered covering problem. Thus, when introducing a novel problem of this type, we should integrate the problem into the present classification scheme and survey the adaptability of available algorithms. Even if the available algorithms are not applicable directly, we might still be able to extract ideas for the development of algorithms precisely fitting our new problem.

The classical Set Cover problem is listed in Karp's famous initial list of 21 NP-complete decision problems, cf. [Kar72]. An instance of this problem consists of a collection of subsets of a finite ground set and a positive integer $B \in \mathbb{N}_{>0}$. With such an instance, the question to answer is whether there exists a cover of the ground set of size at most B, i.e., a selection of at most B subsets such that every element of the ground set appears in at least one selected set. By Karp's result there is no polynomial time algorithm to solve the corresponding optimization problem Min Set Cover, unless $\mathrm{P} = \mathrm{NP}$. Hence, we need to make use of potentially exponential time algorithms to solve every given instance to optimality. The same holds true for every generalization of this elementary problem. In many practical applications today, optimality is a goal in farther distance as the sizes of the instances are enormous. Then, the computation of approximate solutions, which guarantee the obtained solution to attain a solution value located within a ratio of the optimal value, is of particular interest.

Moreover, in practice there are various sources of uncertainty that can affect the performance of a proposed solution, possibly in an undesired direction. On the one hand, the mathematical covering problem may not exactly mirror the practical issue. In this case, adjustments or refinements of the theoretical model are necessary to decrease the degree of uncertainty. On the other hand, the practical problem itself may contain inherent uncertainties, e.g., parameters depending on insecurities in measurements or on unknown future events. For the latter case, there are two mathematical fields handling parameter uncertainties in optimization. If the probability distributions of the uncertain parameters are – potentially only partly – known, stochastic optimizers include this additional information into the model and aim to optimize a certain stochastic measure, for instance, the expected value. On the contrary, the area of robust optimization does not assume a specific probability distribution for each uncertain parameter and is unbiased in that sense. It accounts for the uncertainty by determining a distribution-free set of possible realizations for each parameter. Now, by its worst-case oriented philosophy, the goal is to find a solution that takes into account any variation of the parameters within their prescribed sets. Furthermore, the solution should not behave too badly in case the combination of realizations of the uncertain parameters is disadvantageous. Naturally, there is a higher price to pay when implementing a robust solution as it is uncertainty-immunized and therefore less vulnerable.

Hence, in this thesis, we introduce and elaborate on a new robust covering problem named *Robust Min q-Multiset Multicover* for a fixed integer $q \in \mathbb{N}_{>0}$. It has applications in the strategic location and resource planning of emergency facilities. Compared to other robust covering problems, the distinctive attribute is that the sets of an instance are only implicitly given and their cardinalities are bounded by q. This leads to a decrease in the instance sizes which especially influences the complexity propositions and the design of (approximation) algorithms. Moreover, considering the robust setting we allow to switch the selection of sets between different realizations of the parameters to some extent. This results in a more flexible robust covering model at a lower price of robustness. Altogether, Robust Min q-Multiset Multicover integrates itself nicely into the field of robust covering problems and leads to interesting research questions concerning algorithms and complexity.

Literature Review

Already the classical (non-robust) Min Set Cover problem emerges in many scientific fields, e.g., optimization, computer science, and operations research. By its elementary structure, there is a plenitude of real world applications based on the Min Set Cover problem. Exemplarily, we mention computer virus detection, crew scheduling, and diagnostic systems, see [Wil98; CNS98; Vem98]. From the theoretical perspective, the problem unifies many individual problems, especially in the field of graph theory, e.g., Min Vertex Cover, Min Edge Cover, and Min Graph Coloring [GJ79; HLS09].

In order to capture even more specific issues, more complex extensions of the problem have been formulated and studied. These generalizations include multisets, set costs, integral coverage requirements (demand) of the elements in the ground set, or the option to choose subsets multiple times [Vaz03; KY05]. Possible applications of this setup involve placing service facility locations and increasing the reliability of communication networks [Vem98; KP16].

With the challenge of uncertainty in the input data, stochastic as well as robust Min Set Cover problems have gained more interest in recent years. From the robust perspective, the common idea is to select a cover that is feasible for the considered problem for every realization of the uncertainty. That means we aim to hedge against the uncertainty. In literature, there are various studies on probabilistic and robust covering problems. They mainly differ in terms of the applied uncertainty concept. There are three parameter types whose uncertainty is currently studied: the set costs, the set availability, and the demand of the elements of the ground set.

Uncertainty in the objective is considered in [PA13]. Therein, the Min Set Cover problem with interval uncertainty in the cost coefficients is analyzed and exact algorithms for computing a min-max regret solution are presented.

Uncertain set availability is covered in [HCL04] and [FM12]. In [HCL04], the authors apply concepts of fuzzy set theory, where each element is contained in a set only with a certain probability. They define a fuzzy Min Set Cover problem which is subsequently reduced to a non-linear integer programming problem. In [FM12], it is assumed that each set has a certain probability to disappear while, on the other hand, each element needs to be covered with probability at least a given threshold. The authors study the problem from a polyhedral point of view by presenting compact as well as cutting-plane formulations. Further, in [Lut+17], a robust version of the problem introduced in [FM12] is considered using Γ-robustness [BS04]. Various formulations are derived and the approach is applied to the problem of placing emergency service facilities.

Demand uncertainty, which is also called right-hand side uncertainty, is studied in various aspects. In [BR02], the requirement of covering an element is modeled using a binary random variable. Then, joint probabilistic constraints are introduced to account for the occurring uncertainty. Based on this, equivalent MIP formulations for this problem are given in [SGL10]. In [GNR14], the authors study another robust variant of the Min Set Cover problem, where each scenario specifies a subset of the ground set of a certain fixed size to be covered. They provide approximation algorithms for robust two-stage problems: Some of the sets may be selected in a first stage at lower cost and in a second stage, after the scenario is known, the remaining sets are chosen. Originally, this problem was introduced in [Dha+05] in a more general fashion. Therein, the authors give approximation algorithms for several robust versions of well-known combinatorial problems, e.g., Min Cut. Further, in [Fei+07], these results are extended to an exponential number of scenarios and approximation algorithms are developed using an online algorithm for Min Set Cover, cf. [Alo+09; BN05], within an LP-rounding-based algorithm.

Outline and Contribution

In this thesis, we investigate a new robust covering problem called Robust Min q-Multiset Multicover and related problems. We contribute extensive complexity analyses, also for restrictions of these problems. For a given problem, we either provide a strongly polynomial time algorithm or show that, unless the complexity classes P and NP coincide, there is no polynomial time algorithm. Moreover, in most cases, even the existence of a polynomial time approximation scheme, i.e., a polynomial time algorithm that is capable of providing arbitrarily good approximations, is most-likely excluded for the hard problem variants. Furthermore, we aim to provide approximations and approximation algorithms for these hard variants, in particular for Robust

Min q-Multiset Multicover. For a wide class of uncertainty sets, we give the first known polynomial time approximation algorithm for Robust Min q-Multiset Multicover with provable worst-case performance guarantee.

The outline of this thesis is as follows. First of all, Chapter 2 comprises our notational conventions and various overviews on specific mathematical fields that are applied throughout this thesis.

In Chapter 3, we introduce the Min q-Multiset Multicover (Min q-MSMC) problem for a fixed positive integer $q \in \mathbb{N}_{>0}$. An instance of this optimization problem consists of a finite ground set, an integral demand value for every element, and a collection of subsets from this ground set. The goal is to choose a minimum number of subsets, where multiple choices of a subset are allowed, such that the demand of every element is covered. Additionally, we require that each chosen subset may only cover up to q of its elements, where again multiple choices are allowed. We show that Min q-MSMC is APX-complete for any fixed $q \geq 3$. For Min 1-MSMC and Min 2-MSMC, we present strongly polynomial time algorithms.

The story is quite different for the robust version of this problem, called Robust Min q-Multiset Multicover (Robust Min q-MSMC), which forms the main focus of this thesis. In an instance of this robust problem, the demand of the elements is replaced by a set of scenarios $\mathcal{U}$ – the uncertainty set – containing various possible demand vectors. Now, we aim to choose as few subsets as possible from our given collection, possibly multiple times, such that, for every scenario, the selection of sets is a solution to the corresponding instance of Min q-Multiset Multicover where the necessary demand vector is given by the scenario. We prove that Robust Min q-MSMC is APX-hard for any fixed value of q. Further, we show that certifying feasibility of a given tentative solution, a task which is trivial for the non-robust version of the problem, constitutes a hard problem in the robust case. The representation and the handling of the uncertainty present additional challenges in designing solution methods for Robust Min q-MSMC. Due to our hardness results, we provide various integer programming formulations for the non-robust as well as the robust problem which allow us to design exact solution algorithms. It turns out that, especially for the robust problem, these formulations reveal constraint generation to be a promising solution approach. For this approach, the above mentioned feasibility problem is of great interest. Hence, when restricting Robust Min q-MSMC to various classes of uncertainty sets, we investigate the complexity of the problem itself and of its particular feasibility problem. We consider discrete, interval, budgeted, multi-budgeted, ellipsoidal, and Γ-uncertainty. For instance, for discrete uncertainty, we show that Robust Min q-MSMC restricted to instances with $q \cdot |\mathcal{U}| \geq 3$ is APX-complete. On the other hand, instances with $q \cdot |\mathcal{U}| \leq 2$ are polynomial time solvable.

The exhaustive complexity analysis of Robust Min q-MSMC suggests the development of approximations and, in particular, approximation algorithms for this problem. This constitutes the purpose of Chapter 4. First of all, we concentrate on approximate solutions that arise from the problem's close relation to Min Adjustable Robust Covering problems and are widely applied. We focus on common solution policies, e.g., the strict and the affine policy, and analyze their performances in relation to an optimal solution. It turns out that these policies are able to yield constant factor approximations for special cases, but the respective computation of these solutions forms NP-hard problems. Moreover, we show how to extend the ideas of an approximation algorithm for the Multiset Multicover problem to obtain a framework for an approximation algorithm for Robust Min q-MSMC. The crucial point is to guarantee the polynomial running time of the algorithm despite the size of the uncertainty set. If the elements of the uncertainty set can be enumerated in polynomial time, this leads to an approximation algorithm whose approximation guarantee is bounded by $\sum_{i=1}^{k} 1/i$ with $k := q \cdot |\mathcal{U}|$. Otherwise, we utilize the concept of dominating uncertainty sets to regain an approximation algorithm that applies to any uncertainty set having a polynomial time optimization oracle. In this case, the ratio is deteriorated by a factor of $\mathcal{O}(\sqrt{|J|})$ where J denotes the ground set of the given instance.

In Chapter 5, we analyze new variants of Robust Min q-MSMC which limit the power of the decision maker and increase the impact of uncertainty. To that end, we include behavior patterns for the elements of the ground set since the demand of an element can also be interpreted as a present amount of clients. In the original problem, the clients have no influence on the problem's solution. We focus on three patterns: (a) Free clients, that independently choose the subset by which they want to be covered, (b) adapting clients, who are independent as well but need to choose their favorite subset from a restricted set, and (c) ordered clients, who reveal a preference order over their corresponding subsets and are covered by their favorite available subset. This leads to the optimization problems *Robust Min q-Free Clients*, *Robust Min q-Adapting Clients*, and *Robust Min q-Ordered Clients*. For each of these, we provide an extensive complexity analysis for both the non-robust and the robust variant. Further, we also concentrate on restrictions of the problems to the above mentioned classes of uncertainty sets. Our analyses reveal these problems to be substantially different compared to the initial (Robust) Min q-MSMC problem. For instance, Min q-Free Clients can be solved in linear time while its robust variant is NP-hard for any fixed value of $q \in \mathbb{N}_{>0}$. On the other hand, Min q-Adapting Clients leads to a special Min Set Cover problem which provides the basis for proving APX-hardness for any fixed q. Hence, this result also holds for the robust variant of this problem. For Robust Min q-Ordered Clients, we show APX-completeness in case $q \cdot |\mathcal{U}|$ is fixed and exceeds the value of 1. For the only remaining case, where $q = 1$ as well as $|\mathcal{U}| = 1$, we provide a linear time algorithm.

Finally, in Chapter 6, we conclude this thesis and provide various further research directions in the area spanned by the problems encountered.

Publications

As parts of this thesis are already published, we provide a full list of these publications here. In each case, more details on the relation of the content of the chapter and the publication can be found at the very beginning of the corresponding chapter.

Some parts of Chapter 3 are published in [KSS19] and are the result of mutual collaboration with Sven O. Krumke and Manuel Streicher. The basis of Chapter 5 is developed in [Büs+21] which is joint work together with Christina Büsing, Martin Comis, and Manuel Streicher. Hence, some results of Chapter 5 are published in a similar form in Martin Comis' thesis [Com21]. Due to the mentioned journal publications and further joint research in these fields, for both mentioned chapters, several similar considerations also appear in Manuel Streicher's thesis [Str21].

2. Preliminaries

In this chapter, we mainly settle our mathematical notation and definitions to avoid misconceptions. In large parts, we stay in line with common notational conventions and definitions. Further, we give brief overviews on particular mathematical fields as, for instance, robust optimization. For each section, we provide additional literature in which full details are available. We assume the reader is familiar with fundamental mathematical concepts and the theory of linear and integer optimization. For a thorough introduction to these fields, we refer to [Rud76; Beu94; Roc97; Heu08; Heu09] as well as [GLS93; Sch98; NW88].

2.1. Basic Notation

In this section, we list our basic notation concerning sets, numbers, vectors, matrices, and functions. For details on these definitions we additionally refer the reader to [Sch02].

Sets A *collection* is used as a synonym for a set. We denote by $\varnothing$ the *empty set* that contains no elements with $\max \varnothing := -\infty$ as well as $\min \varnothing := \infty$. For two sets A and B, we write $A \subseteq B$ if every element of A is also contained in B, i.e., the set A is a *subset* of B and B is a *superset* of A. If both $A \subseteq B$ and $B \subseteq A$, we write $A = B$. Hence, if $A \subseteq B$ but $A \neq B$, we also write $A \subsetneq B$. Further, with $A \cap B$ we note the *intersection* of these sets while $A \cup B$ denotes their *union*. If additionally $A \cap B = \varnothing$, we write $A \dot{\cup} B$ and say that A and B are *disjoint*. With $A \setminus B$ we denote the *relative complement* of B in A, i.e., the set of elements that are contained in A but not in B. Further, the set $A \times B$ denotes the *product set* of A and B which is the set of all ordered pairs (a, b) with $a \in A$ and $b \in B$. A *partition* of a set A is a collection of pairwise disjoint subsets of A whose union gives A.

Moreover, we write $|A| \in \mathbb{R} \cup \{\infty\}$ for the *cardinality* or *size* of a set A, i.e., the number of elements contained in A. If $|A| = k < \infty$ for some $k \in \mathbb{N}$, we say that A is *finite* and also call A a *k-set*. For a k-set, we implicitly assume that the elements are indexed from 1 to k and identify an element with its index if no ambiguities can occur. With 2^A we refer to the *power set* of A, i.e., the set of all subsets of A. Note that, for a k-set A with $k \in \mathbb{N}$, we have $|2^A| = 2^k$. A *multiset* is a set that may contain an element multiple times. With $\mathrm{m}(x, A) \in \mathbb{N}$ we denote the *multiplicity* of an element x in the set A with $\mathrm{m}(x, A) \coloneqq 0$ if $x \notin A$. Hence, for a multiset or set A, we have $|A| = \sum_{x \in A} \mathrm{m}(x, A)$.

Numbers We denote by $\mathbb{N}$ the set of *natural* numbers, i.e, $\mathbb{N} = \{0, 1, \ldots\}$. Similarly, we let $\mathbb{Z}$, $\mathbb{Q}$, and $\mathbb{R}$ be the sets of *integer*, *rational*, and *real* numbers, respectively. We set $\mathbb{B} \coloneqq \{0, 1\}$. If we only want to consider non-negative or positive elements of these sets, we add a subscript $_{\geq 0}$ or $_{>0}$, e.g., $\mathbb{Z}_{>0} \coloneqq \mathbb{N}_{>0} \coloneqq \{n \in \mathbb{N} \colon n > 0\}$. For other restrictions, we adapt the subscript appropriately.

For a number $x \in \mathbb{R}$, we denote by $\lfloor x \rfloor$ the largest integer y with $y \leq x$ and by $\lceil x \rceil$ we denote the smallest integer y with $y \geq x$. These procedures are called *rounding down* and *rounding up*, respectively. For $k \in \mathbb{N}_{>0}$, the *k^{th} harmonic number*, or simply *harmonic number*, is the value $\mathrm{H}(k) \coloneqq \sum_{i=1}^{k} {}^{1}\!/_{i}$. By [Kla79] we get that $\mathrm{H}(k) \leq 1 + \ln k$ for the natural logarithm ln.

Vectors and Matrices For $n \in \mathbb{N}_{>0}$, we often consider the vector space $\mathbb{R}^n$ equipped with the Euclidean norm $\|\cdot\|_2$. A vector (or point) $x \in \mathbb{R}^n$ is assumed to be a *column vector* and x^T denotes the corresponding *row vector*. For a finite set $A = \{a_1, \ldots, a_n\}$ with $n \in \mathbb{N}_{>0}$, we identify a function $x \colon A \to \mathbb{R}$ with the vector $x' \in \mathbb{R}^n$ defined by $x'_i \coloneqq x(a_i)$. Further, for $i \in \{1, \ldots, n\}$, we denote by $e_i \in \mathbb{R}^n$ the i^{th} unit vector and, with slight abuse of notation, we write 1 and 0 for the vectors in $\mathbb{R}^n$ of all ones and all zeros, respectively. In any case, it will be clear from the context whether we refer to a number or a vector when using this notation.

For two vectors $x, y \in \mathbb{R}^n$, we write $x \leq y$ if $x_i \leq y_i$ for every coordinate $i \in \{1, \ldots, n\}$ and we say that y *dominates* x. Analogously, we define $x < y$, $x > y$, and $x \geq y$. For vectors $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$ with $n + m \geq 3$, we write (x, y) to refer to the (column) vector in $\mathbb{R}^{n+m}$ whose first n entries are determined by x and the last m entries are determined by y. For a vector $x \in \mathbb{R}^n$ and a set $S \subseteq \{1, \ldots, n\}$, we use the common notation $x(S) \coloneqq \sum_{i \in S} x_i$. Furthermore, the *incidence vector* $\chi^S \in \mathbb{B}^n$ of S is given by $\chi_i^S := 1$ if and only if $i \in S$.

A vector $x \in \mathbb{R}^n$ is called a *linear combination* of the vectors $x^1, \dots, x^k \in \mathbb{R}^n$ for $k \in \mathbb{N}_{>0}$ if there is $\lambda \in \mathbb{R}^k$ such that $x = \sum_{i=1}^k \lambda_i x^i$. If additionally $\sum_{i=1}^k \lambda_i = 1$, the vector x is called an *affine combination*. If additionally $\lambda \geq 0$, the vector x is called a *convex combination*. Moreover, the vectors $x^1, \dots, x^k$ of $\mathbb{R}^n$ are *linearly independent* if the only solution $\lambda \in \mathbb{R}^k$ to the equations $\sum_{i=1}^k \lambda_i x^i = 0$ is given by $\lambda = 0$. Otherwise, they are *linearly dependent*. The vectors $x^1, \dots, x^k$ of $\mathbb{R}^n$ are *affinely independent* if the only solution $\lambda \in \mathbb{R}^k$ to the equations $\sum_{i=1}^k \lambda_i x^i = 0$ and $\sum_{i=1}^k \lambda_i = 0$ is given by $\lambda = 0$. Otherwise, they are *affinely dependent*.

With $\mathbb{R}^{m \times n}$ we denote the set of all real matrices with m rows and n columns. Analogously, we define the set of matrices with entries from $\mathbb{B}$, $\mathbb{N}$, $\mathbb{Z}$, or $\mathbb{Q}$. We write I_n for the *unit matrix* in $\mathbb{R}^{n \times n}$, i.e., the diagonal matrix with only ones on the diagonal. For a matrix $A \in \mathbb{R}^{m \times n}$, we refer to the entry of A in row $i \in \{1, \dots, m\}$ and column $j \in \{1, \dots, n\}$ as A_{ij}. Moreover, we write $A_{i\cdot}$ and $A_{\cdot j}$ to refer to the i^{th} row and j^{th} column of A, respectively. A quadratic matrix $A \in \mathbb{R}^{n \times n}$ is *symmetric* if $A = A^T$. Throughout this thesis, we identify a matrix $Y \in \mathbb{R}^{m \times n}$ with the vector $y \in \mathbb{R}^{mn}$ by $y_{m(i-1)+j} := Y_{ij}$ for $i \in \{1, \dots, m\}$ and $j \in \{1, \dots, n\}$, i.e., the rows of the matrix Y successively set up the vector y. For a quadratic matrix $A \in \mathbb{R}^{n \times n}$, we write A^{-1} for its *inverse* if it exists. A matrix $A \in \mathbb{R}^{n \times n}$ is *positive definite* if $x^T A x > 0$ for every $x \in \mathbb{R}^n$. Moreover, we know that A is invertible in this case. If, for a quadratic symmetric matrix $A \in \mathbb{R}^{n \times n}$, we have $|A_{ii}| > \sum_{j \neq i} |A_{ij}|$ and $A_{ii} > 0$ for every $i \in \{1, \dots n\}$, we get that A is positive definite, cf. [HJ12].

Sets in $\mathbb{R}^n$ For a set $X \subseteq \mathbb{R}^n$, a vector $y \in \mathbb{R}^n$, and a scalar $\lambda \in \mathbb{R}$, we set $\lambda \cdot X := \{\lambda \cdot x \colon x \in X\}$ and $X + \lambda \cdot y := \{x + \lambda \cdot y \colon x \in X\}$. For two sets $X, Y \subseteq \mathbb{R}^n$, the set Y *dominates* X if, for every $x \in X$, there is $y \in Y$ dominating x.

A subset $X \subseteq \mathbb{R}^n$ is *convex* if, for any two vectors $x, y \in X$ and any $\lambda \in [0, 1]$, we have $\lambda x + (1 - \lambda) y \in X$. For a general set $X \subseteq \mathbb{R}^n$, we write $\operatorname{conv}(X)$ to denote the *convex hull* of the set X, i.e., the set of all finite convex combinations of vectors of X. If X is a set of $n + 1$ affinely independent vectors, we call $\operatorname{conv}(X)$ a *simplex*. The *affine hull* of X, denoted by $\operatorname{aff}(X)$, is the set of all finite affine combinations of vectors of X. Then, the *dimension* of the set X, $\dim(X)$, is the dimension of $\operatorname{aff}(X)$.

Further, a set $X \subseteq \mathbb{R}^n$ is *bounded* if there is $r \in \mathbb{R}_{\geq 0}$ such that $\|x\|_2 \leq r$ for every $x \in X$. For a vector $x \in \mathbb{R}^n$ and a number $r > 0$, we write $\mathrm{B}_r(x) \coloneqq \{y \in \mathbb{R}^n \colon \|x - y\|_2 < r\}$ to denote the (open) *Euclidean ball* around x with radius r. For a subset $X \subseteq \mathbb{R}^n$, we denote by X^{o} the *interior* of X, i.e., the set of all vectors $x \in X$, such that there exists $r > 0$ with $\mathrm{B}_r(x) \subseteq X$. The *relative interior* of X is the set of all vectors $x \in X$ such that there is $r > 0$ with $\mathrm{B}_r(x) \cap \operatorname{aff}(X) \subseteq X$ and labeled as X^{ri}. Hence, $X^{\mathrm{o}} \subseteq X^{\mathrm{ri}}$. The set X is *open* if $X^{\mathrm{o}} = X$. Further, we say that $Y \subseteq \mathbb{R}^n$ is *closed* if $\mathbb{R}^n \setminus Y$ is open. If Y is additionally bounded, we say that Y is *compact*. In particular, we have that every finite set $Y \subseteq \mathbb{R}^n$ is compact. For a compact set $X \subseteq \mathbb{R}^n$ and a vector $c \in \mathbb{R}^n$, we know that $\max\left\{c^T x \colon x \in X\right\}$ as well as $\min\left\{c^T x \colon x \in X\right\}$ exist. If X is additionally convex and has a non-empty relative interior, we call X a *convex body*, cf. [BF06].

For a set $X \subseteq \mathbb{R}^n$ and a positive integer $p \leq n$, we write $X|_p$ to denote the *projection* of X onto its first p coordinates, i.e., $X|_p \coloneqq \{(x_1, \ldots, x_p) \colon x \in X\}$. A *polyhedron* $\mathrm{P}(A, b)$ in $\mathbb{R}^n$ is defined by a matrix $A \in \mathbb{R}^{m \times n}$ and a vector $b \in \mathbb{R}^m$ with $\mathrm{P}(A, b) \coloneqq \{x \in \mathbb{R}^n \colon Ax \leq b\}$. If $\mathrm{P}(A, b)$ is bounded, we call the set a *polytope*. Furthermore, a polyhedron $\mathrm{P}(A, b)$ is *rational* if $A \in \mathbb{Q}^{m \times n}$ as well as $b \in \mathbb{Q}^m$. For a positive definite matrix $A \in \mathbb{R}^{n \times n}$ and a vector $a \in \mathbb{R}^n$, the set $\mathrm{E}(A, a) \coloneqq \left\{x \in \mathbb{R}^n \colon (x - a)^T A^{-1} (x - a) \leq 1\right\}$ is called an *ellipsoid*. For an ellipsoid $\mathrm{E}(A, a) \subseteq \mathbb{R}^n$, let C be the unique positive definite matrix with $A = C^2$. Then, we have $\mathrm{E}(A, a) = \{Cx + a \colon \|x\|_2 \leq 1\}$. Furthermore, the subsequent lemma concerning the optimization of a linear function in an ellipsoid holds.

Lemma 2.1 ([GLS93])**.** *For an ellipsoid $E(A, a) \subseteq \mathbb{R}^n$ and a vector $c \in \mathbb{R}^n$ it holds true that*

$$\begin{aligned} \max\left\{c^T x \colon x \in E(A, a)\right\} &= c^T a + \sqrt{c^T A c}, \\ \min\left\{c^T x \colon x \in E(A, a)\right\} &= c^T a - \sqrt{c^T A c}. \end{aligned}$$ □

Functions For two sets A, B and a bijection $f \colon A \to B$, we denote by f^{-1} its inverse. For $A \subseteq \mathbb{R}$, a function $f \colon A \to \mathbb{R}$ is *monotonically increasing* if, for $x, y \in A$ with $x < y$, we have $f(x) \leq f(y)$. We write ln for the *natural logarithm* and log for the *logarithm to the base 2*. For two functions $f, g \colon \mathbb{N} \to \mathbb{N}$, we say that $f(n)$ is $\mathcal{O}(g(n))$ if there is $n_0 \in \mathbb{N}$ and a constant $c > 0$ such that, for all $n \geq n_0$, we have $f(n) \leq c \cdot g(n)$. If $f(n)$ is $\mathcal{O}(g(n))$, we also say $g(n)$ is $\Omega(f(n))$. If both $f(n)$ is $\mathcal{O}(g(n))$ and $g(n)$ is $\mathcal{O}(f(n))$, we say that $f(n)$ is $\Theta(g(n))$.

2.2. Complexity and Approximation

In this section, we list our basics concerning complexity and approximation. For details we refer to the books [GJ79; Sch02; Aus+02; WS09; AB17] from which the definitions are extracted.

Decision and Optimization Problems The theory of NP-completeness only applies to decision problems. Hence, we define: A *decision problem* $\mathcal{P}$ consists of a set $D_{\mathcal{P}}$ of *instances* and a subset $Y_{\mathcal{P}} \subseteq D_{\mathcal{P}}$ of *yes-instances*. An instance in $N_{\mathcal{P}} \coloneqq D_{\mathcal{P}} \setminus Y_{\mathcal{P}}$ is called a *no-instance*. The *complementary problem* $\mathcal{P}^c$ corresponding to $\mathcal{P}$ is given by $D_{\mathcal{P}^c} \coloneqq D_{\mathcal{P}}$ and $Y_{\mathcal{P}^c} \coloneqq N_{\mathcal{P}}$. As customary, we define a decision problem by giving a generic instance and stating a yes-no question.

On the other hand, we analyze optimization problems: An *optimization problem* $\mathcal{P}$ is given by

(a) a set $D_{\mathcal{P}}$ of *instances*,

(b) a function $f_{\mathcal{P}}$ that returns, to any instance $\mathcal{I} \in D_{\mathcal{P}}$, the set of *solutions*,

(c) an *objective function* $\mathrm{SOL}_{\mathcal{P}}$ which, given an instance $\mathcal{I}$ and a solution $x \in f_{\mathcal{P}}(\mathcal{I})$, returns the *solution value* $\mathrm{SOL}_{\mathcal{P}}(\mathcal{I}, x) \in \mathbb{Q}_{>0}$ of the solution x, and

(d) the information whether $\mathcal{P}$ is a maximization or a minimization problem.

A solution $x \in f_{\mathcal{P}}(\mathcal{I})$ is also called *feasible*. Similarly, an instance $\mathcal{I} \in D_{\mathcal{P}}$ is called *feasible* if $f_{\mathcal{P}}(\mathcal{I}) \neq \varnothing$. Again, according to the usual practice, we define an optimization problem $\mathcal{P}$ by stating a generic instance, its generic set of solutions, and the objective function. The information whether $\mathcal{P}$ is a minimization or maximization problem is given in the problem's name using the prefixes *Min* or *Max*. Let $\mathcal{P}$ be a minimization problem and $\mathcal{I} \in D_{\mathcal{P}}$. A solution $x^\star \in f_{\mathcal{P}}(\mathcal{I})$ is called *optimal* if $\mathrm{SOL}_{\mathcal{P}}(\mathcal{I}, x^\star) \leq \mathrm{SOL}_{\mathcal{P}}(\mathcal{I}, x)$ for every $x \in f_{\mathcal{P}}(\mathcal{I})$. The *optimal value* $\mathrm{OPT}_{\mathcal{P}}(\mathcal{I})$ of the instance $\mathcal{I}$ is given by $\mathrm{OPT}_{\mathcal{P}}(\mathcal{I}) \coloneqq \mathrm{SOL}_{\mathcal{P}}(\mathcal{I}, x^\star)$. The *decision problem corresponding to* $\mathcal{P}$ asks, for an additionally given integer $B \in \mathbb{N}$, whether there exists a solution

$x \in f_{\mathcal{P}}(\mathcal{I})$ with $\mathrm{SOL}_{\mathcal{P}}(\mathcal{I}, x) \leq B$. Analogously, we define these terms in case of a maximization problem.

Algorithms The *encoding length* $\langle \mathcal{I} \rangle$ of an instance $\mathcal{I}$ of a decision or optimization problem $\mathcal{P}$ is the number of bits needed to encode $\mathcal{I}$ in some natural encoding scheme, i.e., a scheme that does not include superfluous information and in which numbers are encoded in binary. More generally, we denote with $\langle X \rangle$ the encoding length of an object X. For instance, for a vector $x \in \mathbb{N}^n$, we have $\langle x \rangle = \sum_{i=1}^{n} \log x_i$.

Let $\mathcal{A}$ be an algorithm for a decision problem $\mathcal{P}$. This means, for any instance $\mathcal{I} \in D_{\mathcal{P}}$, the algorithm $\mathcal{A}$ returns the correct answer. An algorithm $\mathcal{A}$ for an optimization problem $\mathcal{P}$ returns, for any instance $\mathcal{I} \in D_{\mathcal{P}}$, either an optimal solution or the information that no feasible solution exists. Apart from the problem type of $\mathcal{P}$, we say that $\mathcal{A}$ *solves* $\mathcal{P}$. We measure the *running time* $T_{\mathcal{A}}$ of algorithm $\mathcal{A}$ with input $\mathcal{I} \in D_{\mathcal{P}}$ as a function of $\langle \mathcal{I} \rangle$ using the algorithmic model of the *unit-cost random access machine*. With this model, elementary arithmetic operations like addition, subtraction, multiplication, division, and comparison can be accomplished in one computational step. Other operations, like taking the square root, take time linear in the encoding length of the considered number and $|\log \varepsilon|$ where $\varepsilon > 0$ is the desired precision. This model is convenient as the encoding lengths of the numbers appearing in our algorithms can be bounded by a polynomial in $\langle \mathcal{I} \rangle$.

Further, to suppress constants and technical characteristics of the machines, we concentrate on *asymptotic* running times. The *worst-case running time* of algorithm $\mathcal{A}$ on input size $n \in \mathbb{N}$ is given by

$$T_{\mathcal{A}}(n) \coloneqq \max \left\{ T_{\mathcal{A}}(\langle \mathcal{I} \rangle) \colon \mathcal{I} \in D_{\mathcal{P}} \wedge \langle \mathcal{I} \rangle \leq n \right\}.$$

Then, we say that $\mathcal{A}$ has *time complexity (upper bound)* $\mathcal{O}(g(n))$ if $T_{\mathcal{A}}(n)$ is $\mathcal{O}(g(n))$ for a function $g \colon \mathbb{N} \to \mathbb{N}$. If $T_{\mathcal{A}}(n)$ is $\Omega(g(n))$, we say that $\mathcal{A}$ has *time complexity (lower bound)* $\Omega(g(n))$. If $T_{\mathcal{A}}(n)$ is $\Theta(g(n))$, algorithm $\mathcal{A}$ has time complexity exactly $\Theta(g(n))$. For instance, we say that $\mathcal{A}$ runs in time $\mathcal{O}(g(n))$ and $\mathcal{P}$ is solved in time $\mathcal{O}(g(n))$. Further, the algorithm $\mathcal{A}$ is a *polynomial time* algorithm if its time complexity is $\mathcal{O}(n^k)$ for some fixed $k \in \mathbb{N}$. In this case, we call $\mathcal{P}$ *polynomial time solvable* or *tractable*. Otherwise, algorithm $\mathcal{A}$ is called an *exponential time* algorithm. Moreover, we say that $\mathcal{A}$ is a *strongly polynomial time* algorithm if its time complexity is polynomially bounded in the dimension of the input, cf. [GLS93]. For an instance $\mathcal{I}$ of $\mathcal{P}$, we denote by $\max(\mathcal{I})$ the magnitude of the largest number occurring in $\mathcal{I}$. We call $\mathcal{A}$ a *pseudo-polynomial time* algorithm if, for every instance $\mathcal{I}$ of encoding length $n \in \mathbb{N}$, its time complexity is bounded by a polynomial in n and $\max(\mathcal{I})$.

Let $\mathcal{P}$ be an optimization problem. For some $r \geq 1$, an *r-approximation algorithm* for $\mathcal{P}$ is a polynomial time algorithm $\mathcal{A}$ that, given an instance $\mathcal{I} \in D_{\mathcal{P}}$, either concludes that $\mathcal{I}$ is infeasible or computes a solution $\mathcal{A}(\mathcal{I})$ for $\mathcal{I}$ with the property that $\mathrm{SOL}_{\mathcal{P}}(\mathcal{I}, \mathcal{A}(\mathcal{I})) \leq r \cdot \mathrm{OPT}_{\mathcal{P}}(\mathcal{I})$, if $\mathcal{P}$ is a minimization problem, or $r \cdot \mathrm{SOL}_{\mathcal{P}}(\mathcal{I}, \mathcal{A}(\mathcal{I})) \geq \mathrm{OPT}_{\mathcal{P}}(\mathcal{I})$ if $\mathcal{P}$ is a maximization problem. Disregarding the running time of $\mathcal{A}$, we call $\mathcal{A}(\mathcal{I})$ an *r-approximate solution* or an *r-approximation*. The value r is called *(performance) ratio* or *guarantee*. Analogously, we can define an *$r(n)$-approximation algorithm* for some function $r\colon \mathbb{N} \to [1, \infty)$ which, given an instance $\mathcal{I}$ with $\langle \mathcal{I} \rangle \leq n$ for $n \in \mathbb{N}$, outputs in polynomial time either a solution with performance ratio $r(n)$ or the information that $\mathcal{I}$ is infeasible. The definition of an *$r(n)$-approximate solution* follows easily.

Complexity Classes In general, a complexity class is a collection of either decision or optimization problems. For a function $g\colon \mathbb{N} \to \mathbb{N}$, let $\mathrm{DTIME}(g(n))$ be the set of all decision problems which can be solved by a (deterministic) algorithm with time complexity $\mathcal{O}(g(n))$. Then, we define $\mathrm{P} \coloneqq \bigcup_{k=0}^{\infty} \mathrm{DTIME}(n^k)$, i.e., P is the class of decision problems solvable in polynomial time in the encoding length of the input.

A *non-deterministic* algorithm is an algorithm that is also allowed to "guess" values from a finite set of possibilities, e.g., the set $\{0, 1\}$, during its execution. Depending on these guesses the continuation of the algorithm varies so that its outcome can be represented by a tree, in which each guess corresponds to a vertex, cf. Section 2.3. Given a decision problem $\mathcal{P}$ and $n \in \mathbb{N}$, we say that a non-deterministic algorithm $\mathcal{A}$ *solves $\mathcal{P}$ in time $T(n)$* if, for any instance $\mathcal{I} \in D_{\mathcal{P}}$ with $\langle \mathcal{I} \rangle \leq n$, we have $\mathcal{I} \in Y_{\mathcal{P}}$ if and only if there exists a sequence of guesses such that $\mathcal{A}$ returns YES in time at most $T(n)$. Observe that the length of the guess sequence is bounded by $T(n)$. For a function $g\colon \mathbb{N} \to \mathbb{N}$, let $\mathrm{NTIME}(g(n))$ be the set of all decision problems which can be solved by a non-deterministic algorithm in time $\mathcal{O}(g(n))$. Finally, we define NP to be the class of decision problems that can be solved by some non-deterministic algorithm in polynomial time in the encoding length of the input. Hence, NP $:= \bigcup_{k=0}^{\infty} \mathrm{NTIME}(n^k)$. Let $\mathcal{P}$ be a decision problem. Then, $\mathcal{P} \in$ NP if and only if there exists a polynomial p and, for every instance $\mathcal{I} \in D_{\mathcal{P}}$, we have: $\mathcal{I} \in Y_{\mathcal{P}}$ if and only if there exists a *certificate* $y(\mathcal{I})$ with $\langle y(\mathcal{I})\rangle \leq p(\langle \mathcal{I}\rangle)$ and, given $y(\mathcal{I})$, we can confirm $\mathcal{I} \in Y_{\mathcal{P}}$ in time polynomial in $\langle \mathcal{I}\rangle$. The class of decision problems that are complementary to some problem in NP is denoted by co-NP.

Complexity classes containing optimization problems are given by the two classes NPO and PO. We start by considering the class NPO. An optimization problem $\mathcal{P}$ is contained in NPO if

(a) $D_{\mathcal{P}}$ is recognizable in polynomial time,

(b) there is a polynomial p such that, for any instance $\mathcal{I}$ and any solution $x \in f_{\mathcal{P}}(\mathcal{I})$, $\langle x\rangle \leq p(\langle \mathcal{I}\rangle)$,

(c) for any instance $\mathcal{I}$ and any x with $\langle x\rangle \leq p(\langle \mathcal{I}\rangle)$, we can decide in polynomial time whether x is feasible, and

(d) the objective function $\mathrm{SOL}_{\mathcal{P}}$ can be computed in polynomial time.

For an optimization problem $\mathcal{P} \in$ NPO, we get by definition that its corresponding decision problem is contained in NP. Furthermore, the set of optimization problems in NPO which are polynomial time solvable is denoted by PO.

We now define two additional complexity classes of optimization problems which are contained in NPO. The class APX contains all problems $\mathcal{P}$ in NPO such that, for some fixed $r \geq 1$, there exists an r-approximation algorithm for $\mathcal{P}$. Further, a *polynomial time approximation scheme (PTAS)* for $\mathcal{P}$ is an algorithm $\mathcal{A}$ which, given an instance $\mathcal{I}$ and a fixed rational $r > 1$, returns in time polynomial in $\langle \mathcal{I}\rangle$ either an r-approximate solution or the information that $\mathcal{I}$ is infeasible. With slight abuse of notation, we denote with PTAS the class of problems in NPO that admit a polynomial time approximation scheme.

Reductions A decision problem $\mathcal{P}_1$ *Karp-reduces* to a decision problem $\mathcal{P}_2$ if there exists an algorithm $\mathcal{A}\colon D_{\mathcal{P}_1} \to D_{\mathcal{P}_2}$ with the property that $\mathcal{I} \in Y_{\mathcal{P}_1}$ if and only if $\mathcal{A}(\mathcal{I}) \in Y_{\mathcal{P}_2}$. If $\mathcal{A}$ runs in polynomial time with respect to the input size, the algorithm is also called a *polynomial time reduction* and we write $\mathcal{P}_1 \leq_{\mathrm{p}} \mathcal{P}_2$. Hence, if $\mathcal{P}_2 \in$ P and $\mathcal{P}_1 \leq_{\mathrm{p}} \mathcal{P}_2$, we get $\mathcal{P}_1 \in$ P. We say that P is *closed* with respect to $\leq_{\mathrm{p}}$ and $\leq_{\mathrm{p}}$ *preserves membership* in P. Furthermore, for decision problems $\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3$ with $\mathcal{P}_1 \leq_{\mathrm{p}} \mathcal{P}_2$ and $\mathcal{P}_2 \leq_{\mathrm{p}} \mathcal{P}_3$, we get that $\mathcal{P}_1 \leq_{\mathrm{p}} \mathcal{P}_3$, i.e., polynomial time reductions are *transitive*.

For a more general reduction, we need to define the notion of an oracle: Let $\mathcal{P}$ be a decision or an optimization problem. We say that an algorithm is an *oracle* for $\mathcal{P}$ if we suppose that the algorithm solves any instance of $\mathcal{P}$ in one computational step. Now, let $\mathcal{P}_1$ be a decision or an optimization problem. We say that $\mathcal{P}_1$ *Turing-reduces* to a problem $\mathcal{P}_2$ if there exists an algorithm $\mathcal{A}$ for $\mathcal{P}_1$ which has access to an oracle for $\mathcal{P}_2$. If $\mathcal{A}$ runs in polynomial time with respect to the input size, we write $\mathcal{P}_1 \leq_{\mathrm{T}} \mathcal{P}_2$. Again, for two decision problems $\mathcal{P}_1, \mathcal{P}_2$, we get that $\mathcal{P}_2 \in \mathrm{P}$ and $\mathcal{P}_1 \leq_{\mathrm{T}} \mathcal{P}_2$ implies $\mathcal{P}_1 \in \mathrm{P}$. As above, we also have that P is closed with respect to $\leq_{\mathrm{T}}$. For an optimization problem $\mathcal{P}$ and its corresponding decision problem $\mathcal{P}'$, we directly get that $\mathcal{P}' \leq_{\mathrm{T}} \mathcal{P}$. Note that Turing reductions are transitive as well.

For optimization problems, we need to define approximation preserving reductions. There are several concept available, cf. [Cre97] for a survey. To promote readability, we mostly omit the index $\mathcal{P}$ in the following.

A minimization problem $\mathcal{P}_1$ *AP-reduces* to a minimization problem $\mathcal{P}_2$ if there is a fixed value $\alpha > 0$ such that, for each instance $\mathcal{I}$ of $\mathcal{P}_1$ and any fixed rational $r > 1$, the following holds:

(a) We can compute in polynomial time an instance $\mathcal{I}'_r$ of $\mathcal{P}_2$.

(b) For any solution y_r to $\mathcal{I}'_r$, we can compute in polynomial time a solution $x(y_r)$ to $\mathcal{I}$ and $\mathrm{SOL}(\mathcal{I}'_r, y_r) \leq r \cdot \mathrm{OPT}(\mathcal{I}'_r)$ implies

$$\mathrm{SOL}(\mathcal{I}, x(y_r)) \leq (1 + \alpha \cdot (r-1)) \cdot \mathrm{OPT}(\mathcal{I}).$$

Analogously, we can define an AP-reduction in case $\mathcal{P}_1$ or $\mathcal{P}_2$ is a maximization problem. If $\mathcal{P}_1$ AP-reduces to $\mathcal{P}_2$, we write $\mathcal{P}_1 \leq_{\mathrm{AP}} \mathcal{P}_2$. Both classes APX and PTAS are closed with respect to $\leq_{\mathrm{AP}}$.

Finally, to simplify proving the existence of AP-reductions, we introduce L-reductions: An optimization problem $\mathcal{P}_1$ *L-reduces* to an optimization problem $\mathcal{P}_2$ ($\mathcal{P}_1 \leq_{\mathrm{L}} \mathcal{P}_2$) if there are positive constants $\alpha, \beta > 0$ such that, for each instance $\mathcal{I}$ of $\mathcal{P}_1$, the following holds:

(a) We can compute in polynomial time an instance $\mathcal{I}'$ of $\mathcal{P}_2$ with $\mathrm{OPT}(\mathcal{I}') \leq \alpha \cdot \mathrm{OPT}(\mathcal{I})$.

(b) For any solution y to $\mathcal{I}'$, we can compute in polynomial time a solution $x(y)$ to $\mathcal{I}$ such that

$$|\mathrm{OPT}(\mathcal{I}) - \mathrm{SOL}(\mathcal{I}, x(y))| \leq \beta \cdot |\mathrm{OPT}(\mathcal{I}') - \mathrm{SOL}(\mathcal{I}', y)| \, .$$

As hinted above L-reductions are useful for the following reason: If $\mathcal{P}_1 \leq_{\mathrm{L}} \mathcal{P}_2$ and $\mathcal{P}_1 \in$ APX, then $\mathcal{P}_1 \leq_{\mathrm{AP}} \mathcal{P}_2$. Throughout this thesis, we make use of this observation. Similar to AP-reducibility, we get that L-reductions preserve membership in PTAS and, for minimization problems, also in APX. Furthermore, both reductions are transitive. For instance, if $\mathcal{P}_1 \leq_{\mathrm{AP}} \mathcal{P}_2$ and $\mathcal{P}_2 \leq_{\mathrm{AP}} \mathcal{P}_3$, we have $\mathcal{P}_1 \leq_{\mathrm{AP}} \mathcal{P}_3$.

Completeness and Hardness A decision problem $\mathcal{P}$ is NP-*complete* if $\mathcal{P} \in$ NP and, for any decision problem $\mathcal{P}' \in$ NP, we have $\mathcal{P}' \leq_{\mathrm{p}} \mathcal{P}$. By transitivity it suffices to provide an NP-complete problem $\mathcal{P}'$ such that $\mathcal{P}' \leq_{\mathrm{p}} \mathcal{P}$ to show that a decision problem $\mathcal{P} \in$ NP is NP-complete. Analogously, we define co-NP-completeness. Note that a decision problem is co-NP-complete if and only if its complementary problem is NP-complete.

A decision or optimization problem $\mathcal{P}$ is NP-*hard* if, for every decision problem $\mathcal{P}' \in$ NP, we have $\mathcal{P}' \leq_{\mathrm{T}} \mathcal{P}$. Thus, if there exists some NP-complete problem $\mathcal{P}'$ with $\mathcal{P}' \leq_{\mathrm{T}} \mathcal{P}$, we get that $\mathcal{P}$ is NP-hard as polynomial time reductions are a special case of polynomial time Turing reductions and by transitivity. Further, with this definition we have that any NP-complete problem is also NP-hard. As P is closed with respect to $\leq_{\mathrm{T}}$, we see that an NP-hard problem cannot be solved in polynomial time unless P = NP. Finally, for an optimization problem $\mathcal{P}$ and its corresponding decision problem $\mathcal{P}'$, we get that NP-completeness of $\mathcal{P}'$ directly leads to NP-hardness of $\mathcal{P}$.

Once more, let $\mathcal{P}$ be a decision or optimization problem. Then, $\mathcal{P}$ is *strongly* NP-*hard* if there exists a polynomial p such that $\mathcal{P}$ restricted to instances $\mathcal{I}$ with $\max(\mathcal{I}) \leq p(\langle \mathcal{I} \rangle)$ is NP-hard. If additionally $\mathcal{P} \in$ NP, we say that $\mathcal{P}$ is *strongly* NP-*complete*. Hence, if there is a pseudo-polynomial time algorithm for some strongly NP-hard problem $\mathcal{P}$, we directly get that P = NP. Analogously, we define *strong* co-NP-*completeness*.

An optimization problem $\mathcal{P}$ is APX-*hard* if, for every optimization problem $\mathcal{P}' \in$ APX, we have $\mathcal{P}' \leq_{\mathrm{AP}} \mathcal{P}$. Further, an APX-hard problem $\mathcal{P}$ is APX-*complete* if $\mathcal{P} \in$ APX. If there is a PTAS for some APX-hard problem, we get APX = PTAS as $\leq_{\mathrm{AP}}$ preserves membership in PTAS.

This implies P = NP. Hence, no APX-hard problem can have a PTAS unless P = NP. By our observations, we have that an APX-complete problem $\mathcal{P}'$ with $\mathcal{P}' \leq_{\mathrm{L}} \mathcal{P}$ directly leads to APX-hardness of $\mathcal{P}$. This constitutes our standard technique in proving APX-hardness results.

2.3. Graph Theory

In this section, we briefly formulate our notation concerning graphs and graph-theoretical concepts. This notation is guided by [Sch02; KN12; Die17]. Yet, observe that we directly exclude the possibility of parallel edges in undirected as well as directed graphs.

Graphs An *(undirected) graph* G is a tuple (V, E) in which V is a non-empty set of *vertices* or *nodes* and E is the set of *edges* with $V \cap E = \varnothing$. The set E is a collection of 2-multisubsets of V. Hence, the graphs considered in this thesis do not contain parallels. Let $e = \{u, v\} \in E$. The vertices u and v are called the *end vertices* of the edge e. We also say that e *connects* its end vertices. According to notational convention, we also write $[u, v]$ for an edge $\{u, v\} \in E$. If $\{u, u\} \in E$, we write $[u, u]$ and call this edge a *loop*.

A graph G is of *order* n if $|V| = n$ for some $n \in \mathbb{N}_{>0}$. Two vertices $u, v \in V$ are *adjacent* or *neighbors* if there is an edge $[u, v] \in E$. Further, an edge $e \in E$ and a vertex $v \in V$ are *incident* or *cover* each other if $v \in e$. We denote by $N_G(v)$ the *(open) neighborhood* of $v \in V$ in G, i.e., the set of all vertices adjacent to v. Further, the *closed neighborhood* of v is given by $N_G[v] := N_G(v) \cup \{v\}$. For a subset $S \subseteq V$, $N_G(S)$ is the set of all nodes adjacent to some node in S. We say that a vertex v *dominates* another vertex u if $u \in N_G[v]$. In particular, the vertex v dominates itself. Concerning incidence, we denote by $\delta_G(v)$ the set of incident edges of a vertex $v \in V$ and $\deg_G(v) := \sum_{e \in \delta_G(v)} \mathrm{m}(v, e)$ is called the *degree* of the vertex v. Vertices with degree zero are called *isolated*. The *degree* of G is the maximum degree of its vertices and denoted by Δ_G. Furthermore, two edges $e_1, e_2 \in E$ are *incident* if there is a vertex $v \in V$ such that $e_1, e_2 \in \delta_G(v)$.

A graph without loops is called *simple*. Further, a graph G is *complete* if E is the set of all 2-element subsets of V. A graph G whose vertex set can be partitioned into two sets, i.e., $V = I \cup J$ with $I \cap J = \varnothing$, such that each edge connects a vertex of I with a vertex of J is called *bipartite*. A bipartite graph G is *complete* if $N_G(i) = J$ for every $i \in I$. Note that, by our definitions, bipartite graphs are always simple. When illustrating bipartite graphs in this thesis, we use boxes for the vertices of one vertex set and circles for the other vertices. A *path* P in a graph $G = (V, E)$ is a sequence of vertices $P = (v_0, v_1, \dots, v_n)$ with $v_i \in V$ for every $i \in \{0, \dots, n\}$ such that $[v_i, v_{i+1}] \in E$ for every $i \in \{0, \dots, n-1\}$. Observe that these edges are uniquely defined by their end vertices. We say that P has *length* n and *connects* v_0 and v_n. If $v_0 = v_n$ and $n \geq 2$ we call P a *cycle*. A graph that merely consist of a cycle of length 3 is called a *triangle*. Further, a graph is *connected* if any two of its vertices are connected by a path. A *tree* is a connected graph that does not contain a cycle. For two graphs $G = (V, E)$ and $G' = (V', E')$, we define their *union* $G \cup G'$ as the graph with vertex set $V \cup V'$ and edge set $E \cup E'$. For a graph $G = (V, E)$, a *subgraph* of G is given by a graph $G' = (V', E')$ such that $V' \subseteq V$ and $E' \subseteq E$. Further, for a subset $E' \subseteq E$, the *subgraph induced by* E' is the subgraph $G[E'] = (V', E')$ with $V' \coloneqq \bigcup_{e \in E'} e$.

An *edge cover* of a graph $G = (V, E)$ is a subset of the edges $E' \subseteq E$ such that every vertex $v \in V$ is incident to some edge in E'. Moreover, a *matching* of G is a loopless subset E' of the edges such that any two edges of E' do not share a common end vertex. A matching is *perfect* if it is a matching as well as an edge cover. These concepts naturally lead to optimization problems.

Problem 2.2 (Min Edge Cover)**.**
Instance: A simple graph $G = (V, E)$.
Solution: An edge cover E'.
Measure: The cardinality of E'.

Problem 2.3 (Max Matching)**.**
Instance: A simple graph $G = (V, E)$.
Solution: A matching E'.
Measure: The cardinality of E'.

Both problems can be solved in polynomial time [Sch02]. In particular, an instance G of Min Edge Cover is typically solved using an algorithm for Max Matching on G and augmenting the found matching greedily in time $\mathcal{O}(m)$, where $m \in \mathbb{N}$ is the number of edges of G.

Directed graphs A *directed graph* G is a tuple (V, R) in which V is a non-empty set of *vertices* or *nodes* and $R \subseteq V^2$ is a set of *(directed) edges* or *arcs*. Hence, in correspondence to undirected graphs, we do not allow an arc to appear multiple times. For an arc $a = (u, v) \in R$, we call u its *start vertex* and v its *end vertex*. Moreover, we say that *a leaves* u and *enters* v. Hence, the vertex u is a *predecessor* of v and the vertex v is a *successor* of u. The notions of incidence and adjacency are defined analogously to undirected graphs. For a vertex $v \in V$, we denote by $N_G^+(v)$ the set of successors of v. Analogously, by $N_G^-(v)$ we denote the set of predecessors of v. Further, $\delta_G^+(v)$ denotes the set of arcs leaving v while $\delta_G^-(v)$ is the set of arcs entering v. For a subset $S \subseteq V$, we write $\delta_G^+(S)$ for the set of arcs with start vertex in S and end vertex in $V \setminus S$. Then, the *degree* $\deg_G(v)$ of a vertex $v \in V$ is defined as the sum of its *in-degree* $|\delta_G^-(v)|$ and its *out-degree* $|\delta_G^+(v)|$. A *network* $G = (V, R, c)$ is a directed graph G together with a mapping $c\colon R \to \mathbb{Q}_{\geq 0}$.

For directed as well as for undirected graphs, we omit the subscript G if the considered graph is clear from the context. Further, in both cases, we assume the vertex and edge sets to be finite.

Flows and Cuts Given a network $G = (V, R, c)$ and two distinct vertices $s, t \in V$, a *(feasible) s-t-flow* in G is a mapping $f\colon R \to \mathbb{Q}_{\geq 0}$ such that, for every vertex $v \in V \setminus \{s, t\}$, it holds true that $f(\delta^+(v)) = f(\delta^-(v))$ and, for every arc $r \in R$, we have $f(r) \leq c(r)$. The first set of constraints is called the *flow conservation constraints* while the second set of constraints is referred to as the *capacity constraints*. Further, $\mathrm{val}(f) \coloneqq f(\delta^+(s)) - f(\delta^-(s))$ is called the *flow value* of f. To simplify notation we set $f(u, v) \coloneqq f((u, v))$ for an arc $(u, v) \in R$. Moreover, an *s-t-cut* (S, T) in G is a partition of the vertex set V into two sets S and T with $s \in S$ and $t \in T$. The *capacity* of the s-t-cut is given by $c(S, T) \coloneqq c(\delta^+(S))$. If the vertices s and t are given from the context, we speak of flows and cuts for simplicity. These two concepts on networks lead to the following optimization problems.

Problem 2.4 (Max Flow)**.**
Instance: A network $G = (V, R, c)$ and two distinct vertices $s, t \in V$.
Solution: An s-t-flow f.
Measure: The flow value $\mathrm{val}(f)$.

Problem 2.5 (Min Cut).
Instance: A network $G = (V, R, c)$ and two distinct vertices $s, t \in V$.
Solution: An s-t-cut (S, T).
Measure: The cut capacity $c(S, T)$.

It is a well-known result that, for an instance of these problems, the optimal values coincide and can be computed in polynomial time. For a proof of this result and a survey on possible running times, we refer to [AMO93; Sch02].

Theorem 2.6 (Max-Flow-Min-Cut Theorem). *Given a network $G = (V, R, c)$ and two distinct vertices $s, t \in V$, it holds true that*

$$\max\{\mathrm{val}(f)\colon f \textit{ is an } s\text{-}t\textit{-flow}\} = \min\{c(S,T)\colon (S,T) \textit{ is an } s\text{-}t\textit{-cut}\}. \square$$

2.4. Set Cover Problems

In this section, we define several generalizations and specifications of the well-known Min Set Cover problem that are apparent throughout this thesis. In an informal fashion, some of them have already been introduced in Chapter 1. Nevertheless, Min Set Cover is formally defined as follows, cf. [GJ79; Aus+02]:

Problem 2.7 (Min Set Cover).
Instance: A collection $\mathcal{C}$ of subsets of a finite set S.
Solution: A *(set) cover* for S, i.e., a subset $\mathcal{C}' \subseteq \mathcal{C}$ such that every element in S is contained in at least one set of $\mathcal{C}'$.
Measure: The cardinality of $\mathcal{C}'$.

The decision version of this problem, called Set Cover, is well-known to be NP-complete [Kar72]. This result directly follows from NP-hardness of this special variant of Set Cover [GJ79].

Problem 2.8 (Exact Cover by 3-Sets).
Instance: A set S with $|S| = 3r$ for $r \in \mathbb{N}_{>0}$ and a collection $\mathcal{C}$ of 3-element subsets of S.
Question: Does $\mathcal{C}$ contain an *exact cover* for S, i.e., a subcollection $\mathcal{C}' \subseteq \mathcal{C}$ such that every element of S appears in exactly one set of $\mathcal{C}'$?

Yet, a simple polynomial time greedy algorithm, which always chooses the set covering the greatest number of new elements, achieves an approximation ratio of $\mathrm{H}(k)$, where $k \in \mathbb{N}$ is the maximum subset size of the given instance [Joh74; Lov75]. Up to additive constants in the ratio, this guarantee is essentially optimal: We have $\mathrm{H}(k) \leq 1 + \ln k$, see [Kla79], and Min Set Cover cannot be approximated within $(1 - \varepsilon) \ln n$ for any fixed $\varepsilon > 0$ unless $\mathrm{P} = \mathrm{NP}$ where $n := |S|$, cf. [DS14; Mos15]. A further inapproximability result is given in [KR08] and more details on approximation algorithms for covering problems are presented later in this thesis, cf. Section 4.2.1. Furthermore, a study on recent exact algorithms for Min Set Cover can be found in [CTF00].

Another specification of the Min Set Cover problem is the restriction to instances whose sizes of the subsets are bounded by some constant $K \in \mathbb{N}$. This problem is denoted by Min Set Cover(K) and is APX-complete for fixed $K \geq 3$, compare [Aus+02; AK00]. For $K = 1$, the problem is trivially solvable in linear time. Further, for $K = 2$, an instance of Min Set Cover(2) corresponds to an instance of Min Edge Cover by identifying the finite set S with the vertices of a graph, in which each edge corresponds to one subset $C \in \mathcal{C}$. Hence, Min Set Cover(2) can also be solved in polynomial time according to Section 2.3.

In the following, further relevant generalization of Min Set Cover are given in Table 2.1. To ease notation we introduce the *element-set incidence matrix* $M^{SC} \in \mathbb{B}^{m \times n}$ for a given finite set $S = \{s_1, \ldots, s_m\}$ and a collection $\mathcal{C} = \{C_1, \ldots, C_n\}$ of subsets of S: For $s_i \in S$ and $C_j \in \mathcal{C}$, we set $M^{SC}_{ij} := 1$ if and only if $s_i \in C_j$. For each problem in Table 2.1, the variable x_j refers to the frequency of choosing the j^{th} column of the corresponding constraint matrix. Constraints which serve as upper bounds for the variables x, i.e., $x \leq d$ for $d \in \mathbb{N}^n$, are called *multiplicity constraints (MC)*. Note that these constraints are redundant for Min Set Cover but constitute a different problem when added to its generalizations. Here, the multiplicity constraints are only appended to the most general problem Min CIP to capture all necessary definitions.

Considering Table 2.1, we see that in the weighted Min Set Cover problem, compared to the classical problem, the objective is to minimize the total cost of the chosen subsets. Moreover, in an instance of Min Set Multicover, each element $s_i \in S$ is assigned a demand value $b_i \in \mathbb{N}_{>0}$ expressing the number of times the element s_i needs to be covered, cf. [HH92]. A further generalization is given by Min Multiset Multicover in which the

Name	Instance	Definition	
Weighted Min Set Cover	$S, \mathcal{C}$ $c \in \mathbb{N}^n_{>0}$	min s.t.	$c^T x$ $M^{SC} x \geq 1$ $x \in \mathbb{B}^n$
Min Set Multicover	$S, \mathcal{C}$ $b \in \mathbb{N}^m_{>0}$ $c \in \mathbb{N}^n_{>0}$	min s.t.	$c^T x$ $M^{SC} x \geq b$ $x \in \mathbb{N}^n$
Min Multiset Multicover	$A \in \mathbb{N}^{m \times n}$ $b \in \mathbb{N}^m_{>0}$ $c \in \mathbb{N}^n_{>0}$	min s.t.	$c^T x$ $Ax \geq b$ $x \in \mathbb{N}^n$
Min Covering IP (Min CIP_∞)	$A \in \mathbb{Q}^{m \times n}_{\geq 0}$ $b \in \mathbb{Q}^m_{>0}$ $c \in \mathbb{N}^n_{>0}$	min s.t.	$c^T x$ $Ax \geq b$ $x \in \mathbb{N}^n$
Min Covering IP with MC (Min CIP)	$A \in \mathbb{Q}^{m \times n}_{\geq 0}$ $b \in \mathbb{Q}^m_{>0}$ $c, d \in \mathbb{N}^n_{>0}$	min s.t.	$c^T x$ $Ax \geq b$ $x \leq d$ $x \in \mathbb{N}^n$

Table 2.1.: Min Set Cover generalizations.

subsets in the collection $\mathcal{C}$ of a given instance are allowed to be multisets, see [Hua+09]. This fact is represented by the matrix $A \in \mathbb{N}^{m \times n}$ as an entry A_{ij} can be interpreted as the multiplicity of the element s_i in the set C_j. Analogously to Min Set Cover(K), we define Min Set Multicover(K) as well as Min Multiset Multicover(K). Dropping the integrality constraints on A and b results in a general covering integer program (Min CIP_∞), cf. [KY05]. Adding multiplicity constraints gives Min CIP. Furthermore, besides Min Set Cover all problems are weighted by definition although this is not directly implied by the problem's name. Concerning complexity, the above mentioned hardness results for Min Set Cover carry over. However, if each set in an instance of Min Multiset Multicover contains at most two elements, we can solve the instance in polynomial time by reducing the problem to a generalized Min Edge Cover problem, cf. Section 3.1 and [Sch02]. Exact exponential time algorithms for Min Set Multicover and Min Multiset Multicover are given, for instance, in [Hua+09] based on dynamic programming and in [HH92] based on heuristics together with branch and bound.

2.5. Constraint Generation

Constraint generation algorithms are widely used when it comes to solving (mixed integer) linear programs containing a huge number of constraints. Famous examples here are the Traveling Salesman problem, cf. [GJ79; PR91], or the application of Benders decomposition [Ben62; NW88]. We briefly sketch the basic idea of constraint generation. Different applications of this idea lead to varying implementations of constraint generation algorithms, e.g., see [PR91; Mit02] for the concept of *branch and cut* or [NW88]. Furthermore, note that by duality constraint generation for linear programs is closely related to column generation and the Dantzig-Wolfe reformulation [DDS05; Lüb10].

For the general concept, suppose we aim to solve the following linear program

$$\min \left\{ c^T x \colon Ax \leq b, x \geq 0 \right\} \tag{2.1}$$

with $A \in \mathbb{Q}^{m \times n}$, $b \in \mathbb{Q}^m$, $c \in \mathbb{Q}^n$, and a non-empty polytope $P := P(A, b) \cap \mathbb{R}^n_{\geq 0}$. Denote with $z \in \mathbb{R}$ its optimal value. If this program contains a very large number of constraints, i.e., m is large, a possible strategy is to start with the consideration of only a subset of the constraints. Let the constraints be indexed from 1 to m and let $M' \subseteq M := \{1, \dots, m\}$. Then, a relaxed problem is given by

$$\min \left\{ c^T x \colon A_{M'} x \leq b_{M'}, x \geq 0 \right\} \tag{2.2}$$

in which $A_{M'}$ and $b_{M'}$ are the restrictions of A and b to the rows of M', respectively. Let $z' \in \mathbb{R} \cup \{-\infty\}$ be its optimal value. The idea is that (2.2) is potentially easier to solve as $|M'| \leq m$. Trivially, any solution to (2.1) is also feasible for (2.2) so that $z' \leq z$. On the other hand, if (2.2) has a finite optimal solution x' which is also feasible for (2.1), we have found an optimal solution to (2.1). In case x' is not feasible for (2.1), there must be a constraint $i' \in M \setminus M'$ which is violated by x', i.e., $A_{i'}.x' > b_{i'}$. The same holds for a vector $x \in \mathrm{P}(A_{M'}, b_{M'}) \cap \mathbb{R}^n_{\geq 0}$ if (2.2) is unbounded. Hence, in the *separation step*, given a finite optimal solution x' (or a solution x' with $c^T x'$ small in case (2.2) is unbounded) of the relaxed problem, we need to find a constraint $i' \in M$ which is violated by x' or assess correctly that none exits. If no such constraint exists, x' is optimal for (2.1) and the procedure stops. On the other hand, if we find a violating constraint i', we know that $i' \in M \setminus M'$ and we update M' by adding i'. This means, we add a *feasibility cut* to the relaxed problem. Now, the relaxed problem is resolved and the procedure iterates. Initially, we can start with $M' = \varnothing$ or some arbitrary desired subset $M' \subseteq M$. As $m < \infty$, the algorithm terminates in a finite number of steps and in the worst case we reach a point with $|M'| = m$.

Hence, for the effectiveness of constraint generation methods, the computational complexity of the performed separation steps is an important issue. Depending on the applied constraint generation algorithm, these steps differ slightly. In this thesis, we stay with the presented classical approach. With this approach, a famous theorem in [GLS88] states that, if a bounded polyhedron $P \subseteq \mathbb{R}^n$ is well-described[1], the following problems are polynomial time equivalent:

(a) Given a vector $c \in \mathbb{Q}^n$, find a vector $x \in P$ maximizing $c^T x$ on P or assert that P is empty.

(b) Given a vector $y \in \mathbb{Q}^n$, decide whether $y \in P$. If not, find a vector $c \in \mathbb{Q}^n$ with $c^T y > \max\left\{c^T x \colon x \in P\right\}$.

This means, (a) can be solved in time polynomial in $\langle P \rangle + \langle c \rangle$ if and only if (b) can be solved in time polynomial in $\langle P \rangle + \langle y \rangle$. We also refer to (b) as the *separation problem* for P. Hence, if the separation step in the above constraint generation algorithm can be solved in polynomial time and $\{x \in \mathbb{R}^n \colon Ax \leq b, x \geq 0\}$ is well-described, we can solve our original problem (2.1) in polynomial time.

[1]A polyhedron $P \subseteq \mathbb{R}^n$ is *well-described* if there is $\rho \in \mathbb{N}$ such that P can be described by rational linear inequalities, each of which having encoding length at most ρ. The encoding length of P is $n + \rho$, cf. [GLS88].

2.6. Robust Optimization

In recent years, the concept of robustness and the theory of robust optimization have received an increasing interest in the operations research community since people are interested in hedging against various eventualities, e.g., data uncertainty. The basic idea of robust optimization can be described as follows: When solving an instance of an optimization problem, not all data of the instance may be known exactly at the time of computation. To account for these uncertainties, instead of given values for the parameters of the instance, we are given a *set of scenarios* $\mathcal{U}$, the *uncertainty set*, in which each scenario defines values for $p \in \mathbb{N}_{>0}$ predefined parameters. In this thesis, a scenario is given by a vector $\xi \in \mathbb{R}^p$ in which each entry corresponds to some particular parameter of the instance. We assume that any of the scenarios contained in $\mathcal{U}$ may actually occur. Yet, we do not know the *true* scenario beforehand. Hence, the aim is to find a solution which takes all scenarios of the given uncertainty set into account.

The idea of robust optimization was pioneered in [Soy73] and, subsequently, it became a major research area within the optimization community with Ben-Tal and Nemirovsky, cf. [BN98; BN99; BN00], and El Ghaoui et al., cf. [EL97; EOL98]. A thorough general introduction and overview is available in [BGN09; KY97]. Furthermore, a recent survey is given in [GMT14]. A detailed compilation of various robustness concepts, such as *strict*, *regret*, or *adjustable* robustness, is given in [GS16]. Here, we focus on strict and adjustable robustness.

More formally, in robust optimization we deal with *uncertain optimization problems* in which an instance $\mathcal{I}$ corresponds to a collection of instances of an optimization problem $\mathcal{P}$ and each scenario $\xi \in \mathcal{U}$ defines a single instance of the optimization problem, i.e., $\mathcal{I} = \{\mathcal{I}_\xi \in D_\mathcal{P} \colon \xi \in \mathcal{U}\}$. For this introduction, we let $\mathcal{P}$ be a minimization problem. An analogous consideration is certainly possible for maximization problems as well.

Strict Robustness We start with considering the concept of strict robustness. Due to our above interpretation of the uncertainty set $\mathcal{U}$, we seek a solution x in the intersection $\bigcap_{\xi \in \mathcal{U}} f_\mathcal{P}(\mathcal{I}_\xi)$, i.e., a solution x which is feasible for every given instance $\mathcal{I}_\xi$ encoded by the uncertainty set $\mathcal{U}$. Such a solution is called *robust feasible* or a *robust solution* for $\mathcal{I}$ and $\sup \{\mathrm{SOL}(\mathcal{I}_\xi, x) \colon \xi \in \mathcal{U}\}$ is called the corresponding *robust solution value*. Note that this implicitly claims the instances $\mathcal{I}_\xi$ for $\xi \in \mathcal{U}$ to be of a common structure. To hedge against the prevalent uncertainty, it is now natural to ask for a robust solution with the best possible worst-case outcome. Hence, we aim to find a robust solution such that its robust solution value is minimum. To achieve this, we aggregate the instances $\mathcal{I}_\xi$ for $\xi \in \mathcal{U}$ and consider the *robust counterpart* of the given uncertain problem, i.e., the optimization problem

$$\min \{t\colon \operatorname{SOL}(\mathcal{I}_\xi, x) \leq t, x \in f_{\mathcal{P}}(\mathcal{I}_\xi) \text{ for } \xi \in \mathcal{U}\}. \tag{2.3}$$

We call an optimization problem of type (2.3) *robust optimization problem.* The corresponding decision problem is called *robust decision problem.* For an optimal solution $(x^\star, t^\star)$ to (2.3), we get that $x^\star$ is robust feasible for $\mathcal{I}$ and $t^\star$ is the minimum robust solution value. We also say that $x^\star$ is *robust optimal* for $\mathcal{I}$.

In current literature, various methods for defining classes of uncertainty sets have been proposed. We list [BS04; BS03; KY97; Kas08] for a general overview. In this thesis, an uncertainty set $\mathcal{U}$ is a non-empty and compact subset of $\mathbb{R}^p_{\geq 0}$ for some $p \in \mathbb{N}_{>0}$. Moreover, we assume that $\langle \mathcal{U} \rangle$ is $\Omega(p)$ as well as $\max_{\xi \in \mathcal{U}} \xi_j > 0$ for every coordinate $j \in \{1, \ldots, p\}$. Further, we suppose that the encoding length of every scenario $\xi \in \mathcal{U}$ is polynomial in $\langle \mathcal{U} \rangle$. We mostly deal with uncertainty sets contained in $\mathbb{N}^p$. Hence, these uncertainty sets are always finite.

In the following, we specify some classes of uncertainty sets that we encounter in this thesis. Note that, unusually, all presented classes are composed of subsets of $\mathbb{N}^p$. Let $J \coloneqq \{1, \ldots, p\}$ denote the corresponding index set.

Discrete Uncertainty One arising concept is that of *discrete uncertainty* in which, for some fixed $k \in \mathbb{N}_{>0}$, the uncertainty set is given as an explicit list of k scenarios, cf. [KY97; KZ16]. Hence, we have $\mathcal{U}_{\mathrm{D}} \coloneqq \left\{\xi^1, \ldots, \xi^k\right\} \subseteq \mathbb{N}^p$ and the encoding length is given by $\langle \mathcal{U}_{\mathrm{D}} \rangle = \sum_{l=1}^k \left\langle \xi^l \right\rangle$. A robust optimization or decision problem $\mathcal{P}$ restricted to instances with discrete uncertainty is referred to as $\mathcal{P}$ *with discrete uncertainty.* For an application of discrete uncertainty, we mention [Dha+05].

Interval Uncertainty Soyster introduced *interval uncertainty* in the 1970s in the early stages of the theory of robust optimization, cf. [Soy73]. When considering interval uncertainty, the uncertainty set is described by the integral points of a hypercube, i.e., $\mathcal{U}_\mathrm{I} := \{\xi \in \mathbb{N}^p : a \leq \xi \leq b\}$ for some $a, b \in \mathbb{N}^p$ with $a \leq b$. Here, we have $\langle \mathcal{U}_\mathrm{I} \rangle = \langle a \rangle + \langle b \rangle$. A robust optimization or decision problem $\mathcal{P}$ restricted to instances with interval uncertainty is referred to as $\mathcal{P}$ *with interval uncertainty.* Despite its conservatism, interval uncertainty is frequently used and it gives first insights into a robust problem by its simple structure [KY97]. An application of interval uncertainty to Min Set Cover is given in [PA13].

Budgeted Uncertainty We investigate *budgeted uncertainty* as a generalization of interval uncertainty, i.e., $\mathcal{U}_\mathrm{B} := \{\xi \in \mathbb{N}^p : a \leq \xi \leq b, \xi(J) \leq \Gamma\}$ with $a, b \in \mathbb{N}^p$, $a \leq b$, and $\Gamma \in \mathbb{N}$. We have $\langle \mathcal{U}_\mathrm{B} \rangle = \langle a \rangle + \langle b \rangle + \langle \Gamma \rangle$. Note that this definition of budgeted uncertainty set differs from other settings using the same expression, e.g. [NO13; Cha+18; BPP19; HG18]. Here, we bound the total sum of the uncertain values. Moreover, this type of uncertainty is especially meaningful if there is a common interpretation of the uncertain parameters among the entries $\xi_1, \ldots, \xi_p$ of a scenario $\xi \in \mathcal{U}_\mathrm{B}$. A robust optimization or decision problem $\mathcal{P}$ restricted to instances with budgeted uncertainty is referred to as $\mathcal{P}$ *with budgeted uncertainty.* Budgeted uncertainty sets are widely applied in practice, e.g., in scheduling surgery blocks [Den+10], in emergency logistics planning [Ben+11], in inventory control [AP05], as well as in energy hub management [PVV12].

Multi-budgeted Uncertainty When considering *multi-budgeted uncertainty*, we extend the notion of budgeted uncertainty. To that end, let $\mathcal{S}$ be a finite collection of subsets of the given index set J. Then, for each $S \in \mathcal{S}$, we are given two non-negative integers a_S, b_S with $a_S \leq b_S$. The corresponding multi-budgeted uncertainty set is defined as $\mathcal{U}_\mathrm{M} := \{\xi \in \mathbb{N}^p : a_S \leq \xi(S) \leq b_S \text{ for } S \in \mathcal{S}\}$ and for the encoding length we get $\langle \mathcal{U}_\mathrm{M} \rangle = \langle \mathcal{S} \rangle + \sum_{S \in \mathcal{S}} (\langle a_S \rangle + \langle b_S \rangle)$. A robust optimization or decision problem $\mathcal{P}$ restricted to instances with multi-budgeted uncertainty is referred to as $\mathcal{P}$ *with multi-budgeted uncertainty.*

Ellipsoidal Uncertainty *Ellipsoidal uncertainty* sets and its generalizations belong to the most extensively studied uncertainty sets because of their wide range of advantages in modeling and computing. In this thesis, an ellipsoidal uncertainty set is given by an ellipsoid $\mathrm{E}(A, a) \subseteq \mathbb{R}^p_{\geq 0}$

for a positive definite matrix $A \in \mathbb{Q}^{p \times p}$ and a vector $a \in \mathbb{N}^p$. We have $\mathcal{U}_{\mathrm{E}} \coloneqq \{\xi \in \mathbb{N}^p \colon (\xi - a)^T A^{-1} (\xi - a) \leq 1\}$, i.e., $\mathcal{U}_{\mathrm{E}}$ contains all integral vectors of the *underlying* ellipsoid $\mathrm{E}(A, a)$. We get $\langle \mathcal{U}_{\mathrm{E}} \rangle = \langle A \rangle + \langle a \rangle$. Observe that this definition of ellipsoidal uncertainty is more restrictive than the one given in [BN99]. A robust optimization or decision problem $\mathcal{P}$ restricted to instances with ellipsoidal uncertainty is referred to as *$\mathcal{P}$ with ellipsoidal uncertainty*. Concerning applications, ellipsoidal uncertainty is often motivated from the field of stochastics, especially from the normal distribution. Here, deterministic ellipsoidal uncertainty can be used to model stochastic uncertainty [BN98; BN99]. In [BN99], this approach is applied to a portfolio problem.

Γ-Uncertainty *Γ-uncertainty* was first introduced in [BS03; BS04] as a possibility to overcome the drawbacks of Soyster's conservative model of uncertainty, cf. [Soy73]. In this thesis, a Γ-uncertainty set is given by $\mathcal{U}_\Gamma \coloneqq \{\xi \in \mathbb{N}^p \colon a \leq \xi \leq a + \hat{a}, |\{k \colon \xi_k \neq a_k\}| \leq \Gamma\}$ with *nominal vector* $a \in \mathbb{N}^p$, *deviation* $\hat{a} \in \mathbb{N}^p$, and $\Gamma \in \{0, \dots, p\}$. Hence, $\langle \mathcal{U}_\Gamma \rangle = \langle a \rangle + \langle \hat{a} \rangle + \langle \Gamma \rangle$. Observe that, unlike [BS03; BS04], we only allow a positive deviation from the nominal vector. We will see that this restriction is without loss of generality as the uncertainty set will model demand scenarios. Note that Γ-uncertainty is closely related to budgeted uncertainty: Atamtürk shows that Γ-uncertainty is a special case of a generalization of budgeted uncertainty in which, for every scenario, the weighted total sum of its entries is bounded [Ata06]. A robust optimization or decision problem $\mathcal{P}$ restricted to instances with Γ-uncertainty is referred to as *$\mathcal{P}$ with Γ-uncertainty*. In [BPP19], the concept of Γ-uncertainty is applied to a scheduling problem while, in [Gab+14], it is used for a location-transportation problem.

Polyhedral Uncertainty More generally, a *polyhedral uncertainty* set is defined as $\mathcal{U}_{\mathrm{P}} \coloneqq \mathrm{P}(A, b)|_p \cap \mathbb{N}^p$ for a polytope $\mathrm{P}(A, b) \subseteq \mathbb{R}^{p+l}_{\geq 0}$ for some $l \in \mathbb{N}$. The polytope $\mathrm{P}(A, b)$ is then called the *underlying polytope*. We have $\langle \mathcal{U}_{\mathrm{P}} \rangle = \langle A \rangle + \langle b \rangle + \langle p \rangle$. Observe that polyhedral uncertainty sets include interval, budgeted, multi-budgeted, as well as Γ-uncertainty sets. If the polytope $\mathrm{P}(A, b)$ is integral, we call $\mathcal{U}$ *integral polyhedral*. A robust optimization or decision problem $\mathcal{P}$ restricted to instances with (integral) polyhedral uncertainty is referred to as *$\mathcal{P}$ with (integral) polyhedral uncertainty*. For applications of polyhedral uncertainty, we refer to the already mentioned examples above.

In particular, with slight abuse of notation, we note that interval, budgeted, and Γ-uncertainty sets are integral polyhedral as

$$\begin{aligned}\mathcal{U}_{\mathrm{I}} &= \{x \in \mathbb{R}^p : a \leq x \leq b\} \cap \mathbb{N}^p,\\ \mathcal{U}_{\mathrm{B}} &= \{x \in \mathbb{R}^p : a \leq x \leq b, x(J) \leq \Gamma\} \cap \mathbb{N}^p,\\ \mathcal{U}_{\Gamma} &= \left\{x \in \mathbb{R}^p : \begin{array}{c} a_j \leq x_j \leq a_j + \hat{a}_j \cdot w_j \text{ for } j \in J, \\ w(J) \leq \Gamma, 0 \leq w \leq 1, x \geq 0 \end{array}\right\} \cap \mathbb{N}^p,\end{aligned} \tag{2.4}$$

and the underlying polytopes are integral [BS04; Ata06].

A robust optimization or decision problem $\mathcal{P}$ is referred to as $\mathcal{P}$ *with polynomial time optimization uncertainty* if, for every instance $\mathcal{I} \in D_{\mathcal{P}}$ with uncertainty set $\mathcal{U} \subseteq \mathbb{R}^p_{\geq 0}$ and every vector $c \in \mathbb{Q}^p$, the optimization problem $\max_{\xi \in \mathcal{U}} c^T x$ can be solved in polynomial time in the encoding length of $\mathcal{I}$ and c. We also say that $\mathcal{U}$ has a *polynomial time optimization oracle*. By (2.4), this applies to discrete, interval, budgeted, and Γ-uncertainty.

Further, a robust optimization or decision problem $\mathcal{P}$ is referred to as $\mathcal{P}$ *with polynomial time enumeration uncertainty* if, for every instance $\mathcal{I} \in D_{\mathcal{P}}$ with uncertainty set $\mathcal{U} \subseteq \mathbb{R}^p_{\geq 0}$, $|\mathcal{U}|$ is polynomial in $\langle \mathcal{I} \rangle$, there exists an algorithm $\mathcal{A}$ and a point $\xi^0 \in \mathcal{U}$ such that the elements of $\mathcal{U}$ can be recursively generated from ξ^0, i.e., $\mathcal{U} = \left\{\xi^0, \mathcal{A}(\xi^0), \mathcal{A}(\mathcal{A}(\xi^0)), \dots\right\}$, and $\mathcal{A}$ runs in time polynomial in $\langle \mathcal{I} \rangle$, cf. [Hem+91]. We also say that $\mathcal{U}$ is *polynomial time enumerable*. It can readily be seen that this applies to discrete uncertainty.

2.6.1. Adjustable Robustness

The concept of *adjustable robustness* was introduced in [Ben+04] to overcome some modeling drawbacks of the general strict robustness approach. A recent survey concerning this approach is given in [YGH19]. According to the strict approach, it is assumed that all decisions have to be made before the actual scenario reveals itself, i.e., all variables correspond to so-called *here and now* decisions. But in many applications we encounter variables which do not represent this kind of decisions. For instance, these could be auxiliary variables for linearizing a model or variables corresponding to actual *wait and see* decisions, which can be made when the true scenario is already known. To account for this differentiation, when applying adjustable robustness, we split up the set of variables of a given optimization problem into two sets: a set of *non-adjustable* variables corresponding to here and now decisions and a set of *adjustable* variables corresponding to auxiliary variables or wait and see decisions. Hence, the adjustable variables may depend on the data and can be adapted after the uncertainty is known. More generally, this idea can be extended to model variables that depend on only a part of the data, cf. [BGN09]. For our purposes, we stay with the more specific approach. Hence, for an uncertain problem $\{\mathcal{I}_\xi \in D_\mathcal{P} \colon \xi \in \mathcal{U}\}$ with minimization problem $\mathcal{P}$, we have a set of non-adjustable variables x and a set of adjustable variables $\{y(\xi) \colon \xi \in \mathcal{U}\}$ and the *adjustable robust counterpart* is given by

$$\min\{t \colon \operatorname{SOL}(\mathcal{I}_\xi, (x, y(\xi))) \leq t, (x, y(\xi)) \in f_\mathcal{P}(\mathcal{I}_\xi) \text{ for } \xi \in \mathcal{U}\}\,. \tag{2.5}$$

We also call problems of type (2.5) *adjustable robust optimization problems* and we refer to the corresponding decision problems as *adjustable robust decision problems.* For a solution $\{x, \{y(\xi) \colon \xi \in \mathcal{U}\}\}$ to (2.5), we also write (x, y) to stay in line with vector-type notation. On the contrary, the usual robust counterpart of $\{\mathcal{I}_\xi \in D_\mathcal{P} \colon \xi \in \mathcal{U}\}$ is

$$\min\{t \colon \operatorname{SOL}(\mathcal{I}_\xi, (x, y)) \leq t, (x, y) \in f_\mathcal{P}(\mathcal{I}_\xi) \text{ for } \xi \in \mathcal{U}\}\,, \tag{2.6}$$

which is more stringent than the adjustable robust counterpart. Observe that, by definition of (2.5) and (2.6), we can distinguish between a solution (x, y) to (2.5) and a solution (x, y) to (2.6) so that no ambiguities can occur. In case $\mathcal{P}$ is a maximization problem, we define the adjustable robust counterpart analogously.

In [Ben+04], the authors particularly consider adjustable robust linear programs and analyze polynomial time solvable cases. But in general, the optimization problem given in (2.5) is NP-hard. Therefore, the extension of the notion of approximation algorithms for adjustable robust problems is of interest. To incorporate the adjustable variables appropriately we define:

Definition 2.9 ((Adjustable) Approximation Algorithm)**.** Let an adjustable robust minimization problem $\mathcal{P}'$ as in (2.5) be given. For some $r \geq 1$, an *(adjustable) r-approximation algorithm* for $\mathcal{P}'$ is an algorithm $\mathcal{A}$ that, given an instance $\mathcal{I} = \{\mathcal{I}_\xi \in D_\mathcal{P} \colon \xi \in \mathcal{U}\}$ of $\mathcal{P}'$, computes a solution (x, y) to $\mathcal{I}$ with the property that $\operatorname{SOL}(\mathcal{I}, (x, y)) \leq r \cdot \operatorname{OPT}(\mathcal{I})$. The algorithm $\mathcal{A}$ computes x in time polynomial in $\langle \mathcal{I} \rangle$ and, for every $\xi \in \mathcal{U}$, $\mathcal{A}$ computes $y(\xi)$ in time polynomial in $\langle \mathcal{I} \rangle$. Analogously, we define an adjustable r-approximation algorithm in case $\mathcal{P}'$ is a maximization problem.

In the remainder of this section, we focus on a specific adjustable robust problem, namely the Min Adjustable Robust Covering problem (Min ARC) which is introduced, for instance, in [BG10]. Given

(a) constraint matrices $A \in \mathbb{Q}^{m \times n_1}$ and $B \in \mathbb{Q}^{m \times n_2}$,

(b) objective function vectors $c \in \mathbb{Q}^{n_1}_{\geq 0}$ and $d \in \mathbb{Q}^{n_2}_{\geq 0}$,

(c) two non-negative integers $p_1 \leq n_1$ and $p_2 \leq n_2$,

(d) and an uncertainty set $\mathcal{U} \subseteq \mathbb{R}^m_{\geq 0}$,

solving an instance of Min ARC corresponds to solving the mixed integer program $\mathrm{P}_{\mathrm{arc}}(\mathcal{U})$:

$$
\begin{array}{llrl}
\mathrm{P}_{\mathrm{arc}}(\mathcal{U}) & \min\limits_{x,\, y} & c^T x + \max\limits_{\xi \in \mathcal{U}} d^T y(\xi) & \\
 & \text{s.t.} & Ax + By(\xi) \geq \xi & \text{for } \xi \in \mathcal{U} \\
 & & x, y(\xi) \geq 0 & \text{for } \xi \in \mathcal{U} \\
 & & x \in \mathbb{R}^{n_1 - p_1} \times \mathbb{N}^{p_1} & \\
 & & y(\xi) \in \mathbb{R}^{n_2 - p_2} \times \mathbb{N}^{p_2} & \text{for } \xi \in \mathcal{U}.
\end{array}
$$

We assume that $\mathrm{P}_{\mathrm{arc}}(\mathcal{U})$ is feasible and let $z_{\mathrm{arc}}(\mathcal{U}) \in \mathbb{R}_{\geq 0}$ denote its optimal value. This problem and straightforward variations are extensively studied in literature. Exemplarily, we refer to [BG10; BC10; BGS11; BG11]. For our current purposes, we close this section by showing that, for $p_2 = 0$, we can assume the uncertainty set $\mathcal{U}$ to be convex.

Lemma 2.10. *For an instance of Min ARC with $p_2 = 0$, it holds true that $z_{arc}(\mathcal{U}) = z_{arc}(\mathrm{conv}(\mathcal{U}))$. Further, we can assume that, in any solution (x, y) to $P_{arc}(\mathrm{conv}(\mathcal{U}))$, we have that, for $\xi \in \mathrm{conv}(\mathcal{U})$, $y(\xi)$ is a convex combination of the vectors $\{y(\xi') \colon \xi' \in \mathcal{U}\}$.*

Proof. It holds that $z_{\mathrm{arc}}(\mathcal{U}) \leq z_{\mathrm{arc}}(\mathrm{conv}(\mathcal{U}))$ as $\mathcal{U} \subseteq \mathrm{conv}(\mathcal{U})$. Let (x, y) be feasible for $\mathrm{P}_{\mathrm{arc}}(\mathcal{U})$. Any scenario $\xi' \in \mathrm{conv}(\mathcal{U})$ can be written as a finite convex combination of $k \in \mathbb{N}$ scenarios in $\mathcal{U}$, i.e.,

$$\xi' = \sum_{l=1}^{k} \alpha_l \cdot \xi^l$$

with $\xi^l \in \mathcal{U}$ for every $l \in \{1, \dots, k\}$, $\alpha \in \mathbb{R}^k_{\geq 0}$ as well as $\sum_{l=1}^k \alpha_l = 1$. We define a solution (x', y') to $\mathrm{P}_{\mathrm{arc}}(\mathrm{conv}(\mathcal{U}))$ by setting $x' := x$ and, for $\xi' \in \mathrm{conv}(\mathcal{U})$,

$$y'(\xi') := \sum_{l=1}^{k} \alpha_l \cdot y(\xi^l).$$

Then, we have $x' \geq 0$ and, for all $\xi' \in \mathrm{conv}(\mathcal{U})$, $y'(\xi') \geq 0$ as well as

$$\begin{aligned} Ax' + By'(\xi') &= Ax + \sum_{l=1}^{k} \alpha_l \cdot By(\xi^l) = \sum_{l=1}^{k} \alpha_l \cdot \big(Ax + By(\xi^l)\big) \\ &\geq \sum_{l=1}^{k} \alpha_l \cdot \xi^l = \xi'. \end{aligned}$$

Thus, the constructed solution is feasible. For the solution value, it holds true that

$$\begin{aligned} &c^T x' + \max_{\xi' \in \mathrm{conv}(\mathcal{U})} d^T y'(\xi') \\ &\quad = c^T x + \max \left\{ \sum_{\xi \in \mathcal{U}'} \alpha_\xi \cdot d^T y(\xi) \colon \mathcal{U}' \subseteq \mathcal{U} \text{ finite}, \alpha \in \mathbb{R}^{|\mathcal{U}'|}_{\geq 0}, \alpha(\mathcal{U}') = 1 \right\} \\ &\quad = c^T x + \max_{\xi \in \mathcal{U}} d^T y(\xi). \end{aligned}$$

In particular, if (x, y) is optimal for $\mathrm{P}_{\mathrm{arc}}(\mathcal{U})$, we get that

$$z_{\mathrm{arc}}(\mathrm{conv}(\mathcal{U})) \leq c^T x + \max_{\xi \in \mathcal{U}} d^T y(\xi) = z_{\mathrm{arc}}(\mathcal{U}). \qquad \square$$

3. Robust Min q-Multiset Multicover

In this chapter, we define and investigate our main problem of interest, namely the Robust Min q-Multiset Multicover problem for a fixed integer $q \in \mathbb{N}_{>0}$. This problem forms a special robust Multiset Multicover problem as the multisets are not explicitly given and their cardinalities are bounded by q. We present two integer programming formulations of the problem that further allow for interpretations as a robust flow or a robust cut problem.

We begin with the non-robust variant of the problem in Section 3.1. We provide two equivalent formulations of the problem that we make use of in subsequent chapters and sections. Furthermore, we give polynomial time algorithms for the cases $q = 1$ and $q = 2$ and show APX-completeness for the remaining values of q. In Section 3.2, the problem is extended in a robust manner by introducing demand scenarios. Here, the equivalence result of the previous section neatly carries over and leads to two robust integer programs. After the problem's complexity analysis depending on the value of q, we describe the means for solving it using both formulations in Section 3.3. We propose constraint generation as a possible solution method and therefore analyze the emerging separation problem. It turns out that the structure of the considered uncertainty set plays a major role during the solution process. Therefore, in Section 3.4, we investigate restrictions of the problem to specific classes of uncertainty sets. We conclude this chapter with a brief consideration of special instances in Section 3.5.

Some parts of this chapter are also published in [KSS19] and [Str21] where, in [KSS19], the main focus is on budgeted uncertainty. Further, all findings presented in this chapter are joint work with Sven O. Krumke and Manuel Streicher. With this collaboration, also Manuel Streicher's thesis contains a chapter that covers some of the results mentioned here, cf. [Str21].

3.1. Problem Definition and Classification

Let us directly start with formally defining the non-robust version of our Multiset Multicover problem for a fixed positive integer $q \in \mathbb{N}_{>0}$. Note that the value q is not part of the input of an instance of our problem. We will see that it determines the maximum possible cardinality of the implicitly given sets.

Problem 3.1 (q-Multiset Multicover (q-MSMC)).
Instance: A finite set J, a demand value $d_j \in \mathbb{N}_{>0}$ for each $j \in J$, a collection of subsets $\mathcal{J} \subseteq 2^J$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|\mathcal{J}|}$ with $x(\mathcal{J}) \leq B$ such that there exists $y \in \mathbb{N}^{|\mathcal{J}|\times|J|}$ satisfying

$$\sum_{\substack{A\in\mathcal{J}:\\ j\in A}} y_{Aj} \geq d_j \text{ for } j \in J \text{ and } \sum_{j\in A} y_{Aj} \leq q \cdot x_A \text{ for } A \in \mathcal{J}?$$

To obtain a first intuition of the problem consider the following reformulation of the stated question: Can we choose no more than B subsets of $\mathcal{J}$, with multiple choices being allowed ($x_A \in \mathbb{N}$), such that the demand of each element $j \in J$ is covered, when each subset may only cover up to q of its elements, where again multiple choices are allowed ($y_{Aj} \in \mathbb{N}$). For a subset $A \in \mathcal{J}$ and an element $j \in J$, the integer y_{Aj} in the problem definition models the amount of the demand of element j covered by the subset A. Note that we can assume $y_{Aj} = 0$ if $j \notin A$. The problem is closely related to Multiset Multicover as the following remark reveals.

Remark 3.2 (Relation to Multiset Multicover). Let an instance of q-MSMC be given. If, instead of regarding the subset $A \in \mathcal{J}$, we regard all multi-subsets of cardinality q of A, we get an instance of Multiset Multicover(q), cf. Section 2.4: For $A \in \mathcal{J}$ and $l_A \in \mathbb{N}_{>0}$, let $A_1, \dots, A_{l_A}$ denote the multi-subsets of size q of A. Our constructed instance of Multiset Multicover(q) is given by the ground set J and the subsets $\mathcal{C} := \{A_1, \dots, A_{l_A} : A \in \mathcal{J}\}$. The demand of element $j \in J$ remains $d_j \in \mathbb{N}_{>0}$. Let $\bar{x} \in \mathbb{N}^{|\mathcal{C}|}$ be a solution to the constructed instance of Multiset Multicover(q), i.e., $\bar{x}_{A_k} \in \mathbb{N}$ represents how many times subset A_k is chosen. We define a solution (x, y) to the given instance of q-MSMC by $x_A := \sum_{k=1}^{l_A} \bar{x}_{A_k}$ and $y_{Aj} := \sum_{k=1}^{l_A} \mathrm{m}(j, A_k) \cdot \bar{x}_{A_k}$ for $A \in \mathcal{J}$ and $j \in J$. As by construction $\sum_{j\in A} \mathrm{m}(j, A_k) \leq q$ for every

$k \in \{1, \ldots, l_A\}$, we obtain, for $A \in \mathcal{J}$,

$$\begin{aligned}\sum_{j \in A} y_{Aj} &= \sum_{j \in A} \sum_{k=1}^{l_A} \mathrm{m}(j, A_k) \cdot \bar{x}_{A_k} = \sum_{k=1}^{l_A} \sum_{j \in A} \mathrm{m}(j, A_k) \cdot \bar{x}_{A_k} \\ &\leq \sum_{k=1}^{l_A} q \cdot \bar{x}_{A_k} = q \cdot x_A\end{aligned}$$

and the second set of constraints in Problem 3.1 is already fulfilled. Rewriting the remaining constraints we get, for $j \in J$,

$$\sum_{\substack{A \in \mathcal{J}: \\ j \in A}} y_{Aj} = \sum_{\substack{A \in \mathcal{J}: \\ j \in A}} \sum_{k=1}^{l_A} \mathrm{m}(j, A_k) \cdot \bar{x}_{A_k} \geq d_j$$

as $\bar{x}$ is feasible. Further, $x(\mathcal{J}) = \sum_{A \in \mathcal{J}} \sum_{k=1}^{l_A} \bar{x}_{A_k} \leq B$, so that (x, y) is a solution to the given instance of q-MSMC. Likewise, given a solution (x, y) to the instance of q-MSMC, we can compute a solution $\bar{x}$ to the Multiset Multicover(q) instance in polynomial time. Observe that $l_A = \binom{|A|+q-1}{q}$ is $\mathcal{O}(|A|^q)$ so that the input size is raised only by a polynomial factor as q is not part of the input. Thereby, q-MSMC is, in some sense, a representation of certain Multiset Multicover(q) instances having smaller input sizes as the multisets are not explicitly given.

In the sequel, we mostly work with the following alternative definition of q-MSMC where the given collection of subsets $\mathcal{J}$ is represented using a bipartite graph, cf. Section 2.3.

Problem 3.3 (q-Multiset Multicover (q-MSMC))**.**
Instance: Finite sets I, J with $I \cap J = \varnothing$, a demand value $d_j \in \mathbb{N}_{>0}$ for each $j \in J$, a bipartite graph $G = (I \cup J, E)$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that there exists $y \in \mathbb{N}^{|I| \times |J|}$ satisfying

$$\sum_{i \in N(j)} y_{ij} \geq d_j \text{ for } j \in J \text{ and } \sum_{j \in N(i)} y_{ij} \leq q \cdot x_i \text{ for } i \in I?$$

Identifying each element $i \in I$ with its neighborhood $N_G(i) \subseteq J$ yields the equivalence of the two problem definitions. For an instance of q-MSMC,

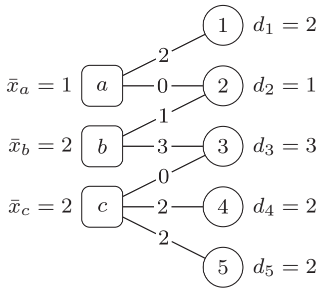


Figure 3.1.: Bipartite graph G defined in Example 3.4. A possible allocation of locations to regions is given as edge labels.

we call the set I *locations* and the set J *regions*. Further, the value d_j describes the number of *clients* or the total *demand* of region $j \in J$. The variable x_i denotes the number of *suppliers* in location $i \in I$ while the allocation variable y_{ij} describes the number of clients of region j covered by the suppliers in location i. Note that the clients in region j can only be covered by locations in $N_G(j)$ and, conversely, every supplier in location i can only cover clients in $N_G(i)$. Therefore, without loss of generality, the variable y_{ij} can be set to zero if $[i, j] \notin E$. Finally, the value q can be interpreted as the *capacity* of a supplier, i.e., the total number of clients a single supplier can serve. The following example illustrates the problem.

Example 3.4. Let $q := 2$ and consider the bipartite graph G shown in Figure 3.1 with locations $I := \{a, b, c\}$, regions $J := \{1, 2, 3, 4, 5\}$, demand vector $d := (2, 1, 3, 2, 2)^T$, and $B := 6$. Throughout this thesis locations are drawn as boxes and regions are illustrated as circles. If we put one supplier in location a and two suppliers in each of the locations b and c, we obtain a solution: For $\bar{x}_1 := 1$ and $\bar{x}_b := \bar{x}_c := 2$, a possible choice of values for $\bar{y} \in \mathbb{N}^{3\times 5}$ is shown as edge labels, e.g., the supplier in location a covers two clients in region 1 and no client in region 2. Note that this interpretation of the suppliers is not always unique, e.g., in location c there are two possible interpretations for the two suppliers: Either each supplier covers a single region or they both split their capacity across the regions 4 and 5. As the total number of suppliers is $5 \leq 6$, we have a yes-instance. When considering the problem as a Multiset Multicover(2) problem, we first identify each location with its neighborhood in G and obtain $N(a) = \{1, 2\}$,

$N(b) = \{2,3\}$, and $N(c) = \{3,4,5\}$. Observe that these three sets would make up the set $\mathcal{J}$ in the original definition, i.e., Problem 3.1. Now, we consider all multisubsets of size 2 of these sets and obtain the sets

$$\begin{aligned}
&N(a) \rightsquigarrow \{\mathbf{1},\mathbf{1}\},\ \{1,2\},\ \{2,2\},\\
&N(b) \rightsquigarrow \{2,2\},\ \{\mathbf{2},\mathbf{3}\},\ \{\mathbf{3},\mathbf{3}\},\\
&N(c) \rightsquigarrow \{3,3\},\ \{3,4\},\ \{3,5\},\ \{4,5\},\ \{\mathbf{4},\mathbf{4}\},\ \{\mathbf{5},\mathbf{5}\}.
\end{aligned}$$

A set $\{j_1, j_2\}$ corresponding to location i captures the information that a supplier in location i can cover one client in region j_1 and one client in region j_2. Selecting the multisets printed in bold corresponds to our choice of y. Note that this is not the only possible selection of multisets: For location c, it is also possible to select the set $\{4,5\}$ twice. This relates to the two possible interpretations of the suppliers in location c. All in all, this example demonstrates another difference to classical Multiset Multicover(q) problems as the presented solution $(\bar{x}, \bar{y})$ corresponds to more than one choice of multisets. ◁

In the optimization version of q-MSMC, that we call Min q-Multiset Multicover (Min q-MSMC), we aim for a minimum number of suppliers. Subsequently, we show that the following mixed integer program P(d) models Min q-MSMC for some demand vector $d \in \mathbb{N}^{|J|}_{>0}$:

$$\begin{aligned}
\mathrm{P}(d) \quad \min_{x,\,y} \quad & \sum_{i \in I} x_i && && \text{(3.1a)}\\
\text{s.t.} \quad & \sum_{i \in N(j)} y_{ij} \geq d_j && \text{for } j \in J && \text{(3.1b)}\\
& \sum_{j \in N(i)} y_{ij} \leq q \cdot x_i && \text{for } i \in I && \text{(3.1c)}\\
& y_{ij} \geq 0 && \text{for } i \in I, j \in J && \text{(3.1d)}\\
& x_i \in \mathbb{N} && \text{for } i \in I. && \text{(3.1e)}
\end{aligned}$$

We also refer to P(d) as the *allocation formulation.* Note that the variables y are not forced to be integral. Theorem 3.8 argues why this is no restriction. We begin with stating some trivial bounds for a given instance of Min q-MSMC.

Observation 3.5. First of all, an instance of Min q-MSMC is feasible if and only if every region $j \in J$ is adjacent to at least one location $i \in I$. This condition can be checked in time $\mathcal{O}(|J|)$. Therefore, we restrict our analysis to feasible instances from now on. Observe that we have $|J| \leq |E|$ then. Further, as we aim to minimize, it suffices to only consider solutions $x \in \mathbb{N}^{|I|}$ where, for every $i \in I$, $x_i \leq d(N(i))$. Hence, for every $i \in I$, the variable x_i can be bounded from above without cutting off the optimal solution. Moreover, any solution (x, y) to $\mathrm{P}(d)$ fulfills

$$\sum_{i \in I} q \cdot x_i \geq \sum_{i \in I} \sum_{j \in N(i)} y_{ij} = \sum_{j \in J} \sum_{i \in N(j)} y_{ij} \geq \sum_{j \in J} d_j = d(J).$$

Thus, $\mathrm{P}(d)$ has a finite optimal solution $(x^\star, y^\star)$ and we have

$$\sum_{i \in I} x_i^\star \geq \left\lceil \frac{d(J)}{q} \right\rceil \quad \text{and} \quad x_i^\star \leq \left\lceil \frac{d(N(i))}{q} \right\rceil$$

for all $i \in I$, cf. [Mey74; Sch98].

Example 3.6. By Observation 3.5 we directly obtain that the solution $\bar{x}$ given in Example 3.4 is optimal. ◁

Furthermore, we observe a link to the capacitated Min Facility Location problem [Sri95]. In an instance of this problem, we have given a set I of locations and a set J of regions. Each location $i \in I$ has a set-up cost $f_i \in \mathbb{N}$ and a capacity $q_i \in \mathbb{N}_{>0}$ and each region $j \in J$ has a demand $d_j \in \mathbb{N}_{>0}$. Furthermore, for $i \in I$ and $j \in J$, $c_{ij} \in \mathbb{N}$ is the cost of transporting one demand unit from location i to region j. The goal is to find an optimal solution to

$$\begin{aligned}
\min_{x, y} \quad & \sum_{i \in I} f_i \cdot x_i + \sum_{i \in I} \sum_{j \in J} c_{ij} \cdot y_{ij} \\
\text{s.t.} \quad & \sum_{i \in I} y_{ij} = 1 && \text{for } j \in J \\
& \sum_{j \in J} d_j \cdot y_{ij} \leq q_i \cdot x_i && \text{for } i \in I \\
& y_{ij} \geq 0 && \text{for } i \in I, j \in J \\
& x_i \in \mathbb{B} && \text{for } i \in I,
\end{aligned}$$

i.e., we want to decide on the locations to be set up and on the proration of locations to regions such that the demand of each region is satisfied, the capacity of each location is respected, and the total cost is minimized. Taking a closer look, there are three main differences to mention: In an instance of Min q-MSMC,

(a) the capacity of a location is a multiple of q and is not bounded from above,

(b) both cost values and the allocation variables y do not appear in the objective function, and

(c) the bipartite graph G does not need to be complete.

Especially the last two items are the reasons why we decided to define the problem as a covering problem. Another reason for this decision is given by the upcoming Theorem 3.8 where we prove that $\mathrm{P_s}(d)$ is an alternative formulation to $\mathrm{P}(d)$ which does not use the variables y.

$$\mathrm{P_s}(d) \quad \min_{x} \quad \sum_{i \in I} x_i \tag{3.2a}$$

$$\text{s.t.} \quad \sum_{i \in N(S)} q \cdot x_i \geq \sum_{j \in S} d_j \quad \text{for } S \subseteq J \tag{3.2b}$$

$$x_i \in \mathbb{N} \quad \text{for } i \in I. \tag{3.2c}$$

In correspondence to the allocation formulation, we sometimes refer to $\mathrm{P_s}(d)$ as the *subset formulation*. Before proving the mentioned theorem, we define a directed network that will be of use not only in Theorem 3.8.

Definition 3.7 (Network $H_G(q \cdot x, d)$)**.** Given an instance $\mathcal{I}$ of Min q-MSMC with bipartite graph $G = (I \cup J, E)$ and a vector $x \in \mathbb{N}^{|I|}$, we define the directed network $H_G(q \cdot x, d) = (V, R, c)$ with $V := I \cup J \cup \{s, t\}$ and $R := R_G \cup R_s \cup R_t$, where $s, t \notin I \cup J$, R_G contains all edges of E directed from I to J, $R_s := \{(s, i) : i \in I\}$, and $R_t := \{(j, t) : j \in J\}$, cf. Figure 3.2. We set the capacity of each arc $r \in R$ to

$$c(r) := \begin{cases} \infty, & r \in R_G, \\ q \cdot x_i, & r \in R_s, \\ d_j, & r \in R_t. \end{cases}$$

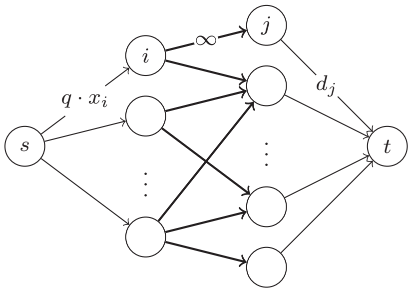


Figure 3.2.: Network $H_G(q \cdot x, d)$ of Definition 3.7 where the capacity of an arc is given as an arc label. All thick arcs have infinite capacity.

Theorem 3.8. *For an instance $\mathcal{I}$ of Min q-MSMC, it holds that $x \in \mathbb{N}^{|I|}$ is a solution to $P_s(d)$ if and only if there exists $y \in \mathbb{R}^{|I|\times|J|}_{\geq 0}$ such that (x, y) is a solution to $P(d)$. Furthermore, without loss of generality, we can assume $y \in \mathbb{N}^{|I|\times|J|}$.*

Proof. If (x, y) is a solution for $\mathrm{P}(d)$, then x is also feasible for $\mathrm{P_s}(d)$, as for any $S \subseteq J$ we have:

$$\begin{aligned}\sum_{i\in N(S)} q \cdot x_i \geq \sum_{i\in N(S)} \sum_{j\in N(i)} y_{ij} &= \sum_{i\in N(S)} \Bigg(\sum_{j\in N(i)\cap S} y_{ij} + \sum_{j\in N(i)\setminus S} y_{ij} \Bigg) \\ \geq \sum_{i\in N(S)} \sum_{j\in N(i)\cap S} y_{ij} &= \sum_{j\in S} \sum_{i\in N(j)} y_{ij} \geq \sum_{j\in S} d_j .\end{aligned}$$

Now, suppose we are given a solution x for $\mathrm{P_s}(d)$. We claim that the maximum s-t-flow in $H_G := H_G(q \cdot x, d) = (V, R, c)$ has flow value $d(J)$, where G is the bipartite graph of the instance $\mathcal{I}$. Note that, given an s-t-flow f with flow value $d(J)$, due to the flow conservation and the capacity constraints we can define a solution (x, y) to $\mathrm{P}(d)$ where $y_{ij} := f(i, j)$ for all $(i, j) \in R_G$. Further, the flow value of any s-t-flow can never be larger than $d(J)$ as the s-t-cut $(S^\star, T^\star)$ with $T^\star := \{t\}$ has cut capacity $d(J)$. Thus, it suffices to show that a maximum s-t-flow in H_G has flow value no less than $d(J)$. To this end, let (S, T) be a finite s-t-cut in H_G. Let $J' := J \setminus S$, possibly being the empty set. If any location in the neighborhood of J' is contained in S, the s-t-cut contains an arc with infinite capacity. Hence, assume $N^-_{H_G}(J') \cap S = \varnothing$ so that $N^-_{H_G}(J') \subseteq T$. Since x is a solution to $\mathrm{P_s}(d)$, we obtain, for any subset $Q \subseteq J$,

$$\sum_{i \in N^-_{H_G}(Q)} q \cdot x_i = \sum_{i \in N_G(Q)} q \cdot x_i \geq \sum_{j \in Q} d_j.$$

We get

$$c(S,T) \geq \sum_{j \in J \cap S} d_j + \sum_{i \in N^-_{H_G}(J')} q \cdot x_i \geq \sum_{j \in J \cap S} d_j + \sum_{j \in J'} d_j = \sum_{j \in J} d_j.$$

Thus, every s-t-cut has capacity larger or equal to $d(J)$ and by Theorem 2.6 we obtain the desired result for $y \in \mathbb{R}^{|I| \times |J|}_{\geq 0}$. Furthermore, the capacities of the arcs in R_s and R_t of the network H_G are integral. Thus, there exists an integral flow f in H_G if and only if there exists a continuous flow f' in H_G and integrality of y can be assumed without loss of generality, cf. [AMO93]. □

In particular, Theorem 3.8 shows that the interpretation of the variables x does not change between $\mathrm{P}(d)$ and $\mathrm{P_s}(d)$.

Observation 3.9. Taking a closer look at the proof of Theorem 3.8 we observe that the integrality of $x \in \mathbb{N}^{|I|}$ is only needed to show that an integral solution for $y \in \mathbb{R}^{|I| \times |J|}_{\geq 0}$ exists. Hence, the corresponding LP-relaxations of $\mathrm{P}(d)$ and $\mathrm{P_s}(d)$ are equivalent as well. Furthermore, Theorem 3.8 also applies for $d \in \mathbb{N}^{|J|}$.

At first glance, this reformulation of Min q-MSMC seems to be of limited use, as we increase the number of constraints exponentially, i.e., we obtain a *non-compact* formulation of the problem. Nevertheless, this new formulation also comes with a decrease in the number of variables and will become helpful when introducing demand uncertainty in Section 3.2.

In the remainder of this section, we analyze the complexity of q-MSMC for different values of $q \in \mathbb{N}_{>0}$. We see that, while there are polynomial time algorithms for the cases $q = 1$ and $q = 2$, we shall not hope for such an algorithm if $q \geq 3$. For $q \in \{1, 2\}$, the polynomial time solvability is directly given by polynomial time solvability of Min Multiset Multicover(K) for $K \leq 2$ [Sch02]. Nevertheless, here we present strongly polynomial time algorithms exploiting the structure of our problem.

Proposition 3.10. *Min 1-MSMC can be solved in time $\mathcal{O}(|I| + |J|)$.*

Proof. Given an instance of Min 1-MSMC, in any solution, each client needs to be assigned a unique supplier. Thus, we initialize $x := 0$ and, for each client in some region $j \in J$, we put a single supplier in some location $i \in N(j)$. This yields a solution with $d(J)$ suppliers which is optimal by Observation 3.5. We can find this solution in time linear in $|I| + |J|$. □

Transforming an instance of Min 1-MSMC to an instance of Min Multiset Multicover(1) leads an instance with $|J|$ many elements and $|E|$ many sets, where E is the edge set of the given bipartite graph. The constructed instance can then be solved in time $\mathcal{O}(|J|)$ leading to a total running time of $\mathcal{O}(|E|)$. Hence, the procedure presented in Proposition 3.10 is evidently superior. For $q = 2$, we can still solve Min q-MSMC in strongly polynomial time. Specifically, we show how to compute an optimal solution using an algorithm for Min b-Edge Cover which is a generalization of Problem 2.2.

Problem 3.11 (Min b-Edge Cover)**.**
Instance: A graph $G = (V, E)$ and a vector $b \in \mathbb{N}_{>0}^{|V|}$.
Solution: A *b-edge cover* $x \in \mathbb{N}^{|E|}$, i.e., $\sum_{e \in \delta(v)} \mathrm{m}(v, e) \cdot x_e \geq b_v$ for $v \in V$.
Measure: The value $x(E)$.

As for Min Edge Cover, an instance of Min b-Edge Cover can be solved in polynomial time by first solving the corresponding instance of *Max b-Matching*. Then, the computed vector is augmented to an optimal solution in time $\mathcal{O}(|E|)$ [Sch02].

Problem 3.12 (Max b-Matching)**.**
Instance: A graph $G = (V, E)$ and a vector $b \in \mathbb{N}_{>0}^{|V|}$.
Solution: A *b-matching* $x \in \mathbb{N}^{|E|}$, i.e., $\sum_{e \in \delta(v)} \mathrm{m}(v, e) \cdot x_e \leq b_v$ for $v \in V$?
Measure: The value $x(E)$.

Theorem 3.13. *Min 2-MSMC can be solved in time $\mathcal{O}(|I|^{5/2}|J|^{5/2})$.*

Proof. Let an instance of Min 2-MSMC be given. Note that we may bound the number of clients d_j of any region $j \in J$ by the number of locations $|I|$: Suppose $d_j \geq |I|+1$ for some $j \in J$. Then, in any solution (x, y)

to $\mathrm{P}(d)$, there is some $i \in N(j)$ such that $y_{ij} \geq 2$. Thus, given an optimal solution $(x^\star, y^\star)$, we can find i' such that $y^\star_{i'j} \geq 2$. Removing one supplier from location i' now yields an optimal solution to the same instance except for the demand of region j being $d_j - 2$. Label this instance as $\mathcal{I}'$. On the other hand, we may also solve $\mathcal{I}'$ and then add an additional supplier to any location adjacent to j to get an optimal solution to $\mathcal{I}$. We can therefore decrease the demands of all regions j with $d_j \geq |I| + 1$ to $|I|$, respectively $|I| - 1$, by adding $\lceil 1/2(d_j - |I|) \rceil$ many suppliers to some location connected to j. This can be done in constant time for every region $j \in J$.

Now regard the following procedure. We transform a given instance of the modified Min 2-MSMC problem into an instance of Min Multiset Multicover(2): Each region $j \in J$ determines an element of the ground set and, for every 2-element multiset $\{j_1, j_2\}$ of J, we check whether there is a location $i \in N(j_1) \cap N(j_2)$. If so, we add the set $\{j_1, j_2\}$. This gives an instance with $|J|$ many elements and at most $|J|^2$ many sets in time $\mathcal{O}(|I||J|^2)$. In fact, this also yields an instance of a Min b-Edge Cover problem if we set up a vertex for each element j, an edge for each set connecting the vertices of the set and $b := d$, cf. [Sch02]. According to [Sch02] the corresponding Max b-Matching instance can be solved in time $\mathcal{O}(|I|^{5/2}|J|^{5/2})$ which determines the running time. □

In the following, we see that Min q-MSMC is a generalization of Min Set Cover(q). Since Min Set Cover(K) is an APX-complete problem for any fixed $K \geq 3$ (cf. Section 2.4), we obtain the subsequent result.

Theorem 3.14. *For any fixed $q \geq 3$, Min q-MSMC is* APX*-hard.*

Proof. We present an L-reduction from the APX-complete problem Min Set Cover(3). To that end, let an instance $\mathcal{I}$ of Min Set Cover(3) be given, i.e., let S be a set and $\mathcal{C}$ be a collection of subsets of S where $|C| \leq 3$ for all $C \in \mathcal{C}$. We create an instance $\mathcal{I}'$ of Min q-MSMC in the following way: Due to legibility, assume the subsets $C \in \mathcal{C}$ have unique indices $i_C \in \mathbb{N}$. Let $I := \{i_C : C \in \mathcal{C}\}$, $J := S$, and define the bipartite graph G by $N(i_C) := C$ for all $C \in \mathcal{C}$. Further, let $d_j := 1$ for all $j \in J$. Clearly, this construction can be accomplished in polynomial time.

Now, let $\mathcal{C}' \subseteq \mathcal{C}$ be an optimal solution to $\mathcal{I}$. Clearly, setting x_{i_C} to 1 if $C \in \mathcal{C}'$ and zero otherwise yields a solution to $\mathcal{I}'$ with $\sum_{C \in \mathcal{C}} x_{i_C} = \mathrm{OPT}(\mathcal{I})$. Therefore, we have $\mathrm{OPT}(\mathcal{I}') \leq \mathrm{OPT}(\mathcal{I})$.

On the other hand, in any solution x to $\mathcal{I}'$, we can assume that, for all $C \in \mathcal{C}$, $x_{i_C} \leq 1$ as $q \geq 3 \geq |N(i_C)|$. Thus, $\mathcal{C}' := \{C \in \mathcal{C} \colon x_{i_C} = 1\}$ is a solution to $\mathcal{I}$ with value $\sum_{C \in \mathcal{C}} x_{i_C}$ and we get

$$\begin{aligned} \mathrm{SOL}(\mathcal{I}, \mathcal{C}') - \mathrm{OPT}(\mathcal{I}) &= \sum_{C \in \mathcal{C}} x_{i_C} - \mathrm{OPT}(\mathcal{I}) \\ &\leq \mathrm{SOL}(\mathcal{I}', x) - \mathrm{OPT}(\mathcal{I}'). \end{aligned}$$

Observe that, if x is an optimal solution for $\mathcal{I}'$, we also get $\mathrm{OPT}(\mathcal{I}) \leq \mathrm{OPT}(\mathcal{I}')$. □

Corollary 3.15. *For any fixed $q \geq 3$, q-MSMC is strongly* NP*-complete.*

Proof. As a consequence of Theorem 3.8, for a given instance of q-MSMC, we may test if a vector $x \in \mathbb{N}^{|I|}$ is feasible by one Max Flow computation. As we can assume that $x_i \leq d(N(i))$ for $i \in I$, cf. Observation 3.5, we get that q-MSMC is contained in NP. Moreover, NP-hardness directly follows from the proof of Theorem 3.14 as a solution to $\mathcal{I}$ with at most $B \in \mathbb{N}$ sets leads to a solution to $\mathcal{I}'$ with value at most B in polynomial time and vice versa. □

Remark 3.2 reveals Min q-MSMC to be a special case of Min Multiset Multicover(q). It is well-known that Min Multiset Multicover can be approximated within a factor of $\mathrm{H}(s)$, where $s \in \mathbb{N}$ is the size of the largest multiset of an instance, compare [Dob82]. If we regard Min q-MSMC as a Min Multiset Multicover problem as in Remark 3.2, all multisets have fixed size q. We therefore automatically get an $\mathrm{H}(q)$-approximation for Min q-MSMC.

Observation 3.16. There is an $\mathrm{H}(q)$-approximation algorithm for Min q-MSMC.

This directly leads to an enhancement of Theorem 3.14 as Min q-MSMC is contained in APX.

Corollary 3.17. *For any fixed $q \geq 3$, Min q-MSMC is* APX*-complete.* □

After having introduced Min q-MSMC and having analyzed its complexity, we now concentrate on a robust version of the problem. Therefore, we often refer to Min q-MSMC as the *non-robust* version.

3.2. Problem Definition and Classification of the Robust Version

In this section, we extend the initial problem Min q-MSMC to include uncertainty in the number of clients d_j of each region $j \in J$. We apply concepts of robust optimization such as strict and adjustable robustness, see Section 2.6. Thus, we consider an uncertainty set $\mathcal{U} \subseteq \mathbb{N}^{|J|}$ of possible demand scenarios that we have to take into account while we do not know the true scenario yet. Hence, each scenario $\xi \in \mathcal{U}$ defines a single problem instance in the fashion of Min q-MSMC when denoting the number of clients of region j by $d_j := \xi_j$ for $j \in J$. But note that, for $j \in J$, $\xi_j = 0$ is feasible now to account for the possibility of absent clients in region j in some scenario ξ. We define the following decision problem for a fixed integer $q \in \mathbb{N}_{>0}$.

Problem 3.18 (Robust q-Multiset Multicover (Robust q-MSMC))**.**
Instance: Finite sets I, J with $I \cap J = \varnothing$, an uncertainty set $\mathcal{U} \subseteq \mathbb{N}^{|J|}$, a bipartite graph $G = (I \cup J, E)$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for every subset $S \subseteq J$ and every scenario $\xi \in \mathcal{U}$, we have

$$\sum_{i \in N(S)} q \cdot x_i \geq \sum_{j \in S} \xi_j ?$$

The minimization problem corresponding to Robust q-MSMC, called Robust Min q-MSMC, can be formulated as follows:

$$\min_{x} \quad \sum_{i \in I} x_i \tag{3.3a}$$

$$\text{s.t.} \quad \sum_{i \in N(S)} q \cdot x_i \geq \sum_{j \in S} \xi_j \quad \text{for } S \subseteq J, \xi \in \mathcal{U} \tag{3.3b}$$

$$x_i \in \mathbb{N} \quad \text{for } i \in I. \tag{3.3c}$$

The intuition of the robust version of Min q-MSMC is to allocate the suppliers among the locations, such that, in any scenario $\xi \in \mathcal{U}$, all clients may be served. However, which client is served by which location can be decided separately for every scenario. Taking a closer look at (3.3), the uncertain data only occurs on the right-hand side of the constraints. Thus, the equivalence to the following formulation is immediate.

$$\mathrm{P_s}(\mathcal{U}) \quad \min_x \quad \sum_{i \in I} x_i \tag{3.4a}$$

$$\text{s.t.} \quad \sum_{i \in N(S)} q \cdot x_i \geq \max_{\xi \in \mathcal{U}} \sum_{j \in S} \xi_j \quad \text{for } S \subseteq J \tag{3.4b}$$

$$x_i \in \mathbb{N} \quad \text{for } i \in I. \tag{3.4c}$$

We refer to this formulation as $\mathrm{P_s}(\mathcal{U})$ for a given uncertainty set $\mathcal{U}$. There should be no risk of confusion to $\mathrm{P_s}(d)$ defined on Page 41 because, in the robust case, the argument is a set instead of a vector. Nevertheless, note that $\mathrm{P_s}(\{d\}) = \mathrm{P_s}(d)$. We also call (3.4) the *(robust) set formulation* where we usually omit the adjective "robust" if the considered variant is clear from the context. All in all, we can reformulate the question posed in Problem 3.18: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for all subsets $S \subseteq J$, we have

$$\sum_{i \in N(S)} q \cdot x_i \geq \max_{\xi \in \mathcal{U}} \sum_{j \in S} \xi_j?$$

Remark 3.19. As the uncertainty set $\mathcal{U}$ is assumed to be compact and non-empty, cf. Section 2.6, we know that $\max_{\xi \in \mathcal{U}} \xi(S)$ exists for every $S \subseteq J$.

The above maximization term will appear more often in this thesis. Thus, we define:

Problem 3.20 (Max Robust Sum).
Instance: An uncertainty set $\mathcal{U} \subseteq \mathbb{N}^n$ and a subset $S \subseteq \{1, \ldots, n\}$.
Solution: A scenario $\xi \in \mathcal{U}$.
Measure: The value $\xi(S)$.

We refer to the decision version of Max Robust Sum as Robust Sum. It is not surprising that Robust Sum is an NP-hard problem as the uncertainty set $\mathcal{U}$ can be used to model the set of solutions of some other NP-complete problem. Nevertheless, we give a short formal proof of this result in which we apply the well-known Independent Set problem [GJ79; PY91].

Problem 3.21 (Independent Set).
Instance: A simple graph $G = (V, E)$ and a positive integer $B \leq |V|$.
Question: Does G contain an *independent set* of size at least B, i.e., a subset $V' \subseteq V$ with $|V'| \geq B$ and no two vertices in V' are neighbors?

Lemma 3.22. *Robust Sum is strongly* NP*-complete even for* $S = \{1, \ldots, n\}$.

Proof. Robust Sum is contained in NP by our assumptions on $\mathcal{U}$, cf. Section 2.6. To show NP-hardness, we consider an instance of Independent Set, i.e., a simple graph $G = (V, E)$ and an integer $B \in \mathbb{N}_{>0}$ with $B \leq |V|$. Let $V = \{1, \ldots, n\}$ for $n \in \mathbb{N}_{>0}$. We define

$$\mathcal{U} \coloneqq \{\xi \in \mathbb{B}^n : \xi_u + \xi_v \leq 1 \text{ for } [u, v] \in E\},$$

i.e., the set $\mathcal{U}$ is the set of all incidence vectors of independent sets of G. Furthermore, we set $S \coloneqq \{1, \ldots, n\}$. Then, a subset $V' \subseteq V$ is an independent set of size at least B if and only if there exists a scenario $\xi \in \mathcal{U}$ with $\xi(S) \geq B$. As the instance of Robust Sum can be constructed in polynomial time, the claim follows. ☐

Remark 3.23. For later reference, we note that Robust Sum is strongly NP-complete even if the bound $B \in \mathbb{N}_{>0}$ of the instance is a multiple of q for some fixed $q \in \mathbb{N}_{>0}$. To see this we apply the above construction to the graph G' that is the union of q copies of G, i.e.,

$$\mathcal{U} \coloneqq \{\xi \in \mathbb{B}^{qn} : \xi_{i \cdot n+u} + \xi_{i \cdot n+v} \leq 1 \text{ for } [u, v] \in E \text{ and } i \in \{0, \ldots, q-1\}\}$$

with $S \coloneqq \{1, \ldots, qn\}$. Then, an independent set of G of size at least B leads to an independent set of G' of size at least $q \cdot B$ and, hence, to a scenario $\xi \in \mathcal{U}$ with $\xi(S) \geq q \cdot B$. On the other hand, let a scenario $\xi \in \mathcal{U}$ with $\xi(S) \geq q \cdot B$ be given. Then, ξ encodes an independent set in G' of size at least $q \cdot B$. There must exist some $i' \in \{0, \ldots, q-1\}$ with $\sum_{v \in V} \xi_{i' \cdot n+v} \geq B$ as otherwise $\xi(S) < q \cdot B$. Then, $V' \coloneqq \left\{v \in V : \xi'_{i' \cdot n+v} = 1\right\}$ is an independent set of G of size at least B. Note that this idea also shows that Independent Set remains NP-complete if the given bound is a multiple of q. We will make use of this restriction in Chapter 5.

Analogously to the proof of Lemma 3.22, we can show that Max Robust Sum is APX-hard by an L-reduction from the APX-complete Max Independent

Set(3) problem, where the degree of the input graph is bounded by 3 [PY91; BF95]. On the other hand, we have:

Observation 3.24. Max Robust Sum with polynomial time optimization uncertainty is contained in P. In particular, this holds if the uncertainty sets are polynomial time enumerable. Altogether, polynomial time solvability is given for discrete, interval, budgeted, and Γ-uncertainty, cf. Page 31.

As in Section 3.1 we aim to obtain an equivalent robust allocation formulation for Robust Min q-MSMC. A first idea is to formulate the following mixed integer program.

$$\min_{x,\,y} \quad \sum_{i\in I} x_i \tag{3.5a}$$

$$\text{s.t.} \quad \sum_{i\in N(j)} y_{ij} \geq \xi_j \qquad \text{for } j\in J, \xi\in\mathcal{U} \tag{3.5b}$$

$$\sum_{j\in N(i)} y_{ij} \leq q\cdot x_i \quad \text{for } i\in I \tag{3.5c}$$

$$y_{ij} \geq 0 \qquad \text{for } i\in I, j\in J \tag{3.5d}$$

$$x_i \in \mathbb{N} \qquad \text{for } i\in I. \tag{3.5e}$$

Consider the worst-case vector ξ^{wc} with $\xi^{\mathrm{wc}}_j := \max_{\xi\in\mathcal{U}} \xi_j$. Since an optimal solution $(x^\star, y^\star)$ to (3.5) needs to be feasible for $\mathrm{P}(\xi)$ for every scenario ξ (see Page 39), we can compute $(x^\star, y^\star)$ by solving $\mathrm{P}(\xi^{\mathrm{wc}})$. In general $\mathrm{P}(\xi^{\mathrm{wc}}) = \mathrm{P}_{\mathrm{s}}(\{\xi^{\mathrm{wc}}\}) = \mathrm{P}_{\mathrm{s}}(\mathcal{U})$ does not hold, e.g., if $\mathcal{U}$ is a budgeted uncertainty set (cf. Section 3.4.3), and we see that we need to apply a relaxed approach.

Actually, computing a global allocation y is far too conservative and applying strict robustness is unrewarding. Moreover, (3.5) does not match the intuition of Robust Min q-MSMC as we have to fix y_{ij} before the actual scenario is revealed. Recalling the interpretation of the variable y_{ij} on Page 47, it is meaningful to fix y_{ij} only after the realization of the true scenario ξ is known. Thus, we merely need to settle the decision over the number of suppliers x_i needed in every location $i\in I$ before the realization becomes apparent. Additionally, we need to ensure the existence of an allocation y of suppliers to clients. Therefore, we apply the concept of adjustable robustness with x representing the non-adjustable variables and y corresponding to the adjustable variables, cf. Section 2.6.1. Then, our aim is to find $x\in\mathbb{N}^{|I|}$ minimizing $x(I)$ such that, for every $\xi\in\mathcal{U}$, there exist $y(\xi)$ with $(x, y(\xi))$ being feasible for $\mathrm{P}(\xi)$. This approach leads to the adjustable robust formulation $\mathrm{P}(\mathcal{U})$ that we also call the *(robust) allocation formulation*:

$$\begin{aligned}
\mathrm{P}(\mathcal{U}) \quad \min_{x,y} \quad & \sum_{i \in I} x_i && && \text{(3.6a)}\\
\text{s.t.} \quad & \sum_{i \in N(j)} y(\xi)_{ij} \geq \xi_j && \text{for } j \in J, \xi \in \mathcal{U} && \text{(3.6b)}\\
& \sum_{j \in N(i)} y(\xi)_{ij} \leq q \cdot x_i && \text{for } i \in I, \xi \in \mathcal{U} && \text{(3.6c)}\\
& y(\xi)_{ij} \geq 0 && \text{for } i \in I, j \in J, \xi \in \mathcal{U} && \text{(3.6d)}\\
& x_i \in \mathbb{N} && \text{for } i \in I. && \text{(3.6e)}
\end{aligned}$$

Again, note the difference between $\mathrm{P}(\mathcal{U})$ and $\mathrm{P}(d)$ defined on Page 39. The variables $y(\xi)$ are also called adjustable variables as they can adjust themselves in the second stage – when the scenario ξ reveals – to a given solution x from the first stage. Problems of this type are also known as *two-stage adaptive* optimization problems [BG10; BGS11].

Now, we are able to prove the equivalence between the robust set formulation and the robust allocation formulation.

Theorem 3.25. *$P_s(\mathcal{U})$ and $P(\mathcal{U})$ are equivalent formulations of Robust Min q-MSMC. In particular, it holds that $x \in \mathbb{N}^{|I|}$ is feasible for $P_s(\mathcal{U})$ if and only if, for every scenario $\xi \in \mathcal{U}$, there is $y(\xi) \in \mathbb{R}_{\geq 0}^{|I|\times|J|}$ such that (x, y) with $y := (y(\xi^1), y(\xi^2), \ldots)$ is feasible for $P(\mathcal{U})$. Furthermore, without loss of generality, we can assume $y(\xi) \in \mathbb{N}^{|I|\times|J|}$ for every $\xi \in \mathcal{U}$.*

Proof. Since the objective functions are identical, it remains to show that any solution (x, y) of $\mathrm{P}(\mathcal{U})$ yields a solution x of $\mathrm{P}_{\mathrm{s}}(\mathcal{U})$ and vice versa. Thus, let $\mathcal{U} = \{\xi^1, \xi^2, \ldots\}$ and (x, y) be feasible for $\mathrm{P}(\mathcal{U})$ with $y = (y(\xi^1), y(\xi^2), \ldots)$. Fix a scenario $\xi \in \mathcal{U}$. Then, $(x, y(\xi))$ is feasible for $\mathrm{P}(\xi)$. Due to the equivalence of the formulations in the non-robust version by Theorem 3.8 and Observation 3.9, we get that x fulfills

$$\sum_{i \in N(S)} q \cdot x_i \geq \sum_{j \in S} \xi_j$$

for $S \subseteq J$. As this argument holds true for any fixed scenario ξ, we obtain that x is feasible for $\mathrm{P}_{\mathrm{s}}(\mathcal{U})$.

On the other hand, given a solution x of $\mathrm{P_s}(\mathcal{U})$, for any fixed scenario ξ, there exists $y(\xi) \in \mathbb{N}^{|I|\times|J|}$ such that $(x, y(\xi))$ is feasible for $\mathrm{P}(\xi)$ due to Theorem 3.8. In total, we obtain that (x, y) with $y = (y(\xi^1), y(\xi^2), \ldots)$ is feasible for $\mathrm{P}(\mathcal{U})$. □

Observation 3.26. Similar to Observation 3.9 we see the following:

(a) The corresponding LP-relaxations of $\mathrm{P}(\mathcal{U})$ and $\mathrm{P_s}(\mathcal{U})$ are equivalent.

(b) If the uncertainty set $\mathcal{U}$ has exponentially many scenarios, both formulations are non-compact.

The equivalence of the two formulations allows for an interpretation of Robust Min q-MSMC as a real world problem. We briefly address this application in the following.

Example 3.27. A possible implementation of Robust Min q-MSMC lies in the strategic facility and resource planning of emergency services. Consider a map area with a given set of potential ambulance stations that forms the set of locations I. Suppose we discretize the map area so that it consists of a finite set of points. For each such point on the map, we can find the subset of potential stations that are located within a reasonable response time, e.g., a car travel time of 15 minutes. Then, points in the map sharing the same subset are aggregated and form a region $j \in J$. In that manner, we construct a bipartite graph G where each location is adjacent to its close regions. Now, we aim to plan the necessary ambulances and, hence, also the necessary ambulance stations for a given shift type, e.g., a general night shift. Strategically, we assume that any ambulance can cover q emergencies in this shift type for a given fixed integer $q \in \mathbb{N}_{>0}$. As the occurrence of emergencies is uncertain, we introduce an uncertainty set $\mathcal{U} \in \mathbb{N}^{|J|}$ comprising all possible emergency scenarios in the given shift type. To that end, for a scenario $\xi \in \mathcal{U}$, the value $\xi_j \in \mathbb{N}$ describes the number of emergencies happening in region j in scenario ξ. This construction forms an instance of Robust Min q-MSMC where the variable x_i describes the number of ambulances needed in location i. Let $(\bar{x}, \bar{y})$ be feasible for $\mathrm{P}(\mathcal{U})$. Then, no matter which scenario ξ occurs, there is an allocation $\bar{y}(\xi)$ of locations to regions saying that the ambulances in location i cover $\bar{y}(\xi)_{ij}$ emergencies in region j in case scenario ξ occurs. Moreover, each emergency occurring in scenario ξ is covered and the capacity $q \cdot \bar{x}_i$ of each location is observed. Altogether, we make sure that, despite the uncertainty of occurring emergencies, every emergency can be reached by some free ambulance within a reasonable time frame. Including the limited number of available ambulances in practice, we aim for a solution using a minimum number of ambulances. For more details on this approach and computational results we refer to [KSS19]. Therein, the authors focus on budgeted uncertainty. $\triangleleft$

The above application of Robust Min q-MSMC leads to an interpretation of the problem as a robust facility location problem similar to the non-robust variant. For a survey on robust facility location problems we refer to [Sny06]. However, as the reasons mentioned in Section 3.1 remain valid also for the robust problem by Theorem 3.25, we refrain from going into detail here.

Similar to the non-robust case, formulation $\mathrm{P}(\mathcal{U})$ contains more variables than $\mathrm{P_s}(\mathcal{U})$. On the other hand, the number of constraints in $\mathrm{P}(\mathcal{U})$ is given by $|\mathcal{U}| \cdot (|I| + |J|)$ compared to $2^{|J|}$ constraints in $\mathrm{P_s}(\mathcal{U})$. Hence $\mathrm{P}(\mathcal{U})$ might be meaningful if the number of scenarios can be bounded from above while $\mathrm{P_s}(\mathcal{U})$ might be of use for large uncertainty sets, compare [KSS19].

Further, we see that our initial link to Multiset Multicover(q) given in Remark 3.2 seems to be lost when including robustness since the allocation variables can be specified in a subsequent step. In the robust case, the value $x_i \in \mathbb{N}$ has to be specified in advance for all locations $i \in I$ so that, for each possible scenario $\xi \in \mathcal{U}$, there exists, for each location i, a selection of x_i multisets of $N(i)$ of size q that satisfy the upcoming demand. Thus, here we have a special robust Multiset Multicover problem. In Section 4.2.3, we present one possibility of regaining a Multiset Multicover representation, but in general the polynomial relation between the input sizes is not retained. Therefore, the inclusion of robustness leads to a new problem in comparison to Section 3.1 which we investigate further in the following. Before we concentrate on the complexity of Robust q-MSMC, we state the analogue to Observation 3.5.

Observation 3.28. By our assumptions on $\mathcal{U}$ in Section 2.6, we get that an instance $\mathcal{I}$ of Robust Min q-MSMC is feasible if and only if, for every region $j \in J$, there is some adjacent location $i \in N(j)$. Again, we restrict our considerations to feasible instances as these can be identified in time $\mathcal{O}(|J|)$. Then, a given instance $\mathcal{I}$ has a finite optimal solution $x^\star \in \mathbb{N}^{|I|}$ with

$$\sum_{i \in I} x_i^\star \geq \left\lceil \frac{\max_{\xi \in \mathcal{U}} \xi(J)}{q} \right\rceil \quad \text{and} \quad x_i^\star \leq \left\lceil \frac{\max_{\xi \in \mathcal{U}} \xi(N(i))}{q} \right\rceil$$

for all $i \in I$.

In the following, we utilize the Min Dominating Set(K) problem to show APX-hardness of Robust Min q-MSMC. The former problem is APX-complete for $K \geq 3$, cf. [LY94; Aus+02], and defined as follows:

Problem 3.29 (Min Dominating Set(K)).
Instance: A simple graph $G = (V, E)$ with $\Delta \leq K$.
Solution: A *dominating set* $V' \subseteq V$ for G, i.e., for all $u \in V \setminus V'$, there is $v \in V'$ dominating u.
Measure: The cardinality of V'.

Theorem 3.30. *For any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-MSMC is* APX*-hard.*

Proof. We show that there exists an L-reduction from Min Dominating Set(3) to Robust Min q-MSMC. To this end, let a simple graph $G = (V, E)$ with $\Delta_G \leq 3$ be given and call the instance $\mathcal{I}$. Without loss of generality, let $V = \{1, \ldots, n\}$ for $n \in \mathbb{N}_{>0}$. To construct an instance $\mathcal{I}'$ of Robust Min q-MSMC we set $I := V$ and $J := \{n+1, \ldots, 2n\}$. For every edge $[u, v] \in E$, we add the edge $[u, n+v]$ and the edge $[v, n+u]$ to the bipartite graph $G' = (I \cup J, E')$. Additionally, for every $v \in V$, the edge $[v, n+v]$ is added to G'. Finally, let

$$\mathcal{U} := \left\{\xi \in \mathbb{N}^{|J|} : 0 \leq \xi \leq 1, \xi(J) \leq 1\right\} = \{0, e_1, \ldots, e_n\} \subseteq \mathbb{N}^n \tag{3.7}$$

so that $\max_{\xi \in \mathcal{U}} \xi(S) = 1$ for every non-empty subset $S \subseteq J$.

Let $V' \subseteq V = I$ be an optimal solution to $\mathcal{I}$. Then, we set $x_i := 1$ for all $i \in V'$ and zero otherwise. Fix a non-empty subset $S \subseteq J$ as otherwise there is nothing to prove. We need to show that

$$\sum_{i \in N_{G'}(S)} q \cdot x_i \geq 1.$$

Thus, we argue that at least one value x_i for $i \in N_{G'}(S)$ is set to 1, i.e., $N_{G'}(S) \cap V' \neq \varnothing$. Choose an arbitrary element $n+v \in S$ with $v \in \{1, \ldots, n\}$. The set V' is a dominating set for G, so we have $v \in V'$ or there is $u \in V'$ adjacent to v in G. In the former case, $x_v = 1$ and $v \in N_{G'}(S)$ since G' contains the edge $[v, n+v]$. In the latter case, $x_u = 1$ and $u \in N_{G'}(S)$ since G' contains the edge $[u, n+v]$. Thus, $q \cdot x(N_{G'}(S)) \geq 1$ holds true in any case. As S was arbitrary, x is a solution to $\mathcal{I}'$ with $\mathrm{SOL}(\mathcal{I}', x) = |V'|$ and we obtain

$$\mathrm{OPT}(\mathcal{I}') \leq \mathrm{OPT}(\mathcal{I}). \tag{3.8}$$

Conversely, suppose that $x \in \mathbb{N}^{|I|}$ is a solution to $\mathcal{I}'$. Since $\max_{\xi \in \mathcal{U}} \xi(S) = 1$ for all $S \subseteq J$, $S \neq \varnothing$, we can assume that $x_i \leq 1$ for all $i \in I$. Otherwise, we can easily improve our given solution. The set V' is defined to contain all vertices $v \in V$ such that $x_v = 1$. We claim that V' is a dominating set for G. To this end, choose a vertex $u \in V$ and consider the set $S := \{n + u\} \subseteq J$. Since x is feasible, there is $v \in N_{G'}(S)$ with $x_v = 1$, i.e., $v \in V'$. Since $v \in N_{G'}(S)$, either $v = u$ or the vertices u and v are adjacent in G by construction of G'. Hence, every vertex in G is dominated by some vertex in V'. We get $\mathrm{SOL}(\mathcal{I}, V') \leq \mathrm{SOL}(\mathcal{I}', x)$ and, in particular, $\mathrm{OPT}(\mathcal{I}) \leq \mathrm{OPT}(\mathcal{I}')$. With (3.8) we obtain that

$$\mathrm{SOL}(\mathcal{I}, V') - \mathrm{OPT}(\mathcal{I}) \leq \mathrm{SOL}(\mathcal{I}', x) - \mathrm{OPT}(\mathcal{I}'). \qquad \square$$

Observe that the previous proof also holds for Robust Min q-MSMC with polynomial time enumeration uncertainty by (3.7). Further, it directly leads to the following:

Corollary 3.31. *For any fixed $q \in \mathbb{N}_{>0}$, Robust q-MSMC is strongly* NP*-hard.* $\square$

Corollary 3.32. *For any fixed $q \in \mathbb{N}_{>0}$ and for any fixed $\varepsilon > 0$, Robust Min q-MSMC cannot be approximated within a factor of $(1 - \varepsilon) \ln |J|$ or $(1 - \varepsilon) \ln \sum_{\xi \in \mathcal{U}} \xi(J)$ unless* $\mathrm{P} = \mathrm{NP}$.

Proof. By [DS14; Mos15] the Min Dominating Set problem cannot be approximated within $(1 - \varepsilon) \ln n$ for any fixed $\varepsilon > 0$ unless $\mathrm{P} = \mathrm{NP}$, where $n \in \mathbb{N}$ is the number of vertices of the given simple graph. The proof of Theorem 3.30 reveals that, for an instance $\mathcal{I}$ of Min Dominating Set with n vertices, we can construct in polynomial time an instance $\mathcal{I}'$ of Robust Min q-MSMC with $|J| = \sum_{\xi \in \mathcal{U}} \xi(J) = n$ such that every solution x to $\mathcal{I}$ with $x \leq 1$ corresponds to a solution to $\mathcal{I}'$ with identical objective values and vice versa. $\square$

Additionally, when inspecting the proof of Theorem 3.30, we see that the result may translate to various classes of uncertainty sets.

Remark 3.33. Let $\mathcal{C}$ be a class of uncertainty sets such that, for $n \in \mathbb{N}_{>0}$, there is $\mathcal{U} \in \mathcal{C}$ with $\mathcal{U} \subseteq \mathbb{R}^n$ and $\max_{\xi \in \mathcal{U}} \xi(S) = 1$ for every non-empty subset $S \subseteq \{1, \dots, n\}$. From the proof of Theorem 3.30 we get that, for any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-MSMC with uncertainty $\mathcal{C}$ is APX-hard and its decision version is strongly NP-hard. Examples of such classes can be found in Section 3.4.3 and Section 3.4.6.

Note that we did not prove NP-completeness of Robust q-MSMC. Given an instance of the problem, a polynomial time algorithm for checking feasibility of a vector $\bar{x} \in \mathbb{N}^{|I|}$ is not obvious at first sight as we potentially have to check exponentially many subsets $S \subseteq J$ or we need to compute $y(\xi)$ for every scenario $\xi \in \mathcal{U}$. This leads to the analysis of the following complementary decision problems.

Problem 3.34 (Feasibility for Robust Min q-MSMC (Feasibility))**.**
Instance: Finite sets I, J with $I \cap J = \varnothing$, a vector $\bar{x} \in \mathbb{N}^{|I|}$, an uncertainty set $\mathcal{U} \subseteq \mathbb{N}^{|J|}$, and a bipartite graph $G = (I \cup J, E)$.
Question: Does $q \cdot \bar{x}(N(S)) \geq \max_{\xi \in \mathcal{U}} \xi(S)$ hold for every subset $S \subseteq J$?

Problem 3.35 (Separation for Robust Min q-MSMC (Separation))**.**
Instance: Finite sets I, J with $I \cap J = \varnothing$, a vector $\bar{x} \in \mathbb{N}^{|I|}$, an uncertainty set $\mathcal{U} \subseteq \mathbb{N}^{|J|}$, and a bipartite graph $G = (I \cup J, E)$.
Question: Is there a subset $S \subseteq J$ such that $q \cdot \bar{x}(N(S)) < \max_{\xi \in \mathcal{U}} \xi(S)$?

For an instance $\mathcal{I}$ of Separation or Feasibility, the corresponding instance $\mathcal{I}'$ of Robust Min q-MSMC is obtained by disregarding the given vector $\bar{x}$. We start with considering Feasibility and Separation restricted to specific instances that will become important in Chapter 4.

Theorem 3.36. *Feasibility and Separation with polynomial time enumeration uncertainty are both contained in* P*.*

Proof. Consider an instance $\mathcal{I}$ of Separation with bipartite graph G and let $\xi \in \mathcal{U}$. Let (S, T) be a minimum s-t-cut in the network $H_G(q \cdot \bar{x}, \xi)$ from Definition 3.7. This s-t-cut can be computed in time $\mathcal{O}(N^2 M)$ with $N := |I| + |J|$ and $M := |I| + |J| + |E|$: We first compute a maximum flow in time $\mathcal{O}(N^2 M)$ and then find a corresponding minimum s-t-cut in time $\mathcal{O}(M)$, cf. [AMO93]. We know that $c(S, T) < \infty$ and, thus, no arc with infinite capacity contributes to the s-t-cut. If $c(S,T) < \xi(J)$, we get that

$$c(S,T) = \sum_{i \in I \cap T} q \cdot \bar{x}_i + \sum_{j \in J \cap S} \xi_j < \sum_{j \in J} \xi_j \quad \Leftrightarrow \quad \sum_{i \in I \cap T} q \cdot \bar{x}_i < \sum_{j \in J \cap T} \xi_j .$$

Furthermore, we have $N_G(J \cap T) \subseteq I \cap T$ as otherwise $c(S,T) = \infty$. Therefore,

$$\sum_{i \in N_G(J \cap T)} q \cdot \bar{x}_i < \sum_{j \in J \cap T} \xi_j$$

and we obtain the solution $J \cap T$ for $\mathcal{I}$. Otherwise, $c(S,T) = \xi(J)$ and $\bar{x}$ is feasible for scenario ξ and we move on to the next scenario. As $\mathcal{U}$ is polynomial time enumerable, we can decide in polynomial time whether $\bar{x}$ is feasible for the corresponding instance of Robust Min q-MSMC or whether there is a solution for $\mathcal{I}$. □

Corollary 3.37. *For any fixed $q \in \mathbb{N}_{>0}$, Robust q-MSMC with polynomial time enumeration uncertainty is strongly* NP*-complete.* □

Similar to Robust Sum, the uncertainty set in an instance of Feasibility or Separation can be used to model NP-complete problems, cf. Lemma 3.22 and Remark 3.23. Thus, in a similar manner we get the subsequent result for general instances of Separation.

Theorem 3.38. *For any fixed $q \in \mathbb{N}_{>0}$, Separation is strongly* NP*-complete even for $\mathcal{U} \subseteq \mathbb{B}^{|J|}$.*

Proof. Separation is contained in NP as, for an instance $\mathcal{I}$ and a scenario $\xi \in \mathcal{U}$, we can compute in polynomial time the maximum flow value in the network $H_G(q \cdot \bar{x}, \xi)$, where G is the corresponding bipartite graph of $\mathcal{I}$. To show NP-hardness we provide a polynomial time reduction from Independent Set. Let a simple graph $G = (V,E)$ and a positive integer $B \leq |V|$ represent an instance of such a problem. Suppose $V = \{1, \dots, n\}$ for $n \in \mathbb{N}_{>0}$. We construct an instance of Separation as follows: Let $I \coloneqq \{0\}$ and $J \coloneqq \{1, \dots, qn\}$. The edge set E' of the bipartite graph $G' = (I \cup J, E')$ consists of the edges $[0,j]$ for every $j \in J$. Furthermore, let $\bar{x}_0 \coloneqq B - 1$ and

$$\mathcal{U} \coloneqq \{\xi \in \mathbb{B}^{qn} \colon \xi_{i \cdot n+u} + \xi_{i \cdot n+v} \leq 1 \text{ for } [u,v] \in E \text{ and } i \in \{0, \dots, q-1\}\} .$$

Thus, the uncertainty set $\mathcal{U}$ comprises all incidence vectors of independent sets of the graph that is the union of exactly q copies of G.

If $V' \subseteq V$ is an independent set of G with $|V'| \geq B$, we consider the scenario ξ' with $\xi'_{i \cdot n+v} = 1$ if $v \in V'$ and $i \in \{0, \dots, q-1\}$ and zero otherwise. By construction of $\mathcal{U}$ we have $\xi' \in \mathcal{U}$ and we get

$$\max_{\xi \in \mathcal{U}} \sum_{j \in J} \xi_j \geq \sum_{j \in J} \xi'_j = q \cdot |V'| \geq q \cdot B > q \cdot (B-1) = q \cdot \bar{x}_0.$$

Thus, J is a solution for the constructed instance of Separation. On the other hand, let S be a subset of J with

$$\max_{\xi \in \mathcal{U}} \sum_{j \in S} \xi_j > \sum_{i \in N_{G'}(S)} q \cdot \bar{x}_i.$$

Then, S is non-empty and we get $q \cdot \bar{x}(N_{G'}(S)) = q \cdot (B-1)$. Thus, there is a scenario $\xi' \in \mathcal{U}$ with at least $q \cdot (B-1)+1$ non-zero entries. There exists some $i' \in \{0, \dots, q-1\}$ with $\sum_{v \in V} \xi'_{i' \cdot n+v} \geq B$ as otherwise we get $\xi'(J) \leq q \cdot (B-1)$. Let $V' := \left\{v \in V : \xi'_{i' \cdot n+v} = 1\right\}$. Then, $|V'| \geq B$ and V' is an independent set in G by construction of $\mathcal{U}$. □

Corollary 3.39. *For any fixed $q \in \mathbb{N}_{>0}$, Feasibility is strongly* co-NP-*complete even for $\mathcal{U} \subseteq \mathbb{B}^{|J|}$.* □

Together with the polynomial time equivalence of optimization and separation of linear programming, cf. Section 2.5, and our general assumptions on $\mathcal{U}$ in Section 2.6, Theorem 3.38 gives:

Corollary 3.40. *The LP-relaxation of Robust Min q-MSMC, i.e., asking for $x \in \mathbb{R}^{|I|}_{\geq 0}$ such that x is feasible for the LP-relaxation of $P_s(\mathcal{U})$, is* NP-*hard to solve.* □

In this section, we have seen that, in general, we cannot even hope for a polynomial time algorithm to solve the LP-relaxation of Robust Min q-MSMC. In the following, we nevertheless analyze solution approaches for Robust Min q-MSMC. Even if these approaches might not yield an optimal solution to every instance of the problem in an acceptable amount of time, they can still be applied to small instances and lead to new insights into the problem's structure.

3.3. Solving Robust Min q-Multiset Multicover

In the previous section, we have shown that Robust Min q-MSMC is an APX-hard problem for any fixed value of $q \in \mathbb{N}_{>0}$. Now, we are aiming for solution techniques. Besides the exponential time exact algorithms for Multiset Multicover problems published in [Hua+10], Robust Min q-MSMC can be solved using constraint generation, cf. Section 2.5. For adjustable robust problems, this is a widely applied practice, cf. [Gab+14; SWW19] and references therein. As both formulations of the problem as (mixed) integer linear programs contain a large, potentially exponential, number of constraints, this approach appears to be intuitive and suitable.

Let an instance of Robust Min q-MSMC be given. If we focus on the subset formulation $\mathrm{P_s}(\mathcal{U})$ and the classical constraint generation approach, at any point in the constraint generation process, a collection of subsets $\mathcal{S} \subseteq 2^J$ is given. We solve a relaxed problem obtained from only considering the constraints corresponding to sets $S \in \mathcal{S}$ in $\mathrm{P_s}(\mathcal{U})$. In the separation step, given an optimal solution $\bar{x}$ to the relaxed problem, we are looking for a solution $S \subseteq J$ to the corresponding instance of the Separation problem, cf. Problem 3.35. If such a set S is found, it must hold that $S \notin \mathcal{S}$ and $\mathcal{S}$ is updated by adding S. Then, the relaxed problem is solved once more. If there exists no such set S, we know that $\bar{x}$ is optimal for the given instance of Robust Min q-MSMC. Initially, the set $\mathcal{S}$ is the empty set yielding the optimal solution $\bar{x} = 0$ in the relaxed problem. Likewise, it is possible to start with a non-empty sets $\mathcal{S}$.

The important step of constraint generating methods is an efficient way to solve the occurring separation problems. Hence, we begin with defining names for the solutions to these problems.

Definition 3.41 (Violating Subset/Scenario)**.** Let an instance of Separation be given. A solution $S \subseteq J$ to this instance is called *violating subset* (with respect to $\bar{x}$). A scenario $\xi \in \mathcal{U}$ such that there is no $y \geq 0$ satisfying

$$\sum_{i \in N(j)} y_{ij} \geq \xi_j \text{ for } j \in J \text{ and } \sum_{j \in N(i)} y_{ij} \leq q \cdot \bar{x}_i \text{ for } i \in I \tag{3.9}$$

is called *violating scenario* (with respect to $\bar{x}$).

Note that in the definition of Separation in Problem 3.35 we can equivalently ask for the existence of a violating scenario instead of a violating subset by Theorem 3.8. Hence, the above exemplary constraint generation procedure can similarly be applied to the allocation formulation $\mathrm{P}(\mathcal{U})$ using a subset $\mathcal{U}' \subseteq \mathcal{U}$ of the scenarios. Here, the relaxed problem is given by $\mathrm{P}(\mathcal{U}')$. If, for an optimal solution $(\bar{x}, \bar{y})$ to $\mathrm{P}(\mathcal{U}')$, the vector $\bar{x}$ is not feasible for the given instance of Robust Min q-MSMC, there exists a violating scenario $\xi \in \mathcal{U} \setminus \mathcal{U}'$ and ξ is added to $\mathcal{U}'$. Observe that this approach generates new variables in every iteration.

Using Farkas' Lemma, cf. [Sch98; GLS93], a scenario $\xi \in \mathcal{U}$ is violating if and only if there are vectors $\mu \in \mathbb{R}^{|I|}_{\geq 0}$ and $\nu \in \mathbb{R}^{|J|}_{\geq 0}$ with $\mu_i \geq \nu_j$ for all locations $i \in I$ and regions $j \in N(i)$ and

$$\sum_{i \in I} q \cdot \bar{x}_i \cdot \mu_i < \sum_{j \in J} \xi_j \cdot \nu_j. \tag{3.10}$$

In the following, we investigate the correspondence of violating subsets and violating scenarios. Due to Theorem 3.25 we immediately obtain the formal statements corresponding to the introduction of this section.

Observation 3.42. Let an instance of Separation be given and let $\mathcal{I}$ be the corresponding instance of Robust Min q-MSMC. With respect to $\bar{x} \in \mathbb{N}^{|I|}$ we get:

(a) There is a violating scenario $\xi \in \mathcal{U}$ if and only if there is a violating subset $S \subseteq J$. In this case, the vector $\bar{x} \in \mathbb{N}^{|I|}$ is not feasible for $\mathcal{I}$.

(b) The vector $\bar{x} \in \mathbb{N}^{|I|}$ is feasible for $\mathcal{I}$ if and only if there does not exist a violating scenario.

(c) The vector $\bar{x} \in \mathbb{N}^{|I|}$ is feasible for $\mathcal{I}$ if and only if there does not exist a violating subset.

Our aim is to strengthen the statements of Observation 3.42 and to analyze whether it is possible to switch between both formulations during the constraint generation. From the proof of Theorem 3.36 we obtain that, for an instance $\mathcal{I}$ of Robust Min q-MSMC and a violating scenario ξ with respect to a vector $\bar{x} \in \mathbb{N}^{|I|}$, every minimum s-t-cut in the network $H_G(q{\cdot}\bar{x}, \xi)$ leads to a violating subset, where G is the corresponding bipartite graph. Thus, such a violating subset can be found in polynomial time and there might be more than one violating subset corresponding to a violating scenario ξ. For instance, in the first iteration with $\mathcal{U}' = \varnothing$ every set $\{j\}$ with $\xi_j > 0$ is violating. For the other direction, we define:

Definition 3.43 (Violating Extreme Scenario)**.** Let an instance of Separation be given. A scenario $\xi \in \mathcal{U}$ is called *violating extreme scenario* (with respect to $\bar{x}$) if there is a violating subset $S \subseteq J$ with respect to $\bar{x}$ such that $\xi \in \arg\max_{\xi' \in \mathcal{U}} \xi'(S)$.

First of all, we need to show that violating extreme scenarios are in fact violating. Moreover, we prove that it suffices to restrict ourselves to violating extreme scenarios when applying constraint generation to $\mathrm{P}(\mathcal{U})$.

Lemma 3.44. *Let an instance of Separation be given and let $\mathcal{I}$ be the corresponding instance of Robust Min q-MSMC. With respect to $\bar{x} \in \mathbb{N}^{|I|}$ we get:*

(a) Every violating extreme scenario $\xi \in \mathcal{U}$ is a violating scenario.

(b) There is a violating extreme scenario $\xi \in \mathcal{U}$ if and only if there is a violating subset $S \subseteq J$.

(c) The vector $\bar{x} \in \mathbb{N}^{|I|}$ is feasible for $\mathcal{I}$ if and only if there does not exist a violating extreme scenario.

Proof. Let $S \subseteq J$ be a violating subset with respect to $\bar{x}$ and let ξ be a violating extreme scenario with $\xi \in \arg\max_{\xi' \in \mathcal{U}} \xi'(S)$.

(a) Let (S', T') be an s-t-cut in $H_G(q \cdot \bar{x}, \xi)$ with $T' := N(S) \cup S$, where G is the corresponding bipartite graph of the given instances. Then, we get

$$c(S', T') = \sum_{i \in N(S)} q \cdot \bar{x}_i + \sum_{j \in J \setminus S} \xi_j < \sum_{j \in S} \xi_j + \sum_{j \in J \setminus S} \xi_j = \sum_{j \in J} \xi_j.$$

Thus, every maximum flow in $H_G(q \cdot \bar{x}, \xi)$ has flow value less than $\xi(J)$ and there is no $y \geq 0$ fulfilling (3.9) due to Theorem 3.8. Therefore, ξ is a violating scenario.

(b) Follows from (a), Observation 3.42, and Definition 3.43.

(c) Follows from (b) and Observation 3.42. □

We see that a violating subset S in turn leads to a set of violating extreme scenarios, namely the set $\arg\max_{\xi \in \mathcal{U}} \xi(S)$. In general, we cannot assume that a violating extreme scenario can be obtained in polynomial time as we have to solve an instance of Max Robust Sum, cf. Theorem 3.22. In Section 3.4, we consider several classes of uncertainty sets, some of which allow for a computation of a violating extreme scenario from a violating subset in polynomial time. For these special classes, it is possible to switch between both formulations in polynomial time. Altogether, each violating subset and violating extreme scenario corresponds to a set of violating extreme scenarios and violating subsets, respectively.

As Separation is NP-complete by Theorem 3.38, we propose the following mixed integer programs to solve this problem. If we are looking for a violating scenario, applying (3.10) leads to the following formulation:

$$\text{Sep}(\mathcal{U}) \qquad \min_{\mu,\nu,\xi} \quad \sum_{i\in I} q\cdot \bar{x}_i \cdot \mu_i - \sum_{j\in J} \xi_j \cdot \nu_j \tag{3.11a}$$

$$\text{s.t.} \qquad \mu_i \geq \nu_j \quad \text{for } i\in I, j\in N(i) \tag{3.11b}$$

$$\mu_i, \nu_j \geq 0 \quad \text{for } i\in I, j\in J \tag{3.11c}$$

$$\xi \in \mathcal{U}. \tag{3.11d}$$

The complexity of Sep($\mathcal{U}$) mainly depends on the structure of $\mathcal{U}$. For example, if $\mathcal{U}$ is polyhedral, Sep($\mathcal{U}$) turns into a (non-convex) quadratic mixed integer program. Observe that the optimal value of Sep($\mathcal{U}$) never exceeds zero as setting $\mu := 0$, $\nu := 0$ and choosing an arbitrary scenario $\xi \in \mathcal{U}$ yields a solution. On the other hand, if there is a solution (μ, ν, ξ) with solution value less than zero, we get that $(\lambda\mu, \lambda\nu, \xi)$ is feasible for every $\lambda > 0$. Hence, the problem is unbounded.

Theorem 3.45. *Let an instance of Separation be given. The optimal value of Sep($\mathcal{U}$) is zero if and only if $\bar{x}$ is feasible for the corresponding instance of Robust Min q-MSMC.*

Proof. For sufficiency fix a scenario $\xi \in \mathcal{U}$. If the optimal value of Sep($\mathcal{U}$) is zero, the following system of inequalities has no solution

$$\begin{aligned} \sum_{i\in I} q\cdot \bar{x}_i\cdot \mu_i < \sum_{j\in J} \xi_j \cdot \nu_j & \\ \mu_i \geq \nu_j & \qquad \text{for } i\in I, j\in N(i) \\ \mu_i, \nu_j \geq 0 & \qquad \text{for } i \in I, j\in J. \end{aligned}$$

Thus, by Farkas' Lemma, there is $y \geq 0$ fulfilling (3.9) and ξ is not violating. Feasibility of $\bar{x}$ follows by Observation 3.42 as this holds for every scenario $\xi \in \mathcal{U}$. Reversing these arguments also yields necessity. □

In addition, we want to regain a linear objective for Sep($\mathcal{U}$). The next lemma provides the basis for this result.

Lemma 3.46. *If there is a solution to Sep($\mathcal{U}$) with value less than zero, there also exists an integral solution with value less than zero and $\mu \in \mathbb{B}^{|I|}$, $\nu \in \mathbb{B}^{|J|}$.*

Proof. If there is a solution to Sep($\mathcal{U}$) with value less than zero, we get from Theorem 3.45 that there exists a (non-empty) violating subset $S \subseteq J$. Setting $v_j := 1$ if $j \in S$ and zero otherwise, $\mu_i := 1$ if $i \in N(S)$ and zero otherwise, and choosing a scenario $\xi \in \arg\max_{\xi' \in \mathcal{U}} \xi'(S)$ yields an integral solution (μ, ν, ξ) to Sep($\mathcal{U}$) with solution value less than zero. □

Thus, it suffices to consider the following mixed integer program $\text{Sep}^{\mathbb{B}}(\mathcal{U})$ with linear objective and binary variables μ and ν. Additionally, the constraints (3.11b) are aggregated.

$$
\begin{aligned}
\text{Sep}^{\mathbb{B}}(\mathcal{U}) \qquad \min_{\mu,\nu,\xi,\omega} \quad & \sum_{i\in I} q \cdot \bar{x}_i \cdot \mu_i - \sum_{j \in J} \omega_j && && (3.12a)\\
\text{s.t.} \quad & |N(i)| \cdot \mu_i \geq \sum_{j \in N(i)} \nu_j && \text{for } i \in I && (3.12b)\\
& \omega_j \leq \xi_j && \text{for } j \in J && (3.12c)\\
& \omega_j \leq \nu_j \cdot \max_{\xi' \in \mathcal{U}} \xi'_j && \text{for } j \in J && (3.12d)\\
& \mu_i, \nu_j \in \mathbb{B} && \text{for } i \in I, j \in J && (3.12e)\\
& \xi \in \mathcal{U}. && && (3.12f)
\end{aligned}
$$

The constraints (3.12c) and (3.12d) together with the objective function ensure that, in an optimal solution $(\mu^\star, \nu^\star, \xi^\star, \omega^\star)$, we have $\omega^\star_j = \xi^\star_j$ if $\nu^\star_j = 1$ and zero otherwise.

Theorem 3.47. *Let $(\mu^\star, \nu^\star, \xi^\star, \omega^\star)$ be an optimal solution to $\text{Sep}^{\mathbb{B}}(\mathcal{U})$ with solution value less than zero. Then, $\xi^\star$ is a violating extreme scenario and $S = \left\{ j \in J \colon \nu^\star_j = 1 \right\}$ is a violating subset.*

Proof. By optimality of $(\mu^\star, \nu^\star, \xi^\star, \omega^\star)$ we get that $N(S) = \{i \in I : \mu^\star_i = 1\}$ and $\omega^\star(J) = \xi^\star(S)$. For the same reason, we have $\xi^\star \in \arg\max_{\xi \in \mathcal{U}} \xi(S)$. As the solution value is less than zero, we directly obtain that S is a violating subset and $\xi^\star$ is a violating extreme scenario. □

Note that, similar to the proof of Theorem 3.47, any solution (μ, ν, ξ, ω) of $\mathrm{Sep}^{\mathbb{B}}(\mathcal{U})$ with solution value less than zero leads to a violating subset and a violating, but not necessarily extreme, scenario. If we are only interested in violating subsets, it suffices to consider

$$
\begin{aligned}
\mathrm{Sep}^{\mathbb{B}}_{\mathrm{s}}(\mathcal{U}) \quad & \min_{\mu,\,\nu} \quad \sum_{i \in I} q \cdot \bar{x}_i \cdot \mu_i - \max_{\xi \in \mathcal{U}} \left\{ \sum_{j \in J} \xi_j \cdot \nu_j \right\} && (3.13a)\\
& \text{s.t.} \quad |N(i)| \cdot \mu_i \geq \sum_{j \in N(i)} \nu_j \quad \text{for } i \in I && (3.13b)\\
& \qquad\quad \mu_i, \nu_j \in \mathbb{B} \qquad\qquad \text{for } i \in I, j \in J. && (3.13c)
\end{aligned}
$$

Analogously to the proofs of Theorem 3.45 and Theorem 3.47, we obtain:

Corollary 3.48. *Let an instance of Separation be given. The optimal value of $Sep^{\mathbb{B}}_{s}(\mathcal{U})$ is zero if and only if $\bar{x}$ is feasible for the corresponding instance of Robust Min q-MSMC. If the optimal value of $Sep^{\mathbb{B}}_{s}(\mathcal{U})$ is less than zero and $(\mu^\star, \nu^\star)$ is an optimal solution, then $S = \left\{ j \in J \colon \nu^\star_j = 1 \right\}$ is a violating subset.* □

Certainly, the crux of this smaller formulation is the maximum term in the objective function. In the following section, we consider some specific classes of uncertainty sets. For some of these classes, this maximum can be replaced by a closed formula which turns the problem into a binary integer program.

3.4. Specific Classes of Uncertainty Sets

We now consider Robust Min q-MSMC for various specific classes of uncertainty sets and analyze the complexity of the emerging problems. We focus on uncertainty sets that are often applied in robust optimization as, e.g., interval uncertainty or ellipsoidal uncertainty. Throughout this section, we fix an instance of Robust Min q-MSMC and we successively assume that its uncertainty set $\mathcal{U}$ belongs to some particular class of uncertainty sets. We refer the reader to Section 2.6 for an introduction to the classes of uncertainty sets appearing in this section.

	$\lvert\mathcal{U}\rvert = 1$	$\lvert\mathcal{U}\rvert = 2$	$\lvert\mathcal{U}\rvert = k \geq 3$
$q = 1$	linear time	polynomial time	APX-complete
$q = 2$	polynomial time	APX-complete	APX-complete
$q \geq 3$	APX-complete	APX-complete	APX-complete

Table 3.1.: Complexity analysis of Robust Min q-MSMC for a fixed number of scenarios $k \in \mathbb{N}_{>0}$.

3.4.1. Discrete Uncertainty

In this section, we assume that $\mathcal{U}$ is given as an explicit list of k scenarios for some fixed $k \in \mathbb{N}_{>0}$, i.e.,

$$\mathcal{U} = \left\{\xi^1, \dots, \xi^k\right\}.$$

By Theorem 3.36 and Corollary 3.37 we already know that Feasibility and Separation are polynomial time solvable and Robust q-MSMC is contained in NP. Furthermore, for discrete uncertainty, the allocation formulation $\mathrm{P}(\mathcal{U})$ is compact and, given a vector $\bar{x} \in \mathbb{N}^{|I|}$, violating scenarios can be found in polynomial time. Additionally, in the constraint generation, the number of iterations is bounded by k. Concerning the value of k we concentrate on $k \geq 2$ as the results for $k = 1$ can be found in Section 3.1. Our findings are summarized in Table 3.1. We begin with analyzing the case where $q = 1$ and $|\mathcal{U}| = 2$.

Theorem 3.49. *Robust Min 1-MSMC restricted to instances with $|\mathcal{U}| = 2$ can be solved in time $\mathcal{O}(|I||J|^2 + |J|^3)$.*

Proof. Given an instance of Robust Min 1-MSMC with $\mathcal{U} = \left\{\xi^1, \xi^2\right\}$, we construct a network $H = (V, R, c)$ similar to the one of Definition 3.7. To ease notation let $J = \{1, \dots, n\}$ and define $J' := \{n+1, \dots, 2n\}$. We set $V := J \cup J' \cup \{s, t\}$ with $s, t \notin J$ and $R := \{(s, j) \colon j \in J\} \cup \{(j, t) \colon j \in J'\} \cup R_J$, where R_J contains the arc (j, j') for $j \in J$, $j' \in J'$ if there is a location $i \in N_G(j) \cap N_G(j' - n)$. The capacities of the arcs are defined as follows

$$c(r) := \begin{cases} \xi^1_j, & \text{if } r = (s, j), \\ \infty, & \text{if } r \in R_J, \\ \xi^2_j, & \text{if } r = (n + j, t), \end{cases}$$

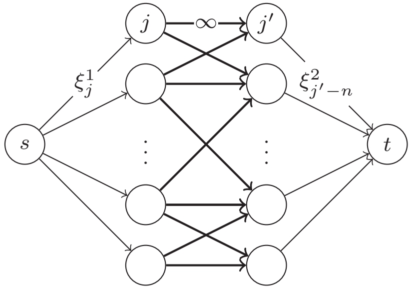


Figure 3.3.: Network H constructed in the proof of Theorem 3.49 where the capacity of an arc is given as an arc label. All thick arcs have infinite capacity.

for $r \in R$. An illustration of the constructed network is shown in Figure 3.3.

Next, we compute an integral maximum flow f in H and use it to define a solution x for our instance of Robust 1-MSMC. We start with initializing $x_i := 0$ for all $i \in I$. For every arc $(j, j') \in R_J$, we select a location $i \in N_G(j) \cap N_G(j' - n)$ and increase x_i by $f(j, j')$. This relates to putting $f(j, j')$ many suppliers into location i who cover $f(j, j')$ many clients in region j in scenario ξ^1 and $f(j, j')$ many clients in region $j' - n$ in scenario ξ^2. Additionally, for every region $j \in J$, we choose a location $i \in N(j)$ and increase x_i by

$$\max\left\{\xi^1_j - f(s, j), \xi^2_j - f(n + j, t)\right\}.$$

Due to the flow conservation constraints and the additional increase in the last step, we directly get that the constructed vector x is feasible. We now show that x is also optimal. First, note that, for $l \in \{1, 2\}$, it holds true that

$$\sum_{j \in J} \xi^l_j = \mathrm{val}(f) + \sum_{j \in J} \left(\xi^l_j - f(s, j)\right) = \mathrm{val}(f) + \sum_{j \in J} \left(\xi^l_j - f(n + j, t)\right). \quad (3.14)$$

Now, consider some solution x' for the instance of Robust 1-MSMC. There exist integral vectors $y(\xi^1)$ and $y(\xi^2)$ such that, for $l \in \{1, 2\}$, we have

$$\sum_{i \in N_G(j)} y(\xi^l)_{ij} \geq \xi^l_j \text{ for } j \in J \text{ and } \sum_{j \in N_G(i)} y(\xi^l)_{ij} \leq x'_i \text{ for } i \in I, \quad (3.15)$$

where we can assume equality in the first set of constraints. Now, we construct a flow f' in the network H beginning with $f' := 0$. Consider a location $i \in I$. Set $T_i := \min_{l\in\{1,2\}} \sum_{j\in N_G(i)} y(\xi^l)_{ij}$. For each $r \in \{1,\ldots,T_i\}$, we can choose two regions $j_1, j_2 \in N_G(i)$ with $f'(s,j_1) < c(s,j_1)$ and $f'(n+j_2,t) < c(n+j_2,t)$ by our assumption on (3.15). We increase f' by 1 along the path $(s, j_1, n+j_2, t)$. This procedure ends in a feasible flow with $\mathrm{val}(f') = T(I)$. With no loss of generality, we can assume that $x'_i = \max_{l\in\{1,2\}} \sum_{j\in N_G(i)} y(\xi^l)_{ij}$ and we get

$$\begin{aligned}
\sum_{i\in I} x'_i &= \sum_{i\in I} x'_i + \mathrm{val}(f') - \mathrm{val}(f') \\
&= \sum_{i\in I} \sum_{j\in N_G(i)} \left(y(\xi^1)_{ij} + y(\xi^2)_{ij}\right) - \mathrm{val}(f') \\
&= \sum_{j\in J} \left(\xi^1_j + \xi^2_j\right) - \mathrm{val}(f').
\end{aligned}$$

On the other hand, we have

$$\begin{aligned}
\sum_{i\in I} x_i &= \mathrm{val}(f) + \sum_{j\in J} \max\left\{\xi^1_j - f(s,j), \xi^2_j - f(n+j,t)\right\} \\
&= \mathrm{val}(f) + \sum_{j\in J} \left(\xi^1_j - f(s,j)\right) + \sum_{j\in J} \left(\xi^2_j - f(n+j,t)\right) \\
&\overset{(3.14)}{=} \sum_{j\in J} \left(\xi^1_j + \xi^2_j\right) - \mathrm{val}(f),
\end{aligned}$$

where the second equality follows as f is maximum. As $\mathrm{val}(f) \geq \mathrm{val}(f')$, we obtain optimality of x.

For the running time, note that constructing H needs time $\mathcal{O}(|I||J|^2)$. Further, computing a maximum flow in H can be done in time $\mathcal{O}(|J|^3)$, cf. [AMO93; Sch02]. Distributing the flow values among the x_i for $i \in I$ takes time $\mathcal{O}(|I| + |J|^2)$ (if we store the location corresponding to each edge during the construction of H) and increasing the x_i in the final step needs time $\mathcal{O}(|J|)$. □

Table 3.1 already reveals that all remaining cases lead to APX-complete problems. What is not revealed is that all these cases break down to showing APX-completeness of the following problem, which is closely related to the APX-complete Max 3-Dimensional Matching problem [Aus+02; Kan91].

Problem 3.50 (Min 3-Dimensional Cover).
Instance: Disjoint sets W, X, Y and a subset $\mathcal{C} \subseteq W \times X \times Y$.
Solution: A cover $\mathcal{C}'$ for $W \cup X \cup Y$, i.e., a subset $\mathcal{C}' \subseteq \mathcal{C}$ such that every element $z \in W \cup X \cup Y$ appears in at least one element (w, x, y) of $\mathcal{C}'$.
Measure: The cardinality of $\mathcal{C}'$.

Problem 3.51 (Max 3-Dimensional Matching).
Instance: Disjoint sets W, X, Y and a subset $M \subseteq W \times X \times Y$.
Solution: A *matching* M' for $W \cup X \cup Y$, i.e., a subset $M' \subseteq M$ such that no two elements of M' agree in any coordinate.
Measure: The cardinality of M'.

Observe that, in an instance of Min 3-Dimensional Cover, we can assume that every element in $W \cup X \cup Y$ appears in at least one element of $\mathcal{C}$. Moreover, if the sets W, X, and Y have equal cardinality $r \in \mathbb{N}_{>0}$, asking for a 3-dimensional matching of size r, i.e. a *perfect* matching, corresponds to asking for a cover of size r. Thus, NP-hardness of the decision version of Problem 3.50 follows directly from NP-hardness of 3-Dimensional Matching [GJ79]. Moreover, in the theory of hypergraphs, Problem 3.50 is also known as the 3-partite Edge Cover problem [BZ08; Leh82]. To the best of our knowledge, the APX-completeness result is not present in the literature yet.

Before proving it, we briefly recall a property of the analogous Min 2-Dimensional Cover problem. In this case, a minimum 2-dimensional cover corresponds to a minimum edge cover in a bipartite graph $G = (W \cup X, \mathcal{C})$ which can be computed in polynomial time by first solving the corresponding (2-dimensional) Max Matching problem on G and then augmenting the matching to an edge cover greedily. Vice versa, a maximum matching in G can be obtained from a minimum edge cover by choosing one edge from each component of the subgraph induced by the edges of the cover [GJ79; Law76]. However, this relation is lost when going from two to three dimensions as the following example shows.

Example 3.52. In the following, we use a graph which is given as the union of $r \in \mathbb{N}_{>0}$ triangles to define an instance of Min 3-Dimensional

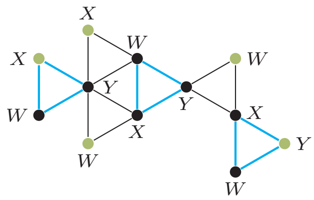


Figure 3.4.: Graph corresponding to Example 3.52.

Cover. The vertex set of the graph can be partitioned into three sets W, X, and Y such that *every* triangle in the graph has exactly one vertex from each of these sets. Thereby, each triangle represents one element of the set $\mathcal{C} \subseteq W \times X \times Y$. Then, a cover $\mathcal{C}' \subseteq \mathcal{C}$ for $W \cup X \cup Y$ corresponds to a selection of triangles in the constructed graph such that every vertex is covered. Observe that the graph may contain more than r triangles.

Consider the graph shown in Figure 3.4 with the set assignments labeled next to the vertices. This graph has a (unique) maximum matching of size 3 as shown by the blue triangles. Yet, taking this matching and augmenting it to a cover leads to a cover of size 6. On the other hand, selecting the triangles that contain the green vertices leads to a cover of size only 5 and the maximum matching contained in that cover has size 2.

The reason for this construction to work is that the green vertices are not contained in any other triangle. Thus, the triangles having a green vertex have to be selected for the cover anyway and, hence, the blue triangle in the middle is already covered by this selection. $\lhd$

For the complexity analysis of Min 3-Dimensional Cover, we consider the following well-known optimization problem that originates from the famous 3-SAT problem, cf. [GJ79; Aus+02]:

Problem 3.53 (Max 3-SAT)**.**
Instance: A set U of variables and a collection C of disjunctive clauses of at most three literals, where a literal is a variable or a negated variable of U.
Solution: A truth assignment $f \colon U \to \{\text{TRUE}, \text{FALSE}\}$.
Measure: The number of clauses satisfied by the truth assignment f.

For later reference, we note that, given an instance $\mathcal{I}$ of Max 3-SAT with

$m \in \mathbb{N}_{>0}$ clauses, there always exists a truth assignment fulfilling at least half of these clauses, i.e.,

$$2 \cdot \mathrm{OPT}(\mathcal{I}) \geq m. \tag{3.16}$$

To see this consider the assignment f that sets every variable to TRUE and the assignment $\bar{f}$ that sets every variable to FALSE. Then, we must have $\mathrm{SOL}(\mathcal{I}, f) + \mathrm{SOL}(\mathcal{I}, \bar{f}) \geq m$ as the clauses which are not fulfilled by f are fulfilled by $\bar{f}$. Hence,

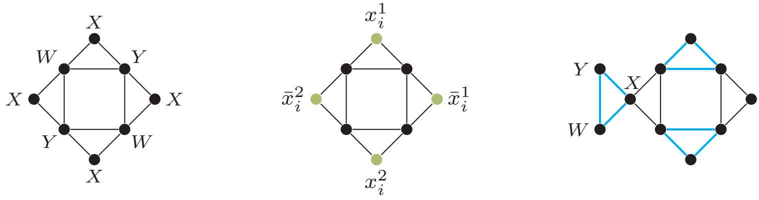
$$\mathrm{SOL}(\mathcal{I}, f) \geq \frac{m}{2} \quad \text{or} \quad \mathrm{SOL}(\mathcal{I}, \bar{f}) \geq \frac{m}{2}.$$

To show APX-hardness of Min 3-Dimensional Cover we provide an L-reduction from Max 3-SAT(3), where each variable appears at most three times and each clause has at most three literals. By [Aus+02] even this restricted version of Max 3-SAT remains APX-complete. Furthermore, as the instances of Max 3-SAT(3) need to fulfill further requirements, we can make some additional assumptions that will be helpful throughout this thesis.

Assumption 3.54. Let an instance of Max 3-SAT(3) with $m \in \mathbb{N}_{>0}$ clauses $C_1, \dots, C_m$ and $n \in \mathbb{N}_{>0}$ variables $X_1, \dots, X_n$, where each variable appears at most three times and each clause has at most three literals, be given. We may assume that each variable X_i appears at least one time negated and at least one time non-negated in the given clauses. Otherwise, we can assume that, in any optimal solution to the given instance, the only appearing literal L of variable X_i is evaluated to TRUE as this assignment can only improve the number of satisfied clauses. Thus, we can disregard the clauses containing L and any solution to the smaller instance leads to a solution to the original instance with linear difference in the solution values. Hence, we can apply this assumption when L-reducing from Max 3-SAT(3). As the total number of literal occurrences over all clauses is bounded by $3m$, this gives

$$2n \leq 3m. \tag{3.17}$$

Furthermore, each literal $L \in \big\{X_i, \overline{X}_i : i \in \{1, \dots, n\}\big\}$ appears at most twice as otherwise there is a variable appearing more than three times. For a literal L, we denote by $\overline{L}$ its corresponding negated literal.



(a) (b) (c)

Figure 3.5.: Constructing of a variable gadget G_{X_i} corresponding to variable X_i. The colored vertices are the literal vertices. Exemplarily, the literal $\overline{X}_i$ only appears once in the set of clauses. On the right, the triangles of a cover (assuming $\bar{x}_i^1$ is already covered) are colored.

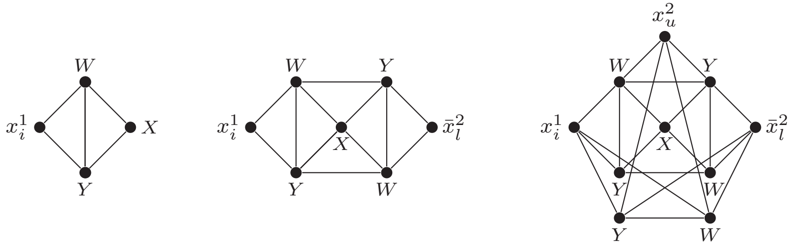
Theorem 3.55. *Min 3-Dimensional Cover is* APX-*complete.*

Proof. Min 3-Dimensional Cover is contained in APX as the Greedy Set Cover approximation algorithm gives a H(3)-approximation, cf. [Joh74; Lov75]. We now present an L-reduction from Max 3-SAT(3) to also obtain APX-hardness. Given an instance $\mathcal{I}$ of this problem with $m \in \mathbb{N}_{>0}$ clauses $C_1, \ldots, C_m$ and $n \in \mathbb{N}_{>0}$ variables $X_1, \ldots, X_n$, we make use of Assumption 3.54. The instance $\mathcal{I}'$ of Min 3-Dimensional Cover is constructed using a graph of triangles as in Example 3.52.

For each variable X_i, we build a "star" containing four triangles as shown in Figure 3.5a and call this subgraph G_{X_i}. Observe the set affiliation of every vertex here. For later reference, we refer to the tips of the star, i.e., the elements contained in X, as x_i^1, $\bar{x}_i^1$, x_i^2, and $\bar{x}_i^2$ as shown in Figure 3.5b. We call these vertices *literal vertices* and the complete gadget is named *variable gadget*. These gadgets are inspired by [Kan91].

Consider the clauses $C_1, \ldots, C_m$ in increasing order of their indices. As each literal appears at most two times, we can label the literals by their first and potentially second occurrence, e.g., $\overline{X}_i^1$ and if existent $\overline{X}_i^2$. If, for some $i \in \{1, \ldots, n\}$, a literal $L \in \{X_i, \overline{X}_i\}$ appears only once, we append an additional triangle to the literal vertex of variable gadget G_{X_i} that corresponds to the missing second occurrence, see Figure 3.5c.

For each clause C_j, we now construct a *clause gadget* G_{C_j} depending on



(a) $C_j = X_i^1$. (b) $C_j = X_i^1 \vee \overline{X}_l^2$. (c) $C_j = X_i^1 \vee \overline{X}_l^2 \vee X_u^2$.

Figure 3.6.: Construction of the clause gadgets G_{C_j} depending on the number of literals appearing in clause C_j. The literals are labeled according to their first and potentially second occurrence in the given formula.

the number of literals present in the clause. If a clause C_j contains only one literal L, we construct the gadget shown in Figure 3.6a. If L is the l^{th} occurrence of literal X_i ($\overline{X}_i$) with $l \in \{1, 2\}$, the leftmost vertex of the clause gadget corresponds to the vertex $x_i^l \in X$ ($\bar{x}_i^l \in X$). Moreover, observe the assignments of vertices to the sets W, X, and Y for the remaining vertices in Figure 3.6a. If a clause C_j contains two literals, i.e., $C_j = L_1 \vee L_2$, the gadget of Figure 3.6b with the shown set correspondences is constructed. Again, if L_1 is the l^{th} occurrence of literal X_i ($\overline{X}_i$) with $l \in \{1, 2\}$, the leftmost vertex of the clause gadget corresponds to the vertex $x_i^l \in X$ ($\bar{x}_i^l \in X$). Analogously, we proceed with L_2 and the rightmost vertex of the clause gadget. Similarly, we construct the clause gadget if $C_j = L_1 \vee L_2 \vee L_3$ (cf. Figure 3.6c). Observe that these constructions are compatible with the set assignments of the variable gadgets. Examples of literal vertices appearing in the clause gadgets are shown in Figure 3.6.

Altogether, the graph

$$G := \left(\bigcup_{i=1}^{n} G_{X_i} \right) \cup \left(\bigcup_{j=1}^{m} G_{C_j} \right),$$

which is a union of triangles, now induces our instance of Min 3-Dimensional Cover, say $\mathcal{I}'$, as in Example 3.52. Denote by $\mathcal{C}$ the set of *all* triangles of G.

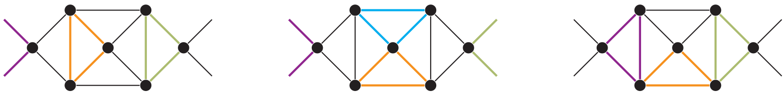


Figure 3.7.: Selection of triangles for a 2-literal clause gadget depending on whether the literal vertices are already covered (indicated by the half-done colored edges).

Observe that the gadgets only share vertices contained in X, so that

$$|\mathcal{C}| = \sum_{j=1}^{m} (4|C_j| - 2) + 4n + 4n - \sum_{j=1}^{m} |C_j| = 8n - 2m + 3 \sum_{j=1}^{m} |C_j|.$$

We now derive an upper bound on the total number of triangles needed to cover every vertex of G, i.e., the total number of elements of $\mathcal{C}$ needed to cover $W \cup X \cup Y$. Given an optimal truth assignment for $\mathcal{I}$, we obtain a cover $\mathcal{C}'$ for $\mathcal{I}'$ by first choosing the $4n - \sum_{j=1}^{m} |C_j|$ additionally appended triangles of the variable gadgets and, for each variable gadget, the two oppositely located triangles that contain the literal vertices corresponding to the true literal.

Now, consider a clause gadget G_{C_j}. If $|C_j| = 1$ and if the literal vertex of this gadget is already covered by its corresponding variable gadget, we choose the triangle of uncovered vertices for $\mathcal{C}'$. Otherwise, we choose both triangles of the clause gadget for $\mathcal{C}'$. Observe that, in both cases, selecting less triangles does not lead to a feasible cover.

Now, suppose $|C_j| = 2$. If one or both literal vertices are already covered by variable gadgets, we add two more triangles to $\mathcal{C}'$ to cover the remaining vertices. Less triangles are not possible as there are two elements of Y that need to be covered. If no literal vertex is already covered, we select three more triangles. Again, choosing less triangles is not possible as there are three elements of X that need to be covered. Illustrations of the various cases are shown in Figure 3.7. An already covered literal vertex is represented by two colored incident edges that have no other end vertex.

Finally, suppose $|C_j| = 3$. If one or more literal vertices are already covered by variable gadgets, we choose three more triangles for $\mathcal{C}'$ to cover all remaining vertices of the gadget. If no literal vertex is already covered by variable gadgets, four more triangles are added to $\mathcal{C}'$ to cover all vertices of the gadget. As before, choosing less triangles is not possible as there

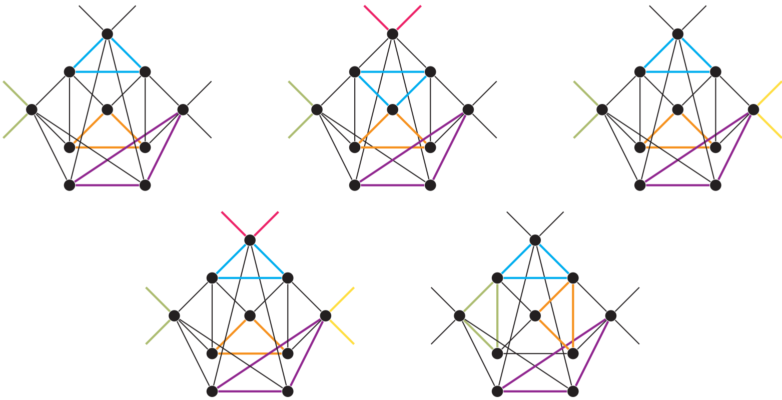


Figure 3.8.: Selection of triangles for a 3-literal clause gadget depending on whether the literal vertices are already covered (indicated by the half-done colored edges).

are either three elements of Y or four elements of X that still need to be covered. Illustrations of the most important cases are shown in Figure 3.8. All remaining cases work out analogously to the shown cases.

Hence, for each clause gadget corresponding to a clause with $r \in \mathbb{N}$ literals, we choose r many additional triangles if the clause is true and $r+1$ many additional triangles if the clause is false, cf. Figures 3.7 and 3.8. Thus, we get

$$\begin{aligned} \mathrm{OPT}(\mathcal{I}') &\leq 4n - \sum_{j=1}^{m} |C_j| + 2n + \sum_{j=1}^{m} (|C_j| + 1) - \mathrm{OPT}(\mathcal{I}) \\ &= 6n + m - \mathrm{OPT}(\mathcal{I}) \\ &\leq 10m - \mathrm{OPT}(\mathcal{I}) \\ &\leq 19 \cdot \mathrm{OPT}(\mathcal{I}), \end{aligned} \tag{3.18}$$

where the inequalities are obtained using the estimates (3.16) and (3.17).

On the other hand, given a cover $\mathcal{C}' \subseteq \mathcal{C}$, we know that each additionally appended triangle of the variable gadgets is contained in $\mathcal{C}'$. Furthermore, we can assume that exactly two more triangles are chosen in each variable gadget and that they are located oppositely: If two more triangles are chosen and not located oppositely, the solution $\mathcal{C}'$ is not feasible. If three more triangles are chosen, there is one triangle T chosen whose opposite triangle is not chosen. Denote by x the literal vertex of triangle T. We do not deteriorate our given cover $\mathcal{C}'$ if we swap the chosen triangle to some other triangle that also contains x, which is either an additionally appended triangle or it is contained in a clause gadget. If four more triangles are chosen, we do not worsen our solution if we analogously swap the triangles containing the negated literal vertices. With these modifications we assign the variable X_i the value TRUE if and only if the two oppositely located triangles of the variable gadget G_{X_i} containing the literal vertices x_i^1 and x_i^2 are in the cover $\mathcal{C}'$. This gives our truth assignment f. An example is given in Figure 3.5c where $f(X_i) = \text{TRUE}$.

By construction of $\mathcal{I}'$ a clause C_j is fulfilled if its corresponding clause gadget G_{C_j} contains a literal vertex which is covered by its corresponding variable gadget. Let $t \in \mathbb{N}$ be the number of clause gadgets where no literal vertex is covered by its corresponding variable gadget. Thus, $\mathrm{SOL}(\mathcal{I}, f) = m - t$. Further, by our above argumentation we can assume:

(a) For a clause gadget G_{C_j} having a literal vertex that is already covered by its variable gadget, the cover $\mathcal{C}'$ contains at least $|C_j|$ many triangles.

(b) For a clause gadget G_{C_j} having no literal vertex that is already covered by its variable gadget, the cover $\mathcal{C}'$ contains at least $|C_j| + 1$ many triangles.

This gives

$$|\mathcal{C}'| \geq 4n - \sum_{j=1}^{m} |C_j| + 2n + \sum_{j=1}^{m} |C_j| + t = 6n + t. \tag{3.19}$$

In total we obtain

$$\begin{aligned} \mathrm{OPT}(\mathcal{I}) - \mathrm{SOL}(\mathcal{I}, f) &= \mathrm{OPT}(\mathcal{I}) - m + t \\ &\overset{(3.18)}{\leq} 6n + m - \mathrm{OPT}(\mathcal{I}') - m + t \\ &\overset{(3.19)}{\leq} |\mathcal{C}'| - \mathrm{OPT}(\mathcal{I}'). \end{aligned}$$

Hence, Max 3-SAT(3) L-reduces to Min 3-Dimensional Cover. $\square$

Thus, unless P = NP, there is no PTAS for Min 3-Dimensional Cover. Note that this result also implies APX-completeness of Min Set Cover(3) even if every set contains *exactly* three elements. Moreover, aside from the argumentation on Page 68, for the decision version 3-Dimensional Cover the above proof reveals:

Corollary 3.56. *3-Dimensional Cover is strongly* NP*-complete.*

Proof. Given a subset $\mathcal{C}' \subseteq \mathcal{C}$, we can check whether each element is covered in polynomial time. Thus, the problem is contained in NP. Moreover, the proof of Theorem 3.55 shows that a solution to $\mathcal{I}$ with at least $t \in \mathbb{N}$ satisfied clauses leads to a solution to $\mathcal{I}'$ with value at most $6n+m-t$ in polynomial time and vice versa. □

The next theorem shows why APX-completeness of Min 3-Dimensional Cover directly leads to APX-hardness of Robust Min q-MSMC restricted to instances with $|\mathcal{U}|$ fixed and $q \cdot |\mathcal{U}| \geq 3$.

Theorem 3.57. *For any fixed $q, k \in \mathbb{N}_{>0}$ with $qk \geq 3$, Robust Min q-MSMC with discrete uncertainty and $|\mathcal{U}| = k$ is* APX*-hard.*

Proof. We show that, for any fixed value of $qk \geq 3$, Min 3-Dimensional Cover L-reduces to Robust Min q-MSMC with exactly k scenarios. Thus, let an instance $\mathcal{I}$ of Min 3-Dimensional Cover be given, i.e., a set $\mathcal{C} \subseteq W \times X \times Y$ for disjoint sets W, X, and Y. Similar to the proof of Theorem 3.14, we construct a bipartite graph $G = (I \cup J, E)$ with $I := \mathcal{C}$, $J := W \cup X \cup Y$ and $N((w,x,y)) := \{w, x, y\}$ for all $(w,x,y) \in \mathcal{C}$. The definition of the uncertainty set $\mathcal{U}$ depends on the value of k.

For $k = 1$, we set $\mathcal{U} := \{1\}$, cf. Theorem 3.14. For $k = 2$, we set $\mathcal{U} := \left\{\xi^1, \xi^2\right\}$ with

$$\xi_j^1 := \begin{cases} 1, & \text{if } j \in W \cup X, \\ 0, & \text{if } j \in Y, \end{cases} \quad \text{and} \quad \xi_j^2 := \begin{cases} 0, & \text{if } j \in W \cup X, \\ 1, & \text{if } j \in Y, \end{cases}$$

for $j \in J$. For $k = 3$, we set $\mathcal{U} := \left\{\xi^1, \xi^2, \xi^3\right\}$ with

$$\xi_j^1 := \begin{cases} 1, & \text{if } j \in W, \\ 0, & \text{otherwise,} \end{cases} \quad \xi_j^2 := \begin{cases} 1, & \text{if } j \in X, \\ 0, & \text{otherwise,} \end{cases} \quad \xi_j^3 := \begin{cases} 1, & \text{if } j \in Y, \\ 0, & \text{otherwise,} \end{cases}$$

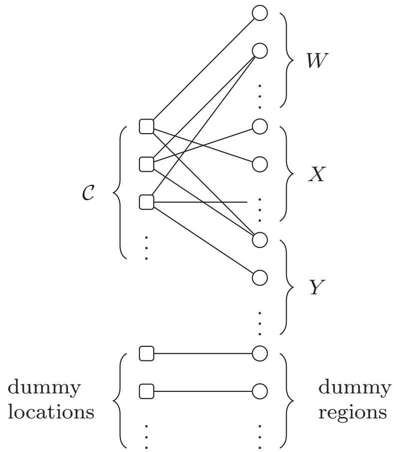


Figure 3.9.: Bipartite graph G constructed in the proof of Theorem 3.57.

for $j \in J$. For $k > 3$, we add $k-3$ many dummy locations and equally many dummy regions to the bipartite graph where one dummy location is connected to one dummy region. The uncertainty set $\mathcal{U}$ contains the scenarios defined for the case of three scenarios plus one additional scenario ξ for each dummy region j with $\xi_j = 1$ and zero otherwise. Note that, in any solution x to the constructed instance $\mathcal{I}'$ of Robust Min q-MSMC, we can assume that $x_i = 1$ if i is a dummy location. An illustration of the constructed graph is given in Figure 3.9.

Let $\mathcal{C}' \subseteq \mathcal{C}$ be an optimal cover for $W \cup X \cup Y$. We set $x_i := 1$ if $i \in \mathcal{C}'$ or i is a dummy location and zero otherwise. By construction this gives a solution to the instance $\mathcal{I}'$ with

$$\begin{aligned} \mathrm{OPT}(\mathcal{I}') &\leq \mathrm{OPT}(\mathcal{I}) + \max\{0, k-3\} \\ &\leq \max\{1, k-2\} \cdot \mathrm{OPT}(\mathcal{I}). \end{aligned} \tag{3.20}$$

Further, the regions in $N(\mathcal{C})$ can be partitioned into three disjoint sets such that each location in $\mathcal{C}$ is adjacent to exactly one element from each set. Additionally, we have, for every location i,

(a) $\max_{\xi \in \mathcal{U}} \xi(N(i)) \leq 3$ if $k = 1$,

(b) $\max_{\xi \in \mathcal{U}} \xi(N(i)) \leq 2$ if $k = 2$, and

(c) $\max_{\xi \in \mathcal{U}} \xi(N(i)) \leq 1$ if $k \geq 3$

by construction of the scenarios. Observe that, if $k = 1$ and $k = 2$, we must have $q \geq 3$ and $q \geq 2$, respectively. Therefore, in any solution x to $\mathcal{I}'$, we can assume that $x_i \leq 1$ for every location $i \in \mathcal{C}$ and $x_i = 1$ for every dummy location i. Given such a solution x we choose $\mathcal{C}' := \{i \in \mathcal{C} \colon x_i = 1\}$. By construction of the scenarios $\mathcal{C}'$ is feasible for $\mathcal{I}$ and we get with (3.20)

$$\begin{aligned}
\mathrm{SOL}(\mathcal{I}, \mathcal{C}') - \mathrm{OPT}(\mathcal{I}) \\
&= x(\mathcal{C}) - \mathrm{OPT}(\mathcal{I}) \\
&\leq (\mathrm{SOL}(\mathcal{I}', x) - \max\{0, k-3\}) - (\mathrm{OPT}(\mathcal{I}') - \max\{0, k-3\}) \\
&= \mathrm{SOL}(\mathcal{I}', x) - \mathrm{OPT}(\mathcal{I}').
\end{aligned}$$

Moreover, if x is optimal for $\mathcal{I}'$, we also get $\mathrm{OPT}(\mathcal{I}) \leq \mathrm{OPT}(\mathcal{I}') - \max\{0, k-3\}$. □

Corollary 3.58. *For any fixed $q, k \in \mathbb{N}_{>0}$ with $qk \geq 3$, Robust q-MSMC with discrete uncertainty and $|\mathcal{U}| = k$ is strongly* NP*-complete.*

Proof. As $|\mathcal{U}|$ is fixed, it is easy to see that the problem is contained in NP. Furthermore, by the proof of Theorem 3.57, we get that a solution to $\mathcal{I}$ with $l \in \mathbb{N}$ elements leads to a solution to $\mathcal{I}'$ with value $l + \max\{0, |\mathcal{U}| - 3\}$ in polynomial time and vice versa. □

In Chapter 4, we provide an $\mathrm{H}(q|\mathcal{U}|)$-approximation algorithm for Robust Min q-MSMC with discrete uncertainty and, hence, we obtain the desired APX-completeness result.

Corollary 3.59. *For any fixed $q, k \in \mathbb{N}_{>0}$ with $qk \geq 3$, Robust Min q-MSMC with discrete uncertainty and $|\mathcal{U}| = k$ is* APX*-complete.* □

Having a discrete set of scenarios is a quite strong restriction. We move on to polyhedral uncertainty sets in the following.

3.4.2. Interval Uncertainty

When applying interval uncertainty, we assume that the demand ξ_j in any region $j \in J$ varies in a given interval $[a_j, b_j]$ with $a_j \leq b_j$, $b_j \geq 1$, and

$a_j, b_j \in \mathbb{N}$, i.e., we consider the integral points of a hypercube and

$$\mathcal{U} = \left\{\xi \in \mathbb{N}^{|J|} \colon a \leq \xi \leq b\right\}.$$

Utilizing interval uncertainty for Robust Min q-MSMC, we can easily solve the maximization problem in $\mathrm{P_s}(\mathcal{U})$ as $\max_{\xi \in \mathcal{U}} \xi(S) = b(S)$. Therefore, it suffices to only consider the worst-case scenario b and we obtain $\mathrm{P_s}(\mathcal{U}) = \mathrm{P_s}(b)$. Thus, by Proposition 3.10 and Theorem 3.13 we get that Robust Min q-MSMC with interval uncertainty is polynomial time solvable for $q = 1$ and $q = 2$. For $q \geq 3$, Robust q-MSMC with interval uncertainty is strongly NP-complete by Corollary 3.15 and its optimization version is APX-complete by Corollary 3.17. To complete this list of results, we note that Feasibility, Separation, and Max Robust Sum are polynomial time solvable for interval uncertainty.

To overcome the drawbacks of conservatism of interval uncertainty, in the next section, we tighten the uncertainty set by additionally introducing an upper bound on the total number of clients.

3.4.3. Budgeted Uncertainty

As with interval uncertainty here we assume that the demand ξ_j in region $j \in J$ is lower bounded by $a_j \in \mathbb{N}$ and upper bounded by $b_j \in \mathbb{N}$. Concerning the total number of clients in all regions, we additionally require this value to not exceed some given bound $\Gamma \in \mathbb{N}$ to prevent the global worst case, which caused the conservatism in interval uncertainty. Thus, we consider uncertainty sets of the form

$$\mathcal{U} = \left\{\xi \in \mathbb{N}^{|J|} \colon a \leq \xi \leq b, \xi(J) \leq \Gamma\right\} \tag{3.21}$$

with $a, b \in \mathbb{N}^{|J|}$, $a \leq b$, and $\Gamma \in \mathbb{N}$.

Assumption 3.60. In order to obtain a meaningful budgeted uncertainty set, we make the following assumptions:

(a) It holds true that $a(J) \leq \Gamma \leq b(J)$ so that $\mathcal{U} \neq \varnothing$.

(b) For every region $j \in J$, we have $b_j \geq 1$ as otherwise region j is redundant.

(c) The value Γ is chosen in such a way that $b_j + a(J \setminus \{j\}) \leq \Gamma$ for every $j \in J$. Otherwise we can decrease the upper bound b_j in the corresponding region.

Observe that, in the special case of $\Gamma = b(J)$, we regain the previous interval uncertainty set. Moreover, the results of this section also apply when considering only scenarios of $\mathcal{U}$ whose entries sum up to exactly Γ. As it suffices to consider only non-dominated scenarios in an instance of Robust Min q-MSMC, we could therefore restrict our analysis to this smaller uncertainty set. Yet, to simplify proofs and notation we decide to stay with the form given in (3.21). First of all, we directly obtain by Remark 3.33:

Theorem 3.61. *For any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-MSMC with budgeted uncertainty is* APX*-hard and its decision version is strongly* NP*-hard.*

Proof. For $n \in \mathbb{N}_{>0}$ many regions, setting $a := 0$, $b := 1$, and $\Gamma := 1$ yields the uncertainty set $\mathcal{U} = \{0, e_1, \ldots, e_n\}$ and the claim follows by Remark 3.33. □

We continue with the consideration of Max Robust Sum restricted to budgeted uncertainty sets as this leads to the key changes compared to the previous section.

Lemma 3.62. *Given an instance of Max Robust Sum with budgeted uncertainty, we have that*

$$\max_{\xi \in \mathcal{U}} \sum_{j \in S} \xi_j = \min \{b(S), \Gamma - a(J \setminus S)\}$$

with $J := \{1, \ldots, n\}$. Thus, Max Robust Sum can be solved in time $\mathcal{O}(n)$.

Proof. Let $\mathcal{U} \subseteq \mathbb{N}^n$ be of the form (3.21) and $S \subseteq J$. By definition of $\mathcal{U}$ it follows that, for $\xi \in \mathcal{U}$, we have $\xi(S) \leq b(S)$ as well as $\xi(S) + a(J \setminus S) \leq \Gamma$. Hence,

$$\max_{\xi \in \mathcal{U}} \xi(S) \leq \min \{b(S), \Gamma - a(J \setminus S)\}.$$

If $b(S) + a(J \setminus S) \leq \Gamma$, then obviously $\max_{\xi \in \mathcal{U}} \xi(S) = b(S)$. On the other hand, if $b(S) + a(J \setminus S) > \Gamma$, we have $b(S) > \Gamma - a(J \setminus S)$ and also $a(S) \leq \Gamma - a(J \setminus S)$ by Assumption 3.60. Thus, there exists a scenario ξ with $\xi(S) = \Gamma - a(J \setminus S)$ and $\xi(J \setminus S) = a(J \setminus S)$ and we have $\max_{\xi \in \mathcal{U}} \xi(S) = \Gamma - a(J \setminus S)$. □

Lemma 3.62 reveals that, in the case of budgeted uncertainty, we can compute a violating extreme scenario given a violating subset S in time $\mathcal{O}(|J|)$ using an iterative procedure. Moreover, we can replace $\max_{\xi \in \mathcal{U}} \xi(S)$ in $\mathrm{P_s}(\mathcal{U})$ with

$$\min\{b(S), \Gamma - a(J \setminus S)\}. \tag{3.22}$$

But in comparison to the non-robust formulation $\mathrm{P_s}(d)$ on Page 41, the value (3.22) cannot be split into a sum over clients in the regions of S anymore.

In the following, we show that Separation with budgeted uncertainty is NP-complete. To prove this result, we additionally need the definition of the Knapsack problem, cf. [GJ79].

Problem 3.63 (Knapsack).
Instance: A finite set U, a size $s_u \in \mathbb{N}_{>0}$ and a profit $p_u \in \mathbb{N}_{>0}$ for each $u \in U$, and two integers $B, K \in \mathbb{N}_{>0}$.
Question: Is there a subset $U' \subseteq U$ such that $\sum_{u \in U'} s_u \leq B$ as well as $\sum_{u \in U'} p_u \geq K$?

Theorem 3.64. *For any fixed $q \in \mathbb{N}_{>0}$, Separation with budgeted uncertainty is* NP*-complete.*

Proof. As in Theorem 3.38 we get that Separation with budgeted uncertainty is contained in NP. We show that Knapsack reduces to Separation in polynomial time. To that end, let an instance of Knapsack be given with a set $U = \{1, \dots, n\}$, a size s_u and a profit p_u associated with each element $u \in U$, and two integers $B, K \in \mathbb{N}_{>0}$. We define a bipartite graph $G = (I \cup J, E)$ with $I := U \cup \{2n+1\}$, $J := \{n+1, \dots, 2n\}$ and

$$E := \{[u, n+u], [2n+1, n+u] \colon u \in U\}.$$

Moreover, we set $\bar{x}_u := s_u$ and $b_{n+u} := q \cdot (p_u + s_u)$ for all $u \in U$. Further, we let $\bar{x}_{2n+1} := K - 1$ and $\Gamma := q \cdot (B + K)$. Finally, we define $a_j := 0$ for all $j \in J$. For an illustration of the constructed bipartite graph see Figure 3.10.

Now, given a solution $U' \subseteq U \subseteq I$ of the Knapsack instance with $s(U') \leq B$ and $p(U') \geq K$, we choose $S := \{n + u \colon u \in U'\} \subseteq J$. Then, S is non-empty since $U' \neq \varnothing$ and we have

$$\frac{\Gamma}{q} - \bar{x}(U') = B + K - s(U') \geq B + K - B = K$$

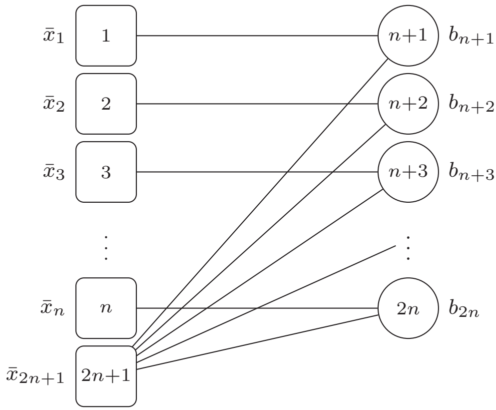


Figure 3.10.: Bipartite graph G constructed in the proof of Theorem 3.64.

as well as

$$\begin{aligned}\frac{b(S)}{q} - \bar{x}(U') &= \sum_{u \in U'} \frac{b_{n+u}}{q} - \sum_{u \in U'} \bar{x}_u = \sum_{u \in U'} (p_u + s_u) - \sum_{u \in U'} s_u \\ &= \sum_{u \in U'} p_u \geq K,\end{aligned}$$

yielding

$$\min\left\{\frac{b(S)}{q}, \frac{\Gamma}{q}\right\} - \bar{x}(U') \geq K.$$

Subtracting $K - 1 = \bar{x}_{2n+1}$ on both sides we obtain:

$$\min\left\{\frac{b(S)}{q}, \frac{\Gamma}{q}\right\} - \bar{x}(U') - \bar{x}_{2n+1} \geq 1 \;\Leftrightarrow\; \min\left\{\frac{b(S)}{q}, \frac{\Gamma}{q}\right\} - \bar{x}(N(S)) \geq 1,$$

i.e., $\min\{b(S), \Gamma\} - q \cdot \bar{x}(N(S)) > 0$. Thus, the set S is a solution for the constructed instance of Separation.

On the other hand, let $S \subseteq J$ be a solution for the instance of Separation with the property

$$\min\{b(S), \Gamma\} > q \cdot \bar{x}(N(S)). \tag{3.23}$$

Thus, the set S is non-empty and contains an element of the form $n + u$ for some $u \in U$. Set $U' := \{u \colon n + u \in S\} \subseteq U$. Our aim is to show that U'

is a solution to the Knapsack instance. We have $N(S) = U' \cup \{2n+1\}$. Reformulating the right-hand side of (3.23) we get

$$\bar{x}(N(S)) = \bar{x}(U') + \bar{x}_{2n+1} = \bar{x}(U') + K - 1.$$

Thus, in total we have $\min\{b(S), \Gamma\} - q \cdot \bar{x}(U') > q \cdot (K-1)$, i.e.,

$$\min\left\{\frac{b(S)}{q}, \frac{\Gamma}{q}\right\} - \bar{x}(U') \geq K.$$

When inserting the above definitions this expression becomes

$$\min\{p(U') + s(U'), B + K\} - s(U') \geq K. \tag{3.24}$$

Now, we need to differentiate between two cases:

If $p(U')+s(U') \leq B+K$, (3.24) yields $p(U') = p(U')+s(U')-s(U') \geq K$ and $s(U') \leq B + K - p(U') \leq B + K - K = B$.

If $p(U')+s(U') > B+K$, (3.24) yields $B+K-s(U') \geq K$, i.e., $s(U') \leq B$. Furthermore, $p(U') > B + K - s(U') \geq B + K - B = K$.

Thus, the set U' is a solution for our given Knapsack instance. □

Corollary 3.65. *For any fixed $q \in \mathbb{N}_{>0}$, Feasibility with budgeted uncertainty is* co-NP*-complete.* □

Due to Lemma 3.62 we can adapt formulation $\mathrm{Sep}_{\mathrm{s}}^{\mathbb{B}}(\mathcal{U})$ on Page 64 to budgeted uncertainty sets $\mathcal{U}_{\mathrm{B}}$. This yields the following MIP that we refer to as $\mathrm{Sep}_{\mathrm{s}}^{\mathbb{B}}(\mathcal{U}_{\mathrm{B}})$:

$$\mathrm{Sep}_{\mathrm{s}}^{\mathbb{B}}(\mathcal{U}_{\mathrm{B}}) \quad \min_{\mu, \nu, \pi} \quad \sum_{i \in I} q \cdot \bar{x}_i \cdot \mu_i - \pi \tag{3.25a}$$

$$\text{s.t.} \quad \pi \leq \sum_{j \in J} b_j \cdot \nu_j \tag{3.25b}$$

$$\pi \leq \Gamma - \sum_{j \in J} a_j + \sum_{j \in J} a_j \cdot \nu_j \tag{3.25c}$$

$$|N(i)| \cdot \mu_i \geq \sum_{j \in N(i)} \nu_j \quad \text{for } i \in I \tag{3.25d}$$

$$\mu_i, \nu_j \in \mathbb{B} \quad \text{for } i \in I, j \in J, \tag{3.25e}$$

where $\bar{x} \in \mathbb{N}^{|I|}$ is the given vector which we wish to test for feasibility. Note that, by Lemma 3.62 and the constraint (3.25b) as well as (3.25c), the variable π is upper bounded by the worst-case demand of the potentially violating subset $S := \{j \in J : \nu_j = 1\}$. In an optimal solution, this bound is attained. When looking for violating extreme scenarios, we can analogously adapt $\mathrm{Sep}^{\mathbb{B}}(\mathcal{U})$ on Page 63 using that $\max_{\xi \in \mathcal{U}_\mathrm{B}} \xi_j = b_j$ and including the definition of $\mathcal{U}_\mathrm{B}$.

3.4.4. Multi-budgeted Uncertainty

As a further step of generalizing interval uncertainty and also generalizing budgeted uncertainty, we consider multi-budgeted uncertainty sets in this section. We assume we have given a set $\mathcal{S} \subseteq 2^J$ of subsets of J and, for each $S \in \mathcal{S}$, two non-negative integers a_S, b_S with $a_S \leq b_S$. Then, the uncertainty set under consideration is

$$\mathcal{U} = \left\{ \xi \in \mathbb{N}^{|J|} : a_S \leq \xi(S) \leq b_S \text{ for } S \in \mathcal{S} \right\}.$$

In comparison to budgeted uncertainty sets, which are able to prevent the global worst case as the total sum of demands is limited, multi-budgeted uncertainty sets are additionally able to prevent local worst cases as, for every set $S \in \mathcal{S}$, the sum of the demands in regions of S is bounded. Observe that interval uncertainty is regained by letting $\mathcal{S} := \{\{j\} : j \in J\}$ with $a_{\{j\}} := a_j$ as well as $b_{\{j\}} := b_j$ for every region $j \in J$. Moreover, budgeted uncertainty is represented by adding the set J to $\mathcal{S}$ and setting $a_J := 0$ and $b_J := \Gamma$. Thus, by Theorem 3.61 we have:

Theorem 3.66. *For any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-MSMC with multi-budgeted uncertainty is* APX*-hard and its decision version is strongly* NP*-hard.* □

Additionally, we can apply more results of the previous section. By Theorem 3.64 and Corollary 3.65 we directly obtain that, for any fixed $q \in \mathbb{N}_{>0}$, Separation and Feasibility with multi-budgeted uncertainty are NP- and co-NP-complete, respectively. Further, recalling the NP-completeness proof of Robust Sum in Lemma 3.22, given a simple graph $G = (V, E)$ with $n \in \mathbb{N}_{>0}$ vertices, we construct the uncertainty set

$$\begin{aligned} \mathcal{U} :=& \{\xi \in \mathbb{B}^n : \xi_u + \xi_v \leq 1 \text{ for } [u, v] \in E\} \\ =& \{\xi \in \mathbb{N}^n : 0 \leq \xi_u + \xi_v \leq 1 \text{ for } \{u, v\} \in \mathcal{S}\} \end{aligned}$$

with $\mathcal{S} \coloneqq \{\{u,v\} : [u,v] \in E\}$. Hence, together with Remark 3.23, we also get the following result.

Lemma 3.67. *Max Robust Sum with multi-budgeted uncertainty is* APX-*hard and its decision version is strongly* NP-*complete even if* $S = \{1,\dots,n\}$ *and* $B \in \mathbb{N}_{>0}$ *is a multiple of* q *for some fixed* $q \in \mathbb{N}_{>0}$. □

By the previous lemma we need to adjust $\mathrm{Sep}^{\mathbb{B}}(\mathcal{U})$ for the usage with multi-budgeted uncertainty sets $\mathcal{U}_{\mathrm{M}}$ and we obtain $\mathrm{Sep}^{\mathbb{B}}(\mathcal{U}_{\mathrm{M}})$:

$$
\begin{array}{llrll}
\mathrm{Sep}^{\mathbb{B}}(\mathcal{U}_{\mathrm{M}}) & \min\limits_{\mu,\nu,\xi,\omega} & \multicolumn{2}{l}{\displaystyle\sum_{i\in I} q\cdot \bar{x}_i \cdot \mu_i - \sum_{j\in J}\omega_j} & \\
& \text{s.t.} & |N(i)|\cdot\mu_i \geq & \displaystyle\sum_{j\in N(i)} \nu_j & \text{for } i \in I \\
& & \omega_j \leq & \xi_j & \text{for } j \in J \\
& & \omega_j \leq & \nu_j \cdot \max\limits_{S\in\mathcal{S}\colon j\in S} b_S & \text{for } j \in J \\
& & a_S \leq & \displaystyle\sum_{j\in S}\xi_j \leq b_S & \text{for } S \in \mathcal{S} \\
& & & \mu_i,\nu_j \in \mathbb{B} & \text{for } i \in I, j \in J \\
& & & \xi_j \in \mathbb{N} & \text{for } j \in J.
\end{array}
$$

Observe that, in comparison to $\mathrm{Sep}^{\mathbb{B}}(\mathcal{U})$, we only included the definition of $\mathcal{U}_{\mathrm{M}}$ and the trivial upper bound $\max\{b_S \colon S \in \mathcal{S} \wedge j \in S\}$ for the j^{th} coordinate of $\xi \in \mathcal{U}_{\mathrm{M}}$.

3.4.5. Ellipsoidal Uncertainty

In this section, we move away from polyhedral uncertainty sets by considering ellipsoidal sets which are widely used in robust optimization, cf. Section 2.6. The results presented here are joint work with Andrea Maier [MS18].

Let us assume that the uncertainty set $\mathcal{U}$ has the form

$$\mathcal{U} = \left\{\xi \in \mathbb{N}^{|J|} \colon (\xi - a)^T A^{-1}(\xi - a) \leq 1\right\} \subseteq \mathrm{E}(A,a) \subseteq \mathbb{R}^{|J|}_{\geq 0}$$

for a vector $a \in \mathbb{N}^{|J|}$ and a positive definite matrix $A \in \mathbb{Q}^{|J|\times|J|}$. Note that, by assumption, we have $x \geq 0$ for every vector x in the underlying ellipsoid $\mathrm{E}(A, a)$. Thus, the uncertainty set $\mathcal{U}$ contains all integral vectors of $\mathrm{E}(A, a)$. We start with analyzing the complexity of Robust Min q-MSMC with ellipsoidal uncertainty. Here, we need to subdivide our proofs depending on the value of q. For $q \geq 3$, we can recover previous results.

Theorem 3.68. *For any fixed $q \geq 3$, Robust Min q-MSMC with ellipsoidal uncertainty is* APX*-hard.*

Proof. For $q \geq 3$, we directly obtain the result from APX-completeness of Min q-MSMC with demand vector $d \in \mathbb{N}_{>0}^{|J|}$ (Theorem 3.14) by choosing $\mathcal{U}$ to be a closed Euclidean ball around d with radius $1/2$, i.e.,

$$\mathcal{U} := \left\{\xi \in \mathbb{N}^{|J|} : 4 \cdot (\xi - d)^T (\xi - d) \leq 1\right\} = \{d\} .$$

Note that the underlying ellipsoid $\left\{x \in \mathbb{R}^{|J|} : 4 \cdot (x - d)^T (x - d) \leq 1\right\}$ is contained in $\mathbb{R}_{\geq 0}^{|J|}$ as $d > 0$. □

For $q = 1$ and $q = 2$, we have to use a different approach. As in Section 3.4.1 we consider Max 3-SAT(3) to provide appropriate L-reductions. To that end, recall Assumption 3.54 which we utilize frequently in the subsequent proofs.

Theorem 3.69. *Robust Min 2-MSMC with ellipsoidal uncertainty is* APX*-hard.*

Proof. We present an L-reduction from Max 3-SAT(3) which is a well-known APX-complete problem [GJ79; Aus+02]. Let such an instance $\mathcal{I}$ be given, i.e., we have $n \in \mathbb{N}_{>0}$ variables $X_1, \ldots, X_n$ and $m \in \mathbb{N}_{>0}$ clauses $C_1, \ldots, C_m$, each containing at most three literals from the set of literals $\left\{X_1, \overline{X}_1, \ldots, X_n, \overline{X}_n\right\}$. We construct the following bipartite graph $G = (V, E)$: Let

$$I := \{x_1, \bar{x}_1, \ldots, x_n, \bar{x}_n, y_1, \ldots y_m\} ,$$
$$J := \{a_1, \ldots, a_n, b_1, \ldots b_m, c_1, \ldots c_m\} ,$$

so that $V = I \cup J$. For every variable X_i, we add the edges $[x_i, a_i]$ and $[\bar{x}_i, a_i]$. For every clause C_j, we add the edges $[y_j, b_j]$ and $[y_j, c_j]$ and, for every literal $L \in C_j$, we add the edge $[l, c_j]$ with l being the location corresponding to L.

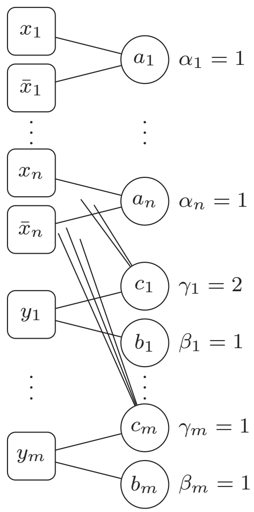


Figure 3.11.: Bipartite graph G constructed in the proof of Theorem 3.69.

Furthermore, for $p \coloneqq n + 2m$, we set

$$\mathcal{U} \coloneqq \left\{ (\alpha, \beta, \gamma) \in \mathbb{N}^p : \sum_{i=1}^{n} 2(\alpha_i - 1)^2 + \sum_{j=1}^{m} \left(2(\beta_j - 1)^2 + (\gamma_j - 1)^2 \right) \leq 1 \right\}$$
$$= \mathrm{E}(A^{-1}, 1) \cap \mathbb{N}^p,$$

where $A \in \mathbb{N}^{p \times p}$ is a diagonal matrix with value 2 in the first $n + m$ columns and value 1 for the remaining columns. Clearly, the matrix A is positive definite. Consider a scenario $(\alpha, \beta, \gamma) \in \mathcal{U}$. For every $i \in \{1, \ldots, n\}$, the entry α_i describes the demand of a_i. For every $j \in \{1, \ldots, m\}$, the entries β_j and γ_j describe the demand of b_j and c_j, respectively. Note that the only possible scenarios are of the form $(1, 1, 1)$ or $(1, 1, 1) \pm (0, 0, e_r)$ for $r \in \{1, \ldots, m\}$. Hence, it suffices to focus on scenarios where, for some $j \in \{1, \ldots, m\}$, the demand of region c_j is 2 and all other regions have unit demand. Furthermore, $\mathrm{E}(A^{-1}, 1) \subseteq \mathbb{R}^{n+2m}_{\geq 0}$ since any negative entry would not satisfy the given ellipsoidal constraint. Clearly, constructing this instance $\mathcal{I}'$ of Robust 2-MSMC can be carried out in polynomial time. An example of the constructed graph G with one possible scenario is shown in

Figure 3.11.

Now, suppose we have an optimal truth assignment for $\mathcal{I}$ satisfying $k \in \mathbb{N}$ of the m clauses. To simplify notation, we utilize a mapping $f' \colon I \to \mathbb{N}$ to represent a solution to $\mathcal{I}'$. If X_i is TRUE, we set $f'(x_i) := 1$ and zero otherwise. Analogously, we proceed with $\overline{X}_i$. Since either X_i or $\overline{X}_i$ is TRUE, it holds that $f'(x_i) + f'(\bar{x}_i) = 1$ for every i. Furthermore, we set $f'(y_j) := 1$ if clause C_j is satisfied and $f'(y_j) := 2$ otherwise. Thus, in total we have a solution value of $n + 2m - k$. We claim that f' is a solution to $\mathcal{I}'$.

Let a scenario $(1,1,1) + (0,0,e_r)$ for some $r \in \{1,\dots,m\}$ be given. For $i \in \{1,\dots,n\}$, the demand of each region a_i is either covered by x_i or $\bar{x}_i$. For $j \in \{1,\dots,m\}$, we have $f'(y_j) \geq 1$ and the demand of each region b_j is covered as well as the demand of each region $c_j \neq c_r$. For region c_r, one of its two clients can be served by y_r. If clause C_r is satisfied by the truth assignment, then there is a literal $L \in C_r$ with $f'(l) = 1$ for the corresponding location l. Hence, location l can cover the remaining client in c_r as $q = 2$. Otherwise, we have set $f'(y_r) = 2$ and the remaining demand of c_r is covered as well. Thus, f' is feasible and we obtain

$$\begin{aligned} \mathrm{OPT}(\mathcal{I}') &\leq n + 2m - \mathrm{OPT}(\mathcal{I}) \\ &\overset{(3.17)}{\leq} \frac{3}{2}m + 2m - \mathrm{OPT}(\mathcal{I}) = \frac{7}{2}m - \mathrm{OPT}(\mathcal{I}) \overset{(3.16)}{\leq} 6 \cdot \mathrm{OPT}(\mathcal{I}). \end{aligned} \tag{3.26}$$

On the other hand, let $f' \colon I \to \mathbb{N}$ be a solution to $\mathcal{I}'$. By construction, it must hold that $f'(y_j) \geq 1$ for all $j \in \{1,\dots,m\}$ and $f'(x_i) + f'(\bar{x}_i) \geq 1$ for all $i \in \{1,\dots,n\}$. First of all, we need to adjust our solution slightly. Note that we can assume $f'(y_j) = 1$ for every j, as otherwise we can switch the additional suppliers to some other location l adjacent to c_j. In a second step, if there is a location x_i with $f'(x_i) \geq 2$, we can decrease the value to 1 without loosing feasibility by construction of the scenarios. The same holds for a location $\bar{x}_i$ with $f'(\bar{x}_i) \geq 2$. Thus, suppose our solution f' already has this form with $\mathrm{SOL}(\mathcal{I}', f') = n + m + k$ for some $k \in \mathbb{N}$. If $f'(x_i) = 1$ and $f'(\bar{x}_i) = 0$, we set the variable X_i to TRUE. If the values are switched, we assign X_i the value FALSE. Additionally, if both $f'(x_i) = 1$ and $f'(\bar{x}_i) = 1$, we choose the literal $L \in \left\{X_i, \overline{X}_i\right\}$ to be TRUE which appears in at least as many clauses as its negation. If X_i and $\overline{X}_i$ both appear once, we choose X_i to be TRUE. Call the constructed truth assignment f.

Fix a variable X_i with $f'(x_i) = f'(\bar{x}_i) = 1$. By Assumption 3.54 the variable X_i appears at most three times in the instance $\mathcal{I}$ of Max 3-SAT(3).

Furthermore, it appears at least one time negated and at least one time non-negated. Hence, at most one clause containing X_i or $\overline{X_i}$ is evaluated to FALSE by our truth assignment. Thus, in total at most k clauses are false and our truth assignment f for $\mathcal{I}$ satisfies at least $m-k$ clauses. This implies

$$\begin{aligned}\mathrm{OPT}(\mathcal{I}) - \mathrm{SOL}(\mathcal{I}, f) &\leq \mathrm{OPT}(\mathcal{I}) - m + k \\ &\overset{(3.26)}{\leq} n + 2m - \mathrm{OPT}(\mathcal{I}') - m + k \\ &= n + m + k - \mathrm{OPT}(\mathcal{I}') \\ &= \mathrm{SOL}(\mathcal{I}', f') - \mathrm{OPT}(\mathcal{I}').\end{aligned}$$

The claim follows as our adjustments only improved the solution f'. □

In the remaining case $q = 1$, we use a similar construction as in the proof of Theorem 3.69. Yet, as $q = 1$, we need to enforce the existence of some additional suppliers.

Theorem 3.70. *Robust Min 1-MSMC with ellipsoidal uncertainty is* APX-*hard.*

Proof. As above, let an instance $\mathcal{I}$ of Max 3-SAT(3) be given. Since every supplier can only cover one client, we need some auxiliary locations and regions. We set

$$\begin{aligned}I &:= \{x_1, \bar{x}_1, \ldots, x_n, \bar{x}_n, y_1, \ldots y_m, z_1, \ldots, z_n\}\,, \\ J &:= \{a_1, \ldots, a_n, b_1, \ldots b_m, c_1, \ldots, c_m, d_1, \ldots, d_n\}\,,\end{aligned}$$

so that the vertex set of the constructed bipartite graph G is given by $V = I \cup J$. The edge set E consists of the edges in the proof of Theorem 3.69 plus the edges $[z_i, a_i]$ and $[z_i, d_i]$ for $i \in \{1, \ldots, n\}$. For $p := 2(n + m)$ we define the positive definite matrix $A \in \mathbb{N}^{p \times p}$ as a diagonal matrix with value 2 on the diagonal for the first n columns and value 1 otherwise. We set $\mathcal{U} := \mathrm{E}(A^{-1}, 1) \cap \mathbb{N}^p$ so that any vector $(\alpha, \beta, \gamma, \delta) \in \mathcal{U}$ satisfies

$$\sum_{i=1}^{n} \left(2(\alpha_i - 1)^2 + (\delta_i - 1)^2\right) + \sum_{j=1}^{m} \left((\beta_j - 1)^2 + (\gamma_j - 1)^2\right) \leq 1.$$

Call the constructed instance $\mathcal{I}'$. The interpretation of the vectors α, β, and γ is as in Theorem 3.69 while the vector δ describes the demand of the

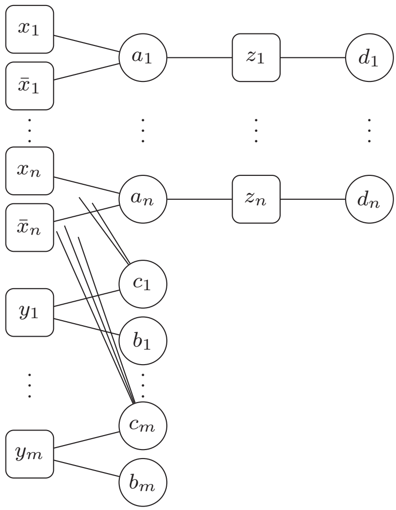


Figure 3.12.: Bipartite graph G constructed in the proof of Theorem 3.70.

regions $d_1, \ldots, d_n$. Again, it suffices to focus on the scenarios of the form $(1,1,1,1) + (0, e_r)$ with $r \in \{1, \ldots, n+2m\}$ and we have $\mathrm{E}(A^{-1}, 1) \subseteq \mathbb{R}^p_{\geq 0}$. This construction can be accomplished in polynomial time. An example of the constructed graph is illustrated in Figure 3.12.

Now, suppose we have an optimal truth assignment for $\mathcal{I}$ satisfying $k \in \mathbb{N}$ of the m clauses. As in the proof of Theorem 3.69 we use a mapping $f'\colon I \to \mathbb{N}$ to construct a solution to $\mathcal{I}'$. If X_i is TRUE, we set $f'(x_i) := 1$ and zero otherwise. Analogously, we proceed with $\overline{X}_i$. Furthermore, we set $f'(z_i) := 2$ for all $i \in \{1, \ldots, n\}$. For $j \in \{1, \ldots, m\}$, we set $f'(y_j) := 2$ if clause C_j is true and $f'(y_j) := 3$ otherwise. Thus, in total we have $\mathrm{SOL}(\mathcal{I}', f') = 3n + 2k + 3(m-k) = 3n + 3m - k$. We claim that f' is feasible for $\mathcal{I}'$.

Let $(\alpha, \beta, \gamma, \delta) = (1,1,1,1) + (0, e_r) \in \mathcal{U}$ for some $r \in \{1, \ldots, n+2m\}$ be given. Depending on the value of r we distinguish three cases:

(a) If $\beta_j = 2$ for some j, the demand of region b_j is covered by y_j. If clause C_j is true, there is one literal $L \in C_j$ with $f'(l) = 1$ for its corresponding location l. Thus, the demand of region c_j can be covered by location l. If clause C_j is false, we have $f'(y_j) = 3$ so that y_j also covers region c_j. Furthermore, the demand of the regions a_i and d_i are covered by z_i for all i. The demand of all remaining regions $b_{j'}$ and $c_{j'}$ for $j' \neq j$ is covered by the suppliers in $y_{j'}$.

(b) If $\gamma_j = 2$ for some j, the argumentation is analogous to item (a).

(c) If $\delta_i = 2$ for some i, the demand of region d_i is covered by z_i. For every i', the demand of region $a_{i'}$ is covered by $x_{i'}$ or $\bar{x}_{i'}$ and, for every j, the demand of the regions c_j and b_j is covered by y_j.

Thus, we obtain

$$\begin{aligned} \mathrm{OPT}(\mathcal{I}') &\leq 3n + 3m - \mathrm{OPT}(\mathcal{I}) \\ &\overset{(3.17)}{\leq} \frac{15}{2}m - \mathrm{OPT}(\mathcal{I}) \overset{(3.16)}{\leq} 14 \cdot \mathrm{OPT}(\mathcal{I}). \end{aligned} \tag{3.27}$$

On the other hand, let $f'\colon I \to \mathbb{N}$ be a solution to $\mathcal{I}'$. Due to the scenarios, we know that $f'(y_j) \geq 2$ for all j and $f'(z_i) \geq 2$ for all i. By the analogous switching argument as in the proof of Theorem 3.69, we can assume $f'(y_j) = f'(z_i) = 2$ for all i, j. Then, it also holds that $f'(x_i) + f'(\bar{x}_i) \geq 1$ for all i due to the scenarios with $\delta_i = 2$. We can additionally ensure that $f'(x_i) \leq 1$ as well as $f'(\bar{x}_i) \leq 1$ by construction of $\mathcal{U}$. Hence, let $\mathrm{SOL}(\mathcal{I}', f') = 3n + 2m + k$ for some $k \in \mathbb{N}$. We obtain $\sum_{i=1}^{n} f'(x_i) + f'(\bar{x}_i) = n + k$ with k being the number of pairs $(x_i, \bar{x}_i)$ such that $f'(x_i) = f'(\bar{x}_i) = 1$. Now, we replicate the truth assignment f of the proof of Theorem 3.69. This gives a solution to $\mathcal{I}$ with at least $m - k$ satisfied clauses implying

$$\begin{aligned} \mathrm{OPT}(\mathcal{I}) - \mathrm{SOL}(\mathcal{I}, f) &\leq \mathrm{OPT}(\mathcal{I}) - m + k \\ &\overset{(3.27)}{\leq} 3n + 3m - \mathrm{OPT}(\mathcal{I}') - m + k \\ &= 3n + 2m + k - \mathrm{OPT}(\mathcal{I}') \\ &= \mathrm{SOL}(\mathcal{I}', f') - \mathrm{OPT}(\mathcal{I}'). \end{aligned}$$

Again, the claim follows as our adjustments only improved the given solution f'. $\square$

Corollary 3.71. *For any fixed $q \in \mathbb{N}_{>0}$, Robust q-MSMC with ellipsoidal uncertainty is strongly* NP*-hard.*

Proof. For $q = 1$, we obtain from the proof of Theorem 3.70 that a solution to $\mathcal{I}$ with at least $k \in \mathbb{N}$ satisfied clauses leads to a solution to $\mathcal{I}'$ with value at most $3n + 3m - k$ in polynomial time and vice versa.

For $q = 2$, we obtain from the proof of Theorem 3.69 that a solution to $\mathcal{I}$ with at least $k \in \mathbb{N}$ satisfied clauses leads to a solution to $\mathcal{I}'$ with value at most $n + 2m - k$ in polynomial time and vice versa.

For $q = 3$, the result follows from Theorem 3.15. □

By Corollary 3.71 and the previous theorems it is necessary to analyze the separation problem corresponding to Robust Min q-MSMC with ellipsoidal uncertainty. To that end, we reconsider the Independent Set problem, cf. Problem 3.21. We show that Robust Sum with ellipsoidal uncertainty is strongly NP-complete which also leads to NP-completeness of Separation for this type of uncertainty.

Theorem 3.72. *Robust Sum with ellipsoidal uncertainty is strongly* NP-*complete even for $S = \{1, \ldots, n\}$.*

Proof. As in Lemma 3.22 we get that Robust Sum with ellipsoidal uncertainty is contained in NP. To show NP-hardness of Robust Sum with ellipsoidal uncertainty we provide a polynomial time reduction from Independent Set. Hence, let an instance of Independent Set be given, i.e., a simple graph $G = (V, E)$ and a positive integer $B \leq |V|$. Let $V = \{1, \ldots, n\}$ for $n \in \mathbb{N}_{>0}$ and let $M \in \mathbb{B}^{n \times n}$ be the adjacency matrix of G, i.e., $M_{uv} = 1$ if and only if the vertices u and v are adjacent in G. Consider the matrix

$$A \coloneqq \frac{1}{2B} \left(2 \cdot I_n + \frac{1}{n} \cdot M \right) \in \mathbb{Q}^{n \times n}.$$

As G contains no loops, we get that $A_{vv} > \sum_{u \neq v} |A_{vu}|$ for every $v \in V$. Hence, by [HJ12] the matrix A is positive definite, cf. Section 2.1, and according to [Var75] we have

$$\left|A^{-1}_{vv}\right| \leq \max_{v' \in V} \sum_{u \in V} \left|A^{-1}_{v'u}\right| \leq \frac{1}{\frac{1}{B} - \frac{n-1}{2Bn}} \leq 2Bn \tag{3.28}$$

for $v \in V$. Consider the ellipsoid $\mathrm{E}(A^{-1}, 0)$. By (3.28) and Lemma 2.1 we get that $x_v \in [-2Bn, 2Bn]$ for every $x \in \mathrm{E}(A^{-1}, 0)$ and $v \in V$. Hence, with $K \coloneqq 2Bn$ and $a \coloneqq K \cdot \mathbb{1}$ we have $\mathrm{E}(A^{-1}, a) \subseteq \mathbb{R}^n_{\geq 0}$ and we define

$\mathcal{U} \coloneqq \mathrm{E}(A^{-1}, a) \cap \mathbb{N}^n$. Observe that A^{-1} can be computed in polynomial time from A. For a scenario $\xi \in \mathcal{U}$, we write $\xi = a + x$ with $x \in \mathbb{Z}^n$ and we infer

$$\begin{aligned} & (\xi - a)^T A (\xi - a) \leq 1 \\ \Leftrightarrow \quad & x^T A x \leq 1 \\ \Leftrightarrow \quad & \sum_{v \in V} 2 \cdot x_v^2 + \sum_{[u,v] \in E} \frac{2}{n} \cdot x_u \cdot x_v \leq 2B \\ \Leftrightarrow \quad & \sum_{v \in V} n \cdot x_v^2 + \sum_{[u,v] \in E} x_u \cdot x_v \leq nB. \end{aligned} \tag{3.29}$$

Finally, we set $S \coloneqq V$ and $B' \coloneqq nK + B$.

Let $V' \subseteq V$ be an independent set of G with $|V'| \geq B$. If necessary, we delete vertices from V' so that $|V'| = B$. For $v \in V$, we set $\bar{x}_v \coloneqq 1$ if $v \in V'$ and zero otherwise. Consider the scenario $\xi \coloneqq a + \bar{x}$. With (3.29) we get that

$$\sum_{v \in V} n \cdot \bar{x}_v^2 + \sum_{[u,v] \in E} \bar{x}_u \cdot \bar{x}_v = n \cdot |V'| = nB$$

so that $\xi \in \mathcal{U}$ with $\xi(S) = nK + \bar{x}(V) = nK + B = B'$.

Now let $\xi \in \mathcal{U}$ be a scenario with $\xi(V) \geq B' = nK + B$. First of all, we rewrite $\xi = a + \bar{x}$ for $\bar{x} \in \mathbb{Z}^n$ so that $\bar{x}(V) \geq B$. We show that we can assume $\bar{x} \geq 0$. First, assume there is $v \in V$ with $\bar{x}_v < 0$ and $|\bar{x}_v| \geq |\bar{x}_u|$ for every $u \in N(v)$. Then, it holds true that

$$n \cdot \bar{x}_v^2 + \sum_{u \in N(v)} \bar{x}_v \cdot \bar{x}_u \geq n \cdot \bar{x}_v^2 - \sum_{u \in N(v)} |\bar{x}_v| \cdot |\bar{x}_u| \geq n \cdot \bar{x}_v^2 - \sum_{u \in N(v)} \bar{x}_v^2 \geq 0.$$

Hence, we can improve our given scenario without loosing feasibility by reassigning $\bar{x}_v \coloneqq 0$. Now, assume that, for every $v \in V$ with $\bar{x}_v < 0$, there is some $u' \in N(v)$ with $|\bar{x}_v| < |\bar{x}_{u'}|$. Let $v' \in V$ be a vertex with $\bar{x}_{v'} \leq \bar{x}_v$ for every $v \in V$. If $\bar{x}_{v'} \geq 0$, we get that $\bar{x} \geq 0$. Otherwise, there is $u' \in N(v')$ with $|\bar{x}_{v'}| < |\bar{x}_{u'}|$. By choice of v' we must have $|\bar{x}_{v'}| < \bar{x}_{u'}$. Furthermore, without loss of generality we can assume that $\bar{x}_{u'} \geq \bar{x}_u$ for every $u \in N(v')$. Consider the new scenario $\xi' \coloneqq a + x'$ with

$$x'_v \coloneqq \begin{cases} \bar{x}_{v'} + \bar{x}_{u'} & \text{if } v = u', \\ 0 & \text{if } v = v', \\ \bar{x}_v & \text{otherwise,} \end{cases}$$

for $v \in V$. Observe that $x'(V) = \bar{x}(V)$ and that the number of negative entries in x' is one less than the number of negative entries in $\bar{x}$. We show that $\xi' \in \mathcal{U}$. With (3.29) we get that

$$
\begin{aligned}
&\sum_{v \in V} n \cdot (x'_v)^2 + \sum_{[u,v] \in E} x'_u \cdot x'_v \\
&\qquad = \sum_{v \in V \setminus \{v', u'\}} n \cdot \bar{x}_v^2 + n \cdot (\bar{x}_{v'} + \bar{x}_{u'})^2 + \sum_{[u,v] \in E} \bar{x}_u \cdot \bar{x}_v - \bar{x}_{u'} \cdot \bar{x}_{v'} \\
&\qquad\quad - \sum_{\substack{u \in N(u'),\\ u \neq v'}} \bar{x}_u \cdot \bar{x}_{u'} - \sum_{\substack{u \in N(v'),\\ u \neq u'}} \bar{x}_u \cdot \bar{x}_{v'} + \sum_{\substack{u \in N(u'),\\ u \neq v'}} \bar{x}_u \cdot (\bar{x}_{v'} + \bar{x}_{u'}) \\
&\qquad \leq nB + (2n-1) \cdot \bar{x}_{u'} \cdot \bar{x}_{v'} - \sum_{\substack{u \in N(v'),\\ u \neq u'}} \bar{x}_u \cdot \bar{x}_{v'} + \sum_{\substack{u \in N(u'),\\ u \neq v'}} \bar{x}_u \cdot \bar{x}_{v'}.
\end{aligned}
$$

Thus, we get $\xi' \in \mathcal{U}$ if we can show that

$$
(2n-1) \cdot \bar{x}_{u'} \geq \sum_{\substack{u \in N(v'),\\ u \neq u'}} \bar{x}_u - \sum_{\substack{u \in N(u'),\\ u \neq v'}} \bar{x}_u. \tag{3.30}
$$

By choice of u' we have

$$
\sum_{\substack{u \in N(v'),\\ u \neq u'}} \bar{x}_u \leq \sum_{\substack{u \in N(v'),\\ u \neq u'}} \bar{x}_{u'} \leq (n-1) \cdot \bar{x}_{u'}.
$$

Furthermore, by choice of v' we have, for $u \in V$, $\bar{x}_{u'} > |\bar{x}_{v'}| \geq |\bar{x}_u|$ whenever $\bar{x}_u < 0$. Hence, we get

$$
\sum_{\substack{u \in N(u'),\\ u \neq v'}} -\bar{x}_u \leq \sum_{\substack{u \in N(u'),\\ u \neq v', \bar{x}_u < 0}} -\bar{x}_u \leq \sum_{\substack{u \in N(u'),\\ u \neq v', \bar{x}_u < 0}} \bar{x}_{u'} \leq (n-1) \cdot \bar{x}_{u'}.
$$

Therefore, Equation (3.30) is fulfilled and $\xi' \in \mathcal{U}$. If there is $v \in V$ with $x'_v < 0$ we can restart the whole procedure. Thus, after at most n iterations we have a scenario $\xi = a + \bar{x}$ in $\mathcal{U}$ with $\bar{x} \in \mathbb{N}^n$ and $\bar{x}(V) \geq B$. First of all, assume that $\bar{x}(V) > B$. But then (3.29) cannot be fulfilled as $\sum_{v \in V} \bar{x}_v^2 > B$ and $\bar{x} \geq 0$. Thus, $\bar{x}(V) = B$. If $\sum_{v \in V} \bar{x}_v^2 > B$, we again obtain a contradiction to (3.29) as $\bar{x} \geq 0$. Hence, $\sum_{v \in V} \bar{x}_v^2 \leq B$. If $\sum_{v \in V} \bar{x}_v^2 < B$, we get that also $\bar{x}(V) < B$ as $\bar{x} \in \mathbb{N}^n$ in contradiction to $\bar{x}(V) = B$. Thus, we also have that $\sum_{v \in V} \bar{x}_v^2 = B$ and, hence, $\bar{x}_u \cdot \bar{x}_v = 0$ for every edge $[u, v] \in E$. Now, assume there is $u \in V$ with $\bar{x}_u^2 > \bar{x}_u$. Then, we have

$$\sum_{v \neq u} \bar{x}_v \leq \sum_{v \neq u} \bar{x}_v^2 = B - \bar{x}_u^2 < B - \bar{x}_u$$

in contradiction to $\bar{x}(V) = B$. Hence, we must have $\bar{x}_v \in \mathbb{B}$ for every $v \in V$. We choose $V' := \{v \in V : \bar{x}_v = 1\}$. By our above argumentation we know that V' is an independent set of G of size B. □

Note that the previous theorem shows that it is NP-hard to maximize a linear function over the integral points of an ellipsoid even if the coefficients of the linear function are binary and the set of feasible solutions is non-empty. As in Theorem 3.38 we can apply the construction of the previous proof to also show NP-hardness of Separation with ellipsoidal uncertainty.

Corollary 3.73. *For any fixed $q \in \mathbb{N}_{>0}$, Separation and Feasibility with ellipsoidal uncertainty are strongly* NP- *and* co-NP*-complete, respectively.*

Proof. We see that Separation with ellipsoidal uncertainty is contained in NP, cf. Theorem 3.38. To show NP-hardness we again consider an instance of Independent Set with a simple graph $G = (V, E)$ and a positive integer $B \in \mathbb{N}_{>0}$. Let $V = \{1, \ldots, n\}$ for some $n \in \mathbb{N}_{>0}$ and let G' be the simple graph that is the union of q copies of G. We apply the construction in the proof of Theorem 3.72 to the graph G' to obtain our instance of Separation with ellipsoidal uncertainty. That means, with $M' \in \mathbb{B}^{qn \times qn}$ being the adjacency matrix of G', we have

$$A := \frac{1}{2(qB)} \cdot \left(2 \cdot I_{qn} + \frac{1}{qn} \cdot M'\right) \in \mathbb{Q}^{qn \times qn}$$

and $a := K \cdot \mathbb{1}$ with $K := 2 \cdot qB \cdot qn = 2Bq^2n$. The uncertainty set is given by $\mathcal{U} := \mathrm{E}(A^{-1}, a) \cap \mathbb{N}^{qn}$ and fulfills the required properties. We set $I := \{0\}$, $J := \{1, \ldots, qn\}$, and $E := \{[0, j] : j \in J\}$ to obtain the bipartite graph $H = (I \cup J, E)$. Furthermore, we set $\bar{x}_0 := 2Bq^2n^2 + B - 1$.

An independent set $V' \subseteq V$ of G with $|V'| = B$ directly leads to an independent set in G' of size exactly $q \cdot B$. Analogously to the proof of Theorem 3.72 we find a scenario $\xi \in \mathcal{U}$ with $\xi(J) = qn \cdot K + q \cdot B =$

$q \cdot (2Bq^2n^2 + B) > q \cdot \bar{x}_0$. Thus, the set J is a solution to the constructed instance of Separation.

On the other hand, let a subset $S \subseteq J$ and a scenario $\xi \in \mathcal{U}$ with $\xi(S) > q \cdot \bar{x}(N_H(S))$ be given. Then, S is non-empty and we get $q \cdot \bar{x}(N_H(S)) = q \cdot (2Bq^2n^2 + B - 1)$. Analogously to the proof of Theorem 3.72 we can write $\xi = a + \bar{x}$ for $\bar{x} \in \mathbb{N}^{qn}$. Then, we have $\bar{x}(J) > q \cdot (B-1)$. By construction of $\mathcal{U}$ there is some $i' \in \{0, \dots, q-1\}$ with $\sum_{v \in V} \bar{x}_{i' \cdot n + v} \geq B$ as otherwise we get $\bar{x}(J) \leq q \cdot (B-1)$. Analogously to the proof of Theorem 3.72 we get that $V' := \{v \in V : \bar{x}_{i' \cdot n + v} = 1\}$ is an independent set of G with $|V'| = B$. □

Remark 3.74. Along the lines of the above proof we can show that Robust Sum with ellipsoidal uncertainty is NP-complete even if the bound $B \in \mathbb{N}_{>0}$ of a given instance is a multiple of q for some fixed $q \in \mathbb{N}_{>0}$. Similarly, we can also see this with Theorem 3.72 as Independent Set remains NP-complete if the given bound is a multiple of q, cf. Remark 3.23. This observation will be of use in Chapter 5.

Notably, when relaxing the integrality constraints on the scenarios, Max Robust Sum with ellipsoidal uncertainty is solvable due to Lemma 2.1. Furthermore, we can use this lemma to deduce

$$\max_{\xi \in \mathcal{U}} \xi_j \leq a_j + \sqrt{A_{jj}}.$$

for an ellipsoidal uncertainty set $\mathcal{U} = \mathrm{E}(A, a) \cap \mathbb{N}^{|J|}$. Thus, for the separation step, we solve the following convex integer program in the case of ellipsoidal uncertainty $\mathcal{U}_\mathrm{E}$ where $M := a_j + \sqrt{A_{jj}}$:

$$\begin{array}{lllr}
\mathrm{Sep}^{\mathbb{B}}(\mathcal{U}_\mathrm{E}) & \displaystyle\min_{\mu,\nu,\xi,\omega} & \displaystyle\sum_{i \in I} q \cdot \bar{x}_i \cdot \mu_i - \sum_{j \in J} \omega_j & \\
& \text{s.t.} & \displaystyle |N(i)| \cdot \mu_i \geq \sum_{j \in N(i)} \nu_j & \text{for } i \in I \\
& & \omega_j \leq \xi_j & \text{for } j \in J \\
& & \omega_j \leq \nu_j \cdot M & \text{for } j \in J \\
& & (\xi - a)^T A^{-1} (\xi - a) \leq 1 & \\
& & \mu_i, \nu_j \in \mathbb{B} & \text{for } i \in I, j \in J \\
& & \xi_j \in \mathbb{N} & \text{for } j \in J.
\end{array}$$

The interpretation of the constraints follows readily from $\mathrm{Sep}^{\mathbb{B}}(\mathcal{U})$ on Page 63 and the definition of $\mathcal{U}_{\mathrm{E}}$. Of course, it is not clear whether $\sqrt{A_{jj}} \in \mathbb{Q}$ for every $j \in J$. For better readibility we use this upper bound here. To ensure encoding we can also use a poorer bound by approximating $\sqrt{A_{jj}}$ from above.

3.4.6. Γ-Uncertainty

As a final common uncertainty set we consider Γ-uncertainty. Let a nominal vector $a \in \mathbb{N}^{|J|}$ and a deviation $\hat{a} \in \mathbb{N}^{|J|}$ be given. Similar to interval uncertainty, the demand ξ_j of region j varies in a given interval $[a_j - \hat{a}_j, a_j + \hat{a}_j]$ around its nominal value $a_j \in \mathbb{N}$. But now, the number of regions whose demand deviates from its nominal value is upper bounded by an integer $\Gamma \in \{0, 1, \ldots, |J|\}$, cf. Section 2.6. In our context, it suffices to focus on positive deviations. Thus, we consider the uncertainty set

$$\mathcal{U} = \left\{ \xi \in \mathbb{N}^{|J|} \colon a \leq \xi \leq a + \hat{a}, |\{k \colon \xi_k \neq a_k\}| \leq \Gamma \right\}. \tag{3.31}$$

By the same arguments as in Section 3.4.3 we refrain from considering only scenarios where the demand of exactly Γ regions deviates. By Remark 3.33 we directly obtain the following result on the complexity of Robust Min q-MSMC with Γ-uncertainty:

Theorem 3.75. *For any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-MSMC with Γ-uncertainty is* APX*-hard and its decision version is strongly* NP*-hard even for $a = 0$ and $\hat{a} = 1$.*

Proof. Setting the parameters to $a \coloneqq 0$, $\hat{a} \coloneqq 1$, and $\Gamma \coloneqq 1$ yields the uncertainty set $\mathcal{U} = \{0, e_1, \ldots, e_{|J|}\}$ as in the proof of Theorem 3.30. □

Thus, the consideration of Separation and Robust Sum with Γ-uncertainty is of interest. Due to the basic structure of Γ-uncertainty sets and their relation to budgeted uncertainty sets, cf. [Ata06], we deduce the following result.

Lemma 3.76. *Max Robust Sum with Γ-uncertainty can be solved in time $\mathcal{O}(|S|)$.*

Proof. Let $\mathcal{U} \subseteq \mathbb{N}^n$ be an uncertainty set of the form (3.31) with $J = \{1, \ldots, n\}$ and $S \subseteq J$. By definition, it holds true that

$$\max_{\xi \in \mathcal{U}} \xi(S) = a(S) + \max \{\hat{a}(T) \colon T \subseteq S, |T| \leq \Gamma\}.$$

If $\Gamma \geq |S|$, this evaluates to $a(S) + \hat{a}(S)$. Otherwise, we can find the Γ-largest element in $\{\hat{a}_j \colon j \in S\}$, say $\alpha \in \mathbb{N}$, in time $\mathcal{O}(|S|)$ using a linear time selection algorithm, cf. [Cor+09]. Then, $\max \{\hat{a}(T) \colon T \subseteq S, |T| \leq \Gamma\}$ is obtained by first summing up all values in $\{\hat{a}_j \colon j \in S\}$ with value greater than α. Suppose this sum consists of $r \in \mathbb{N}$ values. Then, $r < \Gamma$ and we add $(\Gamma - r) \cdot \alpha$. Again, this takes time $\mathcal{O}(|S|)$. ☐

As with budgeted uncertainty, Γ-uncertainty allows to compute a violating extreme scenario given a violating subset in linear time. Therefore, concerning Separation with Γ-uncertainty, we seek a subset $S \subseteq J$ with

$$q \cdot \bar{x}(N(S)) < a(S) + \max_{T \subseteq S, |T| \leq \Gamma} \hat{a}(T).$$

Unfortunately, we see that Separation with Γ-uncertainty is NP-complete. To prove this result, we consider the special case where $a = 0$. For this case, the problem simplifies as shown below.

Lemma 3.77. *For $a = 0$, Separation with Γ-uncertainty is equivalent to asking for a subset $T \subseteq J$ with $|T| \leq \Gamma$ and*

$$q \cdot \bar{x}(N(T)) < \hat{a}(T). \tag{3.32}$$

Proof. Let an instance of Separation with Γ-uncertainty be given and let $S \subseteq J$ be a subset with $q \cdot \bar{x}(N(S)) < \max \{\hat{a}(T) \colon T \subseteq S,\ |T| \leq \Gamma\}$. Further, let $T' \subseteq S$ be an optimal solution to the expression on the right-hand side. Then, we get

$$q \cdot \bar{x}(N(T')) \leq q \cdot \bar{x}(N(S)) < \max_{T \subseteq S, |T| \leq \Gamma} \hat{a}(T) = \hat{a}(T').$$

On the other hand, given a subset $T' \subseteq J$ with $|T'| \leq \Gamma$ satisfying (3.32), we can choose $S := T'$ and obtain

$$q \cdot \bar{x}(N(S)) < \hat{a}(S) = \max_{T \subseteq S, |T| \leq \Gamma} \hat{a}(T).$$ ☐

With the help of Lemma 3.77 we obtain the mentioned hardness result. We present a polynomial time reduction from Clique which is a well-known NP-complete problem, see [GJ79].

Problem 3.78 (Clique).
Instance: A simple graph $G = (V, E)$ and a positive integer $B \leq |V|$.
Question: Does G contain a *clique* of size at least B, i.e., a subset $V' \subseteq V$ with $|V'| \geq B$ such that every two distinct vertices in V' are adjacent in G?

Observe that this problem remains NP-complete even if $B \geq 4$. Furthermore, we can restrict ourselves to instances with $|E| \geq \binom{B}{2}$ as otherwise it must be a no-instance.

Theorem 3.79. *For any fixed $q \in \mathbb{N}_{>0}$, Separation with Γ-uncertainty is strongly* NP*-complete even for $a = 0$.*

Proof. The problem is contained in NP, see Theorem 3.38. To show NP-hardness, we provide a polynomial time reduction from Clique. Thus, given an instance of this problem, i.e., a simple graph $G' = (V', E')$ and an integer B with $4 \leq B \leq |V'|$ and $|E'| \geq \binom{B}{2}$, we construct an instance of Separation with Γ-uncertainty as follows: We set $I := V'$, $J := E'$, and $E := \{[v, e] \colon v \in e\}$ to obtain the bipartite graph $G = (I \cup J, E)$. For all regions $j \in J$, we set $a_j := 0$ and $\hat{a}_j := 2qB$. Moreover, for every location $i \in I$, we define

$$\bar{x}_i := 2B \cdot \left(\frac{B-1}{2} - \frac{1}{B}\right) = B \cdot (B-1) - 2.$$

Finally, let $\Gamma := \binom{B}{2} = \frac{B(B-1)}{2} \leq |J|$. Since $a = 0$, we apply Lemma 3.77.

Given a clique $C \subseteq V'$ in the graph G' of size at least B, we fix B vertices of this clique and obtain a subset $C' \subseteq C$. To define the set $T \subseteq J$ we choose the corresponding edges of C', i.e., $T := \{[u, v] \colon u, v \in C'\}$. Then, $|N_G(T)| = B$ and $|T| = \binom{B}{2} \leq \Gamma$ and we obtain

$$\begin{aligned}\sum_{i \in N_G(T)} q \cdot \bar{x}_i &= |N_G(T)| \cdot 2qB \cdot \left(\frac{B-1}{2} - \frac{1}{B}\right) = B \cdot 2qB \cdot \left(\frac{B-1}{2} - \frac{1}{B}\right) \\ &= 2qB \cdot \left(\binom{B}{2} - 1\right) < 2qB \cdot \binom{B}{2} = 2qB \cdot |T| = \sum_{j \in T} \hat{a}_j.\end{aligned}$$

Now, given a subset $T \subseteq J$ with $|T| \le \Gamma = \binom{B}{2}$ and $q \cdot \bar{x}(N_G(T)) < \hat{a}(T)$, we claim that $N_G(T) \subseteq I$ is a clique of size B in G'. First of all, we insert the given values and reformulate the expression.

$$\begin{aligned} & |N_G(T)| \cdot q \cdot 2B \cdot \left(\frac{B-1}{2} - \frac{1}{B}\right) < |T| \cdot 2qB \le \binom{B}{2} \cdot 2qB \\ \Leftrightarrow \quad & |N_G(T)| \cdot \left(\frac{B-1}{2} - \frac{1}{B}\right) < |T| \le \binom{B}{2}. \end{aligned} \tag{3.33}$$

Suppose $|N_G(T)| \ge B+1$. Then, we have

$$\begin{aligned} |N_G(T)| \cdot \left(\frac{B-1}{2} - \frac{1}{B}\right) &\ge (B+1) \cdot \left(\frac{B-1}{2} - \frac{1}{B}\right) \\ &= \binom{B}{2} - 1 + \frac{B-1}{2} - \frac{1}{B} \\ &= \binom{B}{2} + \frac{B^2 - 3B - 2}{2B} \ge \binom{B}{2}, \end{aligned}$$

where the last inequality holds as $B^2 - 3B - 2 \ge 0$ for $B \ge 4$. This is a contradiction to (3.33). Now, suppose $|N_G(T)| \le B - 1$. Let $l := |N_G(T)|$. We get:

$$\begin{aligned} |N_G(T)| \cdot \left(\frac{B-1}{2} - \frac{1}{B}\right) &= l \cdot \left(\frac{B-1}{2} - \frac{1}{B}\right) - \binom{l}{2} + \binom{l}{2} \\ &= \frac{l\,(B-1)}{2} - \frac{l}{B} - \frac{l\,(l-1)}{2} + \binom{l}{2} \\ &= \frac{l\,(B-l)}{2} - \frac{l}{B} + \binom{l}{2} \\ &\overset{B>l}{\ge} \frac{l}{2} - \frac{l}{B} + \binom{l}{2} \ge \binom{l}{2} \ge |T|. \end{aligned}$$

Again, this contradicts (3.33). Thus, we must have $|N_G(T)| = B$. Due to (3.33) we get $|T| = \binom{B}{2}$. As G' is simple, we get that $N_G(T)$ is a clique of size B. □

Corollary 3.80. *For any fixed $q \in \mathbb{N}_{>0}$, Feasibility with Γ-uncertainty is strongly* co-NP*-complete.* □

Remark 3.81. In the proof of Theorem 3.36 we have seen that, for an instance $\mathcal{I}$ of Separation with Γ-uncertainty with $a = 0$ and bipartite graph G, computing a minimum s-t-cut in the network $H_G(q \cdot \bar{x}, \hat{a})$ leads to an answer to the question whether there exists a subset $T \subseteq J$ with $q \cdot \bar{x}(N_G(T)) < \hat{a}(T)$. In comparison to this result, Theorem 3.79 reveals that the additional constraint $|T| \leq \Gamma$ turns this problem into a hard one.

Thus, to solve the separation step in the case of Γ-uncertainty, we propose the following mixed integer program $\mathrm{Sep}_{\mathrm{s}}^{\mathbb{B}}(\mathcal{U}_\Gamma)$ for an uncertainty set $\mathcal{U}_\Gamma$ of type (3.31).

$$
\begin{aligned}
\mathrm{Sep}_{\mathrm{s}}^{\mathbb{B}}(\mathcal{U}_\Gamma) \qquad \min_{\mu,\nu,\tilde{\nu}} \quad & \sum_{i \in I} q \cdot \bar{x}_i \cdot \mu_i - \sum_{j \in J} a_j \cdot \nu_j - \sum_{j \in J} \hat{a}_j \cdot \tilde{\nu}_j && && (3.34a)\\
\text{s.t.} \quad & |N(i)| \cdot \mu_i \geq \sum_{j \in N(i)} \nu_j && \text{for } i \in I && (3.34b)\\
& \tilde{\nu}_j \leq \nu_j && \text{for } j \in J && (3.34c)\\
& \sum_{j \in J} \tilde{\nu}_j \leq \Gamma && && (3.34d)\\
& \tilde{\nu}_j \geq 0 && \text{for } j \in J && (3.34e)\\
& \mu_i, \nu_j \in \mathbb{B} && \text{for } i \in I, j \in J. && (3.34f)
\end{aligned}
$$

In an optimal solution $(\mu^\star, \nu^\star, \tilde{\nu}^\star)$ we can assume that $\tilde{\nu}^\star \in \mathbb{B}^{|J|}$ as $\Gamma \in \mathbb{N}$ and $\nu^\star \in \mathbb{B}^{|J|}$. Thus, when setting $S := \{j \in J : \nu_j^\star = 1\}$ the variables $\tilde{\nu}^\star$ encode the set $T \subseteq S$ with $|T| \leq \Gamma$ due to the constraints (3.34c) and (3.34d). Then, the term

$$\sum_{j \in J} a_j \cdot \nu_j^\star + \sum_{j \in J} \hat{a}_j \cdot \tilde{\nu}_j^\star$$

reflects the worst-case demand of the set S by Lemma 3.76. A scenario $\xi \in \mathcal{U}_\Gamma$ attaining this worst-case demand can be obtained by setting, for every region $j \in J$,

$$\xi_j = \begin{cases} a_j + \hat{a}_j, & \text{if } \nu_j^\star = \tilde{\nu}_j^\star = 1,\\ a_j, & \text{otherwise.} \end{cases}$$

After this extensive study of various classes of uncertainty sets, we briefly draw our attention to some variants of Robust q-MSMC to which all the previous reductions cannot be applied.

3.5. Bounding Locations or Regions

In the preceding section, we encountered quite a lot of NP-hardness proofs showing that Robust q-MSMC is NP-hard for a broad amount of uncertainty set classes. A common feature of all these reductions is that the number of locations or regions is not bounded by a fixed constant in the construction of the instance of Robust q-MSMC. Thus, the question arises whether this feature is, among others, responsible for NP-hardness in many cases. In this section, we aim to analyze this question by considering instances of Robust q-MSMC with $|I|$ or $|J|$ being bounded from above by a fixed constant. We start with considering the location case.

Let an instance of Robust q-MSMC with $|I| \leq k$ for some fixed $k \in \mathbb{N}$ be given. Then, there are only constantly many variables but exponentially many constraints present in $\mathrm{P_s}(\mathcal{U})$, cf. Page 48. To also decrease the number of constraints, the idea is to change the perspective by considering the problem from the supplier side. We show that $\mathrm{P_s}(\mathcal{U})$ is equivalent to the following integer program:

$$\min_{x} \quad \sum_{i \in I} x_i \tag{3.35a}$$

$$\text{s.t.} \quad \sum_{i \in I'} q \cdot x_i \geq \max_{\xi \in \mathcal{U}} \sum_{j \in S'} \xi_j \quad \text{for } I' \subseteq I \tag{3.35b}$$

$$x_i \in \mathbb{N} \quad \text{for } i \in I, \tag{3.35c}$$

with $S' := \{j \in J \colon N(j) \subseteq I'\}$.

Lemma 3.82. *A vector $x \in \mathbb{N}^{|I|}$ is a solution to $P_s(\mathcal{U})$ if and only if it is a solution to* (3.35).

Proof. Suppose $x \in \mathbb{N}^{|I|}$ is feasible for $\mathrm{P_s}(\mathcal{U})$ and let a subset $I' \subseteq I$ be arbitrary. As $N(S') \subseteq I'$, we have that

$$\sum_{i \in I'} q \cdot x_i \geq \sum_{i \in N(S')} q \cdot x_i \geq \max_{\xi \in \mathcal{U}} \sum_{j \in S'} \xi_j.$$

Therefore, x is feasible for (3.35). Now, let $x \in \mathbb{N}^{|I|}$ be feasible for (3.35) and let $S \subseteq J$ be arbitrary. Consider the set $I' := N(S)$. Note that $S \subseteq S'$ and we get

$$\sum_{i \in N(S)} q \cdot x_i \geq \max_{\xi \in \mathcal{U}} \sum_{j \in S'} \xi_j \geq \max_{\xi \in \mathcal{U}} \sum_{j \in S} \xi_j.$$

Therefore, x is feasible for $\mathrm{P_s}(\mathcal{U})$. □

Note that (3.35) has a fixed number of variables as well as a fixed number of constraints if $|I| \leq k$ for some fixed $k \in \mathbb{N}$. Thus, if, for all $I' \subseteq I$, the right-hand side of (3.35b) can be computed in polynomial time, the formulation (3.35) is polynomial time solvable for any fixed $q \in \mathbb{N}_{>0}$ [Len83]. For example, this is the case with discrete, interval, budgeted, and Γ-uncertainty. Yet, for the general case, we obtain NP-hardness due to the flexibility of the uncertainty set $\mathcal{U}$.

Theorem 3.83. *For any fixed $q \in \mathbb{N}_{>0}$, Robust q-MSMC is* NP*-hard even if $|I| = |J| = 1$.*

Proof. We consider the NP-complete problem Quadratic Congruences. In an instance of this problem, we have given positive integers a, b, and c and the question is whether there is a positive integer x with $x < c$ and $x^2 \equiv a \mod b$, cf. [GJ79].

We construct an instance of Robust q-MSMC by defining a bipartite graph G that consists of the edge $[1, 2]$ with $I := \{1\}$ and $J := \{2\}$. Furthermore, we set $B := 1$ and

$$\mathcal{U} := \left\{1 + \xi \colon \xi \in \mathbb{N}_{>0}, \xi < c, \xi^2 \equiv a \mod b\right\} \cup \{1\},$$

i.e., the uncertainty set $\mathcal{U}$ contains the solutions to the instance of Quadratic Congruences increased by 1 and the value 1.

Now, suppose we have an oracle that returns the answer to the constructed instance of Robust q-MSMC. If the answer is "yes", we know that $x := 1$ is feasible for the constructed instance and hence $\left\{\xi \in \mathbb{N}_{>0} \colon \xi < c, \xi^2 \equiv a \mod b\right\}$ must be empty, i.e., the instance of Quadratic Congruences is a no-instance. If, conversely, the answer of the oracle is "no", we know that

$$\left\{\xi \in \mathbb{N}_{>0} \colon \xi < c, \xi^2 \equiv a \mod b\right\} \neq \varnothing$$

so that the instance of Quadratic Congruences is a yes-instance. □

The above proof heavily relies on the property that the uncertainty set does not need to belong to a specific class. It can be generalized as follows.

Corollary 3.84. *Let $\mathcal{C} \subseteq 2^{\mathbb{R}_{>0}}$ be a class of uncertainty sets such that, given $\mathcal{U} \in \mathcal{C}$, asking whether $\mathcal{U} \neq \varnothing$ is an* NP*-complete problem. Then, for any fixed $q \in \mathbb{N}_{>0}$, Robust q-MSMC with uncertainty $\mathcal{C}$ is* NP*-hard even in the case of $|I| = |J| = 1$.* □

Observe that Theorem 3.83 also covers the case of bounded number of regions. Nevertheless, we aim to establish the result of polynomial time solvability for certain classes of uncertainty sets. If $|J| \leq k$ for some fixed $k \in \mathbb{N}$, $\mathrm{P_s}(\mathcal{U})$ contains at most 2^k many constraints and polynomially many variables. Furthermore, without loss of generality, we can assume that $N(i) \neq N(i')$ for two distinct locations $i, i' \in I$. Hence, there are at most 2^k many distinct locations and the number of variables is upper bounded by this value. Thus, we get that, for any fixed $q, k \in \mathbb{N}_{>0}$, Robust q-MSMC restricted to instances with $|J| \leq k$ and for which $\mathcal{U}$ has a polynomial time optimization oracle can be solved in polynomial time.

Conclusion

In this chapter, we studied the problem Robust Min q-MSMC for general uncertainty sets and also for various specific classes of uncertainty sets. We presented two equivalent formulations of the problem and used them to analyze solution techniques, in particular, constraint generation. Further, we derived numerous results concerning the complexity of Robust Min q-MSMC. If a set of instances of Robust Min q-MSMC formed an APX-hard problem, the analysis of the corresponding Separation and Feasibility problems was conducted. Here, NP-completeness of the Separation problem implies co-NP-completeness of the corresponding Feasibility problem and NP-hardness of the corresponding LP-relaxation of Robust Min q-MSMC. Notably, even for uncertainty sets for which Robust Sum is polynomial time solvable, the corresponding Separation problem is NP-complete in some cases. Moreover, if Robust Sum is polynomial time solvable, it is possible to compute violating extreme scenarios from violating subsets in polynomial time and vice versa. Table 3.2 summarizes our main results.

Uncertainty	Robust Min q-MSMC	Separation/Feasibility	Robust Sum
General	APX-hard	NP/co-NP-complete	NP-complete
Discrete			
$q \cdot \lvert\mathcal{U}\rvert \leq 2$	PO	P	P
$q \cdot \lvert\mathcal{U}\rvert \geq 3$	APX-complete	P	P
Interval			
$q \leq 2$	PO	P	P
$q \geq 3$	APX-complete	P	P
Budgeted	APX-hard	NP/co-NP-complete	P
Multi-budgeted	APX-hard	NP/co-NP-complete	NP-complete
Ellipsoidal	APX-hard	NP/co-NP-complete	NP-complete
Γ	APX-hard	NP/co-NP-complete	P

Table 3.2.: Summary of results for Robust Min q-MSMC and related problems. If not stated otherwise, the results hold for any fixed $q \in \mathbb{N}_{>0}$.

4. Approximating Robust Min q-Multiset Multicover

In the previous chapter, we have seen that almost all considered variants of Robust Min q-MSMC are APX-hard. Hence, the existence of polynomial time solution algorithms for these variants is rather unlikely. This motivates to aim for approximate solutions that can be computed with less effort. These solutions are helpful as they serve as upper bounds and start solutions for the constraint generation process. Since Robust Min q-MSMC admits for an interpretation as an adjustable robust problem and as a (robust) Multiset Multicover problem, we differentiate our considerations in Section 4.1 and 4.2. Our aim in this chapter is to present an approximation algorithm that is simultaneously applicable for various classes of uncertainty sets and also for the non-robust version. Hence, we intend to impose only mild assumptions on the given uncertainty sets.

Observe that, when talking about approximation algorithms for Robust Min q-MSMC, we want to compute a solution x to a given instance in polynomial time. By Theorem 3.8, given x and a scenario $\xi \in \mathcal{U}$, we can compute the corresponding adjustable variables $y(\xi)$ in polynomial time so that we stay in line with Definition 2.9. Furthermore, in this chapter, we again restrict our considerations to feasible instances of Robust Min q-MSMC. By our assumptions on $\mathcal{U}$ in Section 2.6, such an instance can be identified in time $\mathcal{O}(|J|)$, where J denotes the set of regions of the given instance.

To illustrate our results and to gain more insights, we additionally focus on a distinct class of uncertainty sets. We choose budgeted uncertainty, i.e., $\mathcal{U}_{\mathrm{B}} = \left\{\xi \in \mathbb{N}^{|J|} \colon a \leq \xi \leq b, \xi(J) \leq \Gamma\right\}$ with $a, b \in \mathbb{N}^{|J|}$, $a \leq b$, and $\Gamma \in \mathbb{N}$, cf. Section 3.4.3 and Assumption 3.60.

In Section 4.1, we start with an extensive literature review on approximations and approximation algorithms for the Min Adjustable Robust Covering problem which is introduced in Section 2.6.1. The results obtained therein are applied and extended to Robust Min q-MSMC in Section 4.1.2.

Further, Section 4.2 begins with reviewing relevant literature on approximation algorithms for the Min Set Cover problem and its generalizations, cf. Section 2.4. We concentrate on one particular algorithm for Min Set Cover in Section 4.2.2 and utilize this algorithm as our starting point to develop an approximation for Robust Min q-MSMC in Section 4.2.3.

4.1. Approximation based on Adjustable Robustness

Given an instance of Robust Min q-MSMC, we have seen in Section 3.2 that it corresponds to the mixed integer program $\mathrm{P}(\mathcal{U})$, cf. Page 51. In this section, we concentrate on the relation of $\mathrm{P}(\mathcal{U})$ to the Min Adjustable Robust Covering problem (Min ARC) introduced in Section 2.6.1. Of course, this relation is likewise valid for $\mathrm{P}_{\mathrm{s}}(\mathcal{U})$. To reduce recurrences we stay with formulation $\mathrm{P}(\mathcal{U})$ in this section.

4.1.1. Literature Review

Recall that an instance of Min ARC of Section 2.6.1 leads to solving the following MIP:

$$
\begin{aligned}
\mathrm{P}_{\mathrm{arc}}(\mathcal{Z}) \qquad \min_{x,\, y(\zeta)} \quad & c^T x + \max_{\zeta \in \mathcal{Z}} d^T y(\zeta) && && \text{(4.1a)}\\
\text{s.t.} \quad & Ax + By(\zeta) \geq \zeta && \text{for } \zeta \in \mathcal{Z} && \text{(4.1b)}\\
& x, y(\zeta) \geq 0 && \text{for } \zeta \in \mathcal{Z} && \text{(4.1c)}\\
& x \in \mathbb{R}^{n_1 - p_1} \times \mathbb{N}^{p_1} && && \text{(4.1d)}\\
& y(\zeta) \in \mathbb{R}^{n_2 - p_2} \times \mathbb{N}^{p_2} && \text{for } \zeta \in \mathcal{Z}, && \text{(4.1e)}
\end{aligned}
$$

with $A \in \mathbb{Q}^{m \times n_1}$, $B \in \mathbb{Q}^{m \times n_2}$, $c \in \mathbb{Q}^{n_1}_{\geq 0}$, $d \in \mathbb{Q}^{n_2}_{\geq 0}$, $p_1 \leq n_1$, $p_2 \leq n_2$, and an uncertainty set $\mathcal{Z} \subseteq \mathbb{R}^m_{\geq 0}$. If $p_2 = 0$, we can additionally assume that $\mathcal{Z}$ is convex, cf. Lemma 2.10. Let $z_{\mathrm{arc}}(\mathcal{Z})$ denote the optimal value of $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$ which we assume exists.

The optimization problem Min ARC together with formulation $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$ is extensively studied in the literature. Here, we concentrate on results concerning complexity and approximation algorithms that lead to provable approximation ratios. For a general overview on robust multi-stage optimization problems, we refer the reader to [DI15]. It can readily be seen that Min ARC is strongly NP-hard as it captures the famous Min Set Cover problem, compare Section 2.4. Thus, for practical purposes, we seek approximate solutions that are close to an optimal solution in terms of their corresponding solution values. But recall that Min ARC cannot be approximated in polynomial time with ratio better than $\ln m$ unless $\mathrm{P} = \mathrm{NP}$ [DS14; Mos15].

In literature, a wide range of *solution policies*, where the adjustable variables $y(\zeta)$ are assumed to have a specified structure, is used to obtain approximations as we see in the following. We say that a policy is *optimal* if it can be imposed on $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$ without deteriorating its optimal value.

In [Ben+04], Ben-Tal et al. define the adjustable robust counterpart of an uncertain linear programming problem as a relaxation of its robust counterpart, cf. Section 2.6.1. They show that the LP-relaxation of $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$ can be rewritten as a linear program if $\mathcal{Z}$ is given as the convex hull of a finite set. Thus, in this special LP-case the problem is polynomial time solvable. Furthermore, they consider the *strictly robust/static policy* where one single variable vector y replaces all variable vectors $y(\zeta)$, i.e., adjustability is discarded and the problem becomes a single-stage optimization problem. If $\mathcal{Z}$ is a constraint-wise[1] uncertainty set, they prove optimality of the strictly robust policy. Moreover, they introduce a generalization of this policy which is later called *affine policy*: To approximate the adjustable robust counterpart, one assumes that $y(\zeta)$ affinely depends on the uncertain data ζ, i.e., $y(\zeta) = W\zeta + w$ for a matrix W and a vector w of appropriate dimensions. Applying this transformation, problem $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$ turns into a problem with variables x, W, and w. With this assumption and a computationally tractable[2] uncertainty set $\mathcal{Z}$, they show that the LP-relaxation of $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$ is computationally tractable as well. There are already several results on the strong performance of the affine policy in practice, see e.g., [Ben+04; Ben+05].

In [BG10], Bertsimas and Goyal introduce $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$ as a *two-stage adaptive optimization problem*. They define and investigate the *adaptivity gap*, i.e., the ratio between the optimal value $z_{\mathrm{rob}}(\mathcal{Z})$ of the strictly robust policy and the optimal value $z_{\mathrm{arc}}(\mathcal{Z})$ of $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$. In doing so, they prove upper bounds for this gap for several classes of uncertainty sets including *symmetric* and

[1] $\mathcal{Z}$ is a direct product of m uncertainty sets, compare [Ben+04; BGN09].
[2] The separation problem for $\mathcal{Z}$ is polynomial time solvable, cf. Page 26 and [Ben+04].

positive sets, cf. Section 4.1.2.

The notion of the strictly robust policy is extended by Bertsimas and Caramanis in [BC10] by defining a *piecewise static/finitely adaptable* policy. Here, the idea is to partition the uncertainty set into finitely many pieces where each piece $\mathcal{U}_i$ obtains one adjustable variable vector $y(\mathcal{U}_i)$. However, they show that finding the optimal pieces is NP-hard. This policy is similarly applied in [BGS11]. Therein, a finitely adaptable policy is provided yielding approximate solutions to $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$ with ratio depending only on the geometric properties of the uncertainty set $\mathcal{Z}$. But for a two-stage problem, this policy coincides with the strictly robust policy. Subsequently, the geometric results are extended by Bertsimas and Bidkhori in [BB14] who investigate the relation of the optimal value $z_{\mathrm{aff}}(\mathcal{Z})$ of the affine policy to $z_{\mathrm{arc}}(\mathcal{Z})$.

In [BG11], a wide range of further approximation results for $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$ with focus on the affine policy is given by Bertsimas and Goyal. They show that this policy is optimal for simplex uncertainty sets. In particular, this shows that a *piecewise affine policy*, where the uncertainty set is partitioned and one applies the affine policy on each piece, is optimal for polytope uncertainty sets [BGE19]. Note that piecewise affine policies generalize piecewise static policies. Moreover, if the constraint matrix A only contains non-negative data, the optimal value of the affine policy approximates $z_{\mathrm{arc}}(\mathcal{Z})$ within a factor of $\mathcal{O}(\sqrt{m})$ [BG11]. For the general case, Bertsimas and Goyal also give a (non-affine) solution with ratio $\mathcal{O}(\sqrt{m})$ by considering a simplex uncertainty set containing $m+1$ many scenarios and dominating the uncertainty set $\mathcal{Z}$, cf. [BG11]. In [BGE19], similar ideas are used to obtain an $\mathcal{O}(\sqrt{m})$-approximation for the LP-relaxation of $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$. Therein, the ratio heavily relies on the fact that one can assume $\mathcal{Z} \subseteq [0,1]^m$ for the LP-case. Further extensions and improvements of the affine policy for the LP-case are given in [CZ09] and [XB17] but no stronger approximation ratios are presented.

In [Lu16], the idea of a dominating uncertainty set from [BG11] for the LP-relaxation of $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$ is reconsidered. Therein, Lu uses a scaled dominating uncertainty set and presents specific classes of uncertainty sets for which this approach provides better bounds than recently known.

In the following, we investigate the possible implications of the above findings to obtain approximations of Robust Min q-MSMC. In particular, we need to take care of the integrality constraints on x, the integrality of the scenarios, and the complexity of the resulting problems.

4.1.2. Application to Robust Min q-Multiset Multicover

In this section, we analyze the relation between Min ARC and Robust Min q-MSMC and how this link leads to approximation algorithms for special cases of Robust Min q-MSMC. In particular, we focus on budgeted uncertainty. First of all, we establish the following result.

Lemma 4.1. *For any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-MSMC is a special case of Min ARC.*

Proof. Let an instance $\mathcal{I}$ of Robust Min q-MSMC be given with $n := |I|$ and $m := |J|$. Consider the formulation $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$ on Page 108. Then, we set $c := 1$, $d := 0$, and

$$\mathcal{Z} := \left\{(\xi, 0)^T \in \mathbb{N}^{m+n} \colon \xi \in \mathcal{U}\right\} \subseteq \mathbb{N}^{m+n}.$$

As $\mathcal{U}$ is non-empty, compact, and finite, these properties also hold for $\mathcal{Z}$. Note that we only have m uncertain parameters as the last n entries of every scenario $\zeta \in \mathcal{Z}$ are fixed to zero. The matrices $A \in \mathbb{N}^{(m+n)\times n}$ and $B \in \mathbb{Z}^{(m+n)\times(nm)}$ have the following structure:

$$A := \begin{pmatrix} 0 \\ q \cdot I_n \end{pmatrix} \quad \text{and} \quad B := \begin{pmatrix} B_{11} & B_{12} & \cdots & B_{1n} \\ B_{21} & B_{22} & \cdots & B_{2n} \end{pmatrix}$$

where $B_{1i} \in \mathbb{B}^{m\times m}$ and $B_{2i} \in \{-1, 0\}^{n\times m}$ for $i \in \{1, \ldots, n\}$. Finally, for $i \in \{1, \ldots, n\}$, the matrices B_{1i} and B_{2i} have zero entries except for the following ones: $(B_{1i})_{jj} := 1$ if $i \in N(j)$ and $(B_{2i})_{ij} := -1$ if $j \in N(i)$. Note that B_{1i} only has non-zero entries on the diagonal and B_{2i} only has non-zero entries in row i. Last but not least, we require $x \in \mathbb{N}^n$ and $y(\zeta) \in \mathbb{R}^{nm}$ for $\zeta \in \mathcal{Z}$ so that $p_1 := n$ and $p_2 := 0$. By our identification of $\mathbb{R}^{nm}$ with $\mathbb{R}^{n\times m}$, cf. Section 2.1, the constructed instance of Min ARC corresponds to $\mathcal{I}$ as the formulations $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$ and $\mathrm{P}(\mathcal{U})$ coincide. Hence, Robust Min q-MSMC is a special case of Min ARC. □

Using the terminology of stochastic programming and robust optimization the matrix B is called *recourse matrix* and we see that B is a *fixed* recourse as it is not affected by uncertainty [Pré95; Ben+04]. In the following, we analyze approximation results for Robust Min q-MSMC, especially with budgeted uncertainty, based on Section 4.1.1. The findings can be assigned to three different groups: We investigate

(a) the performance of applying strict robustness,

(b) the performance of imposing the affine policy, and

(c) the performance of dominating uncertainty sets.

Performance of Strict Robustness

In this section, we concentrate on the performance of the strictly robust policy applied to Robust Min q-MSMC. We see certain geometric properties of the uncertainty set $\mathcal{U}$ that allow to calculate approximation ratios. To obtain an intuition of these ratios, we analyze the geometric characteristics for budgeted uncertainty sets. Thus, let us apply strict robustness to Robust Min q-MSMC and consider

$$
\begin{array}{llll}
\mathrm{P}_{\mathrm{rob}}(\mathcal{U}) & \min\limits_{x,y} & \sum\limits_{i\in I} x_i & \\
& \text{s.t.} & \sum\limits_{i\in N(j)} y_{ij} \geq \xi_j & \text{for } j \in J, \xi \in \mathcal{U} \\
& & \sum\limits_{j\in N(i)} y_{ij} \leq q \cdot x_i & \text{for } i \in I \\
& & y_{ij} \geq 0 & \text{for } i \in I, j \in J \\
& & x_i \in \mathbb{N} & \text{for } i \in I.
\end{array}
$$

It can readily be seen that $\mathrm{P}_{\mathrm{rob}}(\mathcal{U})$ is feasible. Furthermore, an optimal solution to $\mathrm{P}_{\mathrm{rob}}(\mathcal{U})$ corresponds to an optimal solution to the corresponding instance of Min q-MSMC with the worst-case demand vector ξ^{wc}, where $\xi_j^{\mathrm{wc}} := \max_{\xi\in\mathcal{U}} \xi_j$ for $j \in J$. Let $(x^\star, y^\star)$ be optimal for $\mathrm{P}_{\mathrm{rob}}(\mathcal{U})$. We call $x^\star$ an *optimal strictly robust solution* for $\mathcal{I}$. It is easy to see that a solution (x, y) to $\mathrm{P}_{\mathrm{rob}}(\mathcal{U})$ gives a solution x to $\mathcal{I}$ showing that $x^\star$ is indeed feasible for $\mathcal{I}$. Denote by $z_{\mathrm{rob}}(\mathcal{U})$ the optimal value of $\mathrm{P}_{\mathrm{rob}}(\mathcal{U})$ and by $z(\mathcal{U})$ the analogue of $\mathrm{P}(\mathcal{U})$. Hence, we get that $z(\mathcal{U}) \leq z_{\mathrm{rob}}(\mathcal{U})$. The adaptivity gap, cf. [BG10],

$$\frac{z_{\mathrm{rob}}(\mathcal{U})}{z(\mathcal{U})}$$

is bounded by

$$\sum_{j\in J} \left\lceil \frac{\xi_j^{\mathrm{wc}}}{q} \right\rceil$$

and our aim is to improve this bound. For the cases in which Min q-MSMC is polynomial time solvable, e.g., $q \in \{1,2\}$, and ξ^{wc} can be computed in polynomial time, we then obtain approximation algorithms for Robust Min q-MSMC. Observe that the consideration of the adaptivity gap is only meaningful if $|\mathcal{U}| \geq 2$ as otherwise the optimal values coincide. Thus, for the remainder of this section, we assume that the uncertainty set $\mathcal{U}$ of our instance of Robust Min q-MSMC contains at least two scenarios. All presented ratios rely on the following fact which is similarly proven in [BG10; BGS11].

Lemma 4.2. *Let an instance $\mathcal{I}$ of Robust Min q-MSMC be given. If there is $\xi' \in \operatorname{conv}(\mathcal{U})$ and $\alpha \in \mathbb{R}$ such that $\alpha \cdot \xi' \geq \xi$ for all $\xi \in \mathcal{U}$, then an optimal strictly robust solution is an $\lceil \alpha \rceil$-approximate solution to $\mathcal{I}$.*

Proof. Let $(x^\star, y^\star)$ be an optimal solution to $\mathrm{P}(\operatorname{conv}(\mathcal{U}))$ and let α and ξ' be given as above. Consider the vector (x,y) with $x \coloneqq \lceil \alpha \rceil \cdot x^\star$ and $y \coloneqq \alpha \cdot y^\star(\xi')$. We show that (x,y) is feasible for $\mathrm{P}_{\mathrm{rob}}(\mathcal{U})$. For $j \in J$ and $\xi \in \mathcal{U}$, we have

$$\sum_{i \in N(j)} \alpha \cdot y^\star(\xi')_{ij} \geq \alpha \cdot \xi'_j \geq \xi_j.$$

Furthermore, for $i \in I$, it holds that

$$\sum_{j \in N(i)} \alpha \cdot y^\star(\xi')_{ij} \leq q \cdot \lceil \alpha \rceil \cdot x^\star_i = q \cdot x_i$$

by feasibility of $x^\star$ and $y^\star(\xi')$ for $\mathrm{P}(\operatorname{conv}(\mathcal{U}))$. Thus, the vector (x,y) is feasible for $\mathrm{P}_{\mathrm{rob}}(\mathcal{U})$ and we obtain by Lemma 2.10

$$z_{\mathrm{rob}}(\mathcal{U}) \leq \lceil \alpha \rceil \cdot x^\star(I) = \lceil \alpha \rceil \cdot z(\operatorname{conv}(\mathcal{U})) = \lceil \alpha \rceil \cdot z(\mathcal{U}). \qquad \square$$

To take advantage of Lemma 4.2 we define some properties of subsets of $\mathbb{R}^p$ where $p \in \mathbb{N}_{>0}$. In particular, we focus on convex bodies, i.e., compact and convex sets having a non-empty relative interior.

Definition 4.3 (Symmetric Set [BF06; BG10])**.** A set $P \subseteq \mathbb{R}^p$ is *symmetric* if there exists some $u^0 \in P$ such that, for any $z \in \mathbb{R}^p$, $u^0 + z \in P$ if and only if $u^0 - z \in P$. The point u^0 is called a *point of symmetry* of P.

In fact, by [BG10; BF06], the point of symmetry u^0 of a symmetric convex body $P \subseteq \mathbb{R}^p$ is uniquely given by $u^0 \coloneqq 1/2 \cdot (x+y)$ with $x_j \coloneqq \min_{x' \in P} x'_j$

and $y_j := \max_{x' \in P} x'_j$ for $j \in \{1, \ldots, p\}$. Hence, the subsequent property is well-defined.

Definition 4.4 (Positive Set [BG10])**.** A convex body $P \subseteq \mathbb{R}^p_{\geq 0}$ is *positive* if there exists a symmetric convex body $S \subseteq \mathbb{R}^p_{\geq 0}$ such that $P \subseteq S$ and the point of symmetry of S is contained in P.

As in the proof of Lemma 4.2 we can consider the uncertainty set $\mathcal{U}^+ := \operatorname{conv}(\mathcal{U})$ instead of $\mathcal{U}$. Hence, by our assumptions on $\mathcal{U}$, we get that $\operatorname{conv}(\mathcal{U})$ is a convex body and the results of [BG10] are applicable to our problem. In [BG10], Bertsimas and Goyal show that the adaptivity gap equals 1 if the uncertainty set $\mathcal{U}^+$ is a hypercube, see also Section 3.4.2. Moreover, they prove that, if $\mathcal{U}^+$ is symmetric or positive with point of symmetry ξ^0, then $2 \cdot \xi^0 \geq \xi$ for all $\xi \in \mathcal{U}^+$. By Lemma 4.2 the adaptivity gap is therefore bounded from above by 2 for these classes of uncertainty sets. Thus, if we additionally have $q = 1$ or $q = 2$ and the vector ξ^{wc} can be computed in polynomial time, we obtain a 2-approximation algorithm for Robust Min q-MSMC.

Example 4.5 (Budgeted Uncertainty)**.** For an instance of Robust Min q-MSMC with budgeted uncertainty $\mathcal{U}_{\mathrm{B}} \subseteq \mathbb{N}^{|J|}$, we get that the adaptivity gap is bounded from above by $|J|$: Let $\mathcal{U}_{\mathrm{B}} = \left\{\xi \in \mathbb{N}^{|J|} \colon a \leq \xi \leq b, \xi(J) \leq \Gamma\right\}$ as on Page 107. Then, it holds true that

$$z_{\mathrm{rob}}(\mathcal{U}_{\mathrm{B}}) \leq \sum_{j \in J} \left\lceil \frac{b_j}{q} \right\rceil \leq \sum_{j \in J} \left\lceil \frac{\Gamma}{q} \right\rceil = |J| \cdot \left\lceil \frac{\Gamma}{q} \right\rceil \leq |J| \cdot z(\mathcal{U}_{\mathrm{B}}).$$

The convex set $\mathcal{U}_{\mathrm{B}}^+ := \operatorname{conv}(\mathcal{U}_{\mathrm{B}})$ is a hypercube if $b(J) = \Gamma$ implying that an optimal strictly robust solution $x^\star$ leads to an optimal solution $(x^\star, y^\star)$ for $\mathrm{P}(\mathcal{U}_{\mathrm{B}})$.

For a general consideration, suppose $\mathcal{U}_{\mathrm{B}}^+$ is a convex body. We show that $\mathcal{U}_{\mathrm{B}}^+$ is positive if and only if there are vectors $\delta^1, \delta^2 \geq 0$ such that the following constraints are fulfilled:

$$\begin{aligned} a - \delta^1 &\geq 0 \\ a + \delta^2 - \delta^1 &\leq b \\ b + \delta^2 - \delta^1 &\geq a \\ b(J) + a(J) + \delta^2(J) - \delta^1(J) &\leq 2\Gamma. \end{aligned} \tag{4.2}$$

If $\mathcal{U}_{\mathrm{B}}^{+}$ is positive, let $S \subseteq \mathbb{R}_{\geq 0}^{|J|}$ be a symmetric convex body with $\mathcal{U}_{\mathrm{B}}^{+} \subseteq S$ such that the point of symmetry u^0 of S is contained in $\mathcal{U}_{\mathrm{B}}^{+}$. Define $x, y \in \mathbb{R}^{|J|}$ with $x_j := \min_{x' \in S} x'_j \leq a_j$ and $y_j := \max_{x' \in S} x'_j \geq b_j$ for $j \in J$. Then, by [BG10], we get $u^0 = 1/2 \cdot (x + y)$. With $\delta^1 := a - x$ and $\delta^2 := y - b$ the claim follows as $u^0 \in \mathcal{U}_{\mathrm{B}}^{+}$. On the other hand, suppose there are $\delta^1, \delta^2 \geq 0$ fulfilling (4.2). Then, the hypercube $S := \left\{x \colon a - \delta^1 \leq x \leq b + \delta^2\right\} \subseteq \mathbb{R}_{\geq 0}^{|J|}$ is a symmetric convex body with point of symmetry

$$u^0 := \frac{a - \delta^1 + b + \delta^2}{2}$$

and we get $u^0 \in \mathcal{U}_{\mathrm{B}}^{+}$ as well as $\mathcal{U}_{\mathrm{B}}^{+} \subseteq S$.

Hence, if $\mathcal{U}_{\mathrm{B}}^{+}$ is positive, an optimal strictly robust solution gives a 2-approximation. In particular, this holds if $1/2 \cdot (a + b) \in \mathcal{U}_{\mathrm{B}}^{+}$. ◁

In addition to the above results, in [BGS11], it is shown that $z_{\mathrm{rob}}(\mathcal{U})$ approximates the optimal value $z(\mathcal{U})$ by a factor of

$$\left\lceil 1 + \frac{\rho(\mathcal{U}^{+})}{\operatorname{sym}(\mathcal{U}^{+})} \right\rceil, \tag{4.3}$$

where $\rho(\mathcal{U}^{+})$ is the *translation factor* of $\mathcal{U}^{+}$ and $\operatorname{sym}(\mathcal{U}^{+})$ is the *symmetry factor* of $\mathcal{U}^{+}$, which we subsequently define. The symmetry factor generalizes Definition 4.3 for convex bodies:

Definition 4.6 (Symmetry Factor [BGS11])**.** For a convex body $P \subseteq \mathbb{R}^p$ and some $u \in P$, the *symmetry* of P with respect to u is defined as

$$\operatorname{sym}(u, P) := \max\left\{\alpha \geq 0 \colon u + \alpha \cdot (u - u') \in P \text{ for } u' \in P\right\}. \tag{4.4}$$

Then, the *symmetry factor* of P is defined as

$$\operatorname{sym}(P) := \max\left\{\operatorname{sym}(u, P) \colon u \in P\right\} \tag{4.5}$$

and an optimizer u^0 of (4.5) is called *point of symmetry* of P.

Hence, the value $\operatorname{sym}(u, P)$ describes the maximum possible factor $\alpha \geq 0$ such that any point $u' \in P$ can be reflected through u by the factor α and the reflection is still contained in P. For a point of symmetry u^0 of P, we have $u^0 \in P^{\mathrm{ri}}$ by (4.4). We note that no ambiguities can occur when speaking of a point of symmetry of a given set. Let P be as in Definition 4.6. If P is symmetric with unique point of symmetry u^0 as given in Definition 4.3, we have that, for every $u \in \mathbb{R}^p$, $u \in P$ if and only if $u^0 + (u^0 - u) \in P$. Compactness of P and $P^{\mathrm{ri}} \neq \varnothing$ give $\mathrm{sym}(u, P) \leq 1$ for every $u \in P$. Hence, $\mathrm{sym}(u^0, P) = 1$ and $\mathrm{sym}(P) = 1$ so that u^0 maximizes (4.5).

Definition 4.7 (Translation Factor [BGS11])**.** For a convex body $P \subseteq \mathbb{R}^p_{\geq 0}$ and some $u \in P$, the *translation* of P with respect to u is defined as

$$\rho(u, P) \coloneqq \min \{\alpha \geq 0 \colon u' - (1 - \alpha) \cdot u \geq 0 \text{ for } u' \in P\}.$$

Then, the *translation factor* of P is defined as

$$\rho(P) \coloneqq \min \left\{\rho(u^0, P) \colon u^0 \in \arg\max \{\mathrm{sym}(u, P) \colon u \in P\}\right\}. \tag{4.6}$$

Thus, for some $u \in P$, the set $P' \coloneqq P - (1 - \rho(u, P)) \cdot u$ is the maximum possible translation of P into direction $-u$ such that $P' \subseteq \mathbb{R}^p_{\geq 0}$ still holds. As any point of symmetry of P is contained in P^{ri}, we see that $0 < \rho(P) \leq 1$. The following lemma provides the foundation for the approximation factor (4.3).

Lemma 4.8 ([BGS11])**.** *Let $\mathcal{U} \subseteq \mathbb{N}^p$ be an uncertainty set with $|\mathcal{U}| \geq 2$. Let $\mathcal{U}^+ \coloneqq \mathrm{conv}(\mathcal{U})$ with a point of symmetry ξ^0. Then,*

$$\left(1 + \frac{\rho(\xi^0, \mathcal{U}^+)}{\mathrm{sym}(\mathcal{U}^+)}\right) \cdot \xi^0 \geq \xi$$

for all $\xi \in \mathcal{U}^+$.

Proof. By the properties of $\mathcal{U} \subseteq \mathbb{N}^p$ we get that $\mathcal{U}^+ \subseteq \mathbb{R}^p_{\geq 0}$ is a convex body. Let $\mathcal{U}' \coloneqq \mathcal{U}^+ - (1 - \rho(\xi^0, \mathcal{U}^+)) \cdot \xi^0 \subseteq \mathbb{R}^p_{\geq 0}$. Then, $\mathrm{sym}(\mathcal{U}') = \mathrm{sym}(\mathcal{U}^+)$ by Definition 4.6 and a point of symmetry of $\mathcal{U}'$ is given by $\xi^1 \coloneqq \rho(\xi^0, \mathcal{U}^+) \cdot \xi^0$. Let $\xi \in \mathcal{U}^+$ and $\xi' \coloneqq \xi - (1 - \rho(\xi^0, \mathcal{U}^+)) \cdot \xi^0$. By Definition 4.6 we get

$$\xi^1 + \mathrm{sym}(\mathcal{U}^+) \cdot (\xi^1 - \xi') \geq 0 \quad \Leftrightarrow \quad \left(1 + \frac{1}{\mathrm{sym}(\mathcal{U}^+)}\right) \cdot \xi^1 \geq \xi'.$$

Inserting the definitions for ξ^1 and ξ' gives

$$\left(1 + \frac{\rho(\xi^0, \mathcal{U}^+)}{\mathrm{sym}(\mathcal{U}^+)}\right) \cdot \xi^0 \geq \xi. \qquad \square$$

To obtain the approximation factor given in (4.3) we apply Lemma 4.2 and Lemma 4.8 with a point of symmetry that minimizes (4.6). If such a minimizer cannot be found in practice, Lemma 4.8 still applies to some arbitrary point of symmetry ξ^0 and we get a bound of

$$\left\lceil 1 + \frac{\rho(\xi^0, \mathcal{U}^+)}{\mathrm{sym}(\mathcal{U}^+)} \right\rceil$$

for the adaptivity gap.

As the translation factor of a set is bounded by 1 and $\dim(\mathcal{U}^+) \cdot \mathrm{sym}(\mathcal{U}^+) \geq 1$ by [BF06] we have an upper bound of $1 + \dim(\mathcal{U}^+)$ for (4.3). Note that, if $\mathcal{U}^+$ is symmetric, we reobtain the bound 2 as $\rho(\mathcal{U}^+) > 0$.

In the following, we analyze these geometric factors for an instance of Robust Min q-MSMC with budgeted uncertainty $\mathcal{U}_\mathrm{B} \subseteq \mathbb{N}^{|J|}$. In case $a(J) = \Gamma$ we have $\mathcal{U}_\mathrm{B} = \{a\}$ and $z_\mathrm{rob}(\mathcal{U}_\mathrm{B}) = z(\mathcal{U}_\mathrm{B})$. Thus, for the remainder of this section, we assume that $a(J) < \Gamma \leq b(J)$ so that $\mathcal{U}_\mathrm{B}$ fulfills the requirements of Lemma 4.8.

Lemma 4.9. *Let $\mathcal{U}_B^+$ be the convex hull of a budgeted uncertainty set $\mathcal{U}_B \subseteq \mathbb{N}^{|J|}$ with $a(J) < \Gamma$. The (unique) point of symmetry ξ^0 of $\mathcal{U}_B^+$, the symmetry factor* $\mathrm{sym}(\mathcal{U}_B^+)$*, and the translation factor $\rho(\mathcal{U}_B^+)$ are given by the following formulas:*

$$\begin{aligned}
\xi^0 &= a + \lambda \cdot (b - a), \\
\mathrm{sym}(\mathcal{U}_B^+) &= \frac{\Gamma - a(J)}{b(J) - a(J)}, \\
\rho(\mathcal{U}_B^+) &= \max_{j \in J} \frac{\lambda \cdot (b_j - a_j)}{a_j + \lambda \cdot (b_j - a_j)},
\end{aligned}$$

with

$$\lambda := \frac{\Gamma - a(J)}{\Gamma + b(J) - 2a(J)}.$$

Proof. First consider the scalar $\lambda \in \mathbb{R}_{>0}$. As $\Gamma \leq b(J)$, we get that $\lambda \leq 1/2$. Furthermore, by [BF06], the symmetry factor and a point of symmetry of $\mathcal{U}_\mathrm{B}^+$ are obtained from an optimal solution to the following quadratic program:

$$\max_{s,\xi} \quad s \tag{4.7a}$$

$$\text{s.t.} \quad (\xi_j - a_j) \cdot s \leq b_j - \xi_j \qquad \text{for } j \in J \tag{4.7b}$$

$$(b_j - \xi_j) \cdot s \leq \xi_j - a_j \qquad \text{for } j \in J \tag{4.7c}$$

$$(\xi(J) - a(J)) \cdot s \leq \Gamma - \xi(J) \tag{4.7d}$$

$$\xi \in \mathcal{U}_{\mathrm{B}}^{+}. \tag{4.7e}$$

We argue that

$$s^\star := \frac{\Gamma - a(J)}{b(J) - a(J)} \quad \text{and} \quad \xi^\star := a + \lambda \cdot (b - a)$$

is an optimal variable assignment for (4.7). First of all, note that

$$1 - \lambda = 1 - \frac{\Gamma - a(J)}{\Gamma + b(J) - 2a(J)} = \frac{b(J) - a(J)}{\Gamma + b(J) - 2a(J)} > 0$$

so that $s^\star = \lambda/1-\lambda$. As $\lambda \leq 1/2$, we have $\lambda \leq 1 - \lambda$. Now, let $j \in J$ with $a_j < b_j$ as otherwise (4.7b) and (4.7c) are trivially fulfilled. Then, we have

$$s^\star = \frac{\lambda}{1-\lambda} \leq \frac{1-\lambda}{\lambda} = \frac{(b_j - a_j)(1-\lambda)}{(b_j - a_j)\lambda} = \frac{b_j - \xi_j^\star}{\xi_j^\star - a_j}$$

and

$$s^\star = \frac{\lambda}{1-\lambda} = \frac{\xi_j^\star - a_j}{b_j - \xi_j^\star}.$$

Further, plugging in the definition of λ yields

$$\frac{\Gamma - \xi^\star(J)}{\xi^\star(J) - a(J)} = \frac{\Gamma - a(J) - \lambda(b(J) - a(J))}{\lambda(b(J) - a(J))} = \frac{\lambda}{1-\lambda} = s^\star. \tag{4.8}$$

Note that the constraints (4.7c) and (4.7d) are fulfilled with equality. Last but not least, we have $a \leq \xi^\star \leq b$ and (4.8) gives

$$\begin{aligned} & (1-\lambda) \cdot (\Gamma - \xi^\star(J)) = \lambda \cdot (\xi^\star(J) - a(J)) \\ \Leftrightarrow \qquad & \xi^\star(J) = (1-\lambda) \cdot \Gamma + \lambda \cdot a(J). \end{aligned}$$

As $a(J) \leq \Gamma$, we have $\xi^\star \in \mathcal{U}_\mathrm{B}^+$. Now, let $s \in \mathbb{R}$, $\xi \in \mathcal{U}_\mathrm{B}^+$ be feasible for (4.7). If there is $j \in J$ with $\xi_j \leq \xi_j^\star$, (4.7c) leads to

$$s \leq \frac{\xi_j - a_j}{b_j - \xi_j} \leq \frac{\xi_j^\star - a_j}{b_j - \xi_j^\star} = s^\star.$$

Note that $\xi_j^\star < b_j$. But if $\xi_j > \xi_j^\star$ for all $j \in J$, we get by (4.7d):

$$s \leq \frac{\Gamma - \xi(J)}{\xi(J) - a(J)} < \frac{\Gamma - \xi^\star(J)}{\xi^\star(J) - a(J)} = s^\star.$$

Again, note that $\xi_j^\star > a_j$ for all j. Thus, $s \leq s^\star$ for any solution (s, ξ) to (4.7). In particular, $s < s^\star$ if $\xi \neq \xi^\star$ showing that $(s^\star, \xi^\star)$ is the unique optimal solution. For the translation factor $\rho(\mathcal{U}_\mathrm{B}^+)$, we solve

$$\min \left\{ \alpha \geq 0 \colon \xi - (1 - \alpha) \cdot \xi^\star \geq 0 \text{ for } \xi \in \mathcal{U}_\mathrm{B}^+ \right\}.$$

As $\xi \geq a$ for all $\xi \in \mathcal{U}_\mathrm{B}^+$, it suffices to consider the constraint for scenario a and we get:

$$\begin{aligned}
& a_j - (1 - \alpha) \cdot (a_j + \lambda \cdot (b_j - a_j)) \geq 0 && \text{for } j \in J \\
\Leftrightarrow \quad & -\lambda \cdot (b_j - a_j) + \alpha \cdot (a_j + \lambda \cdot (b_j - a_j)) \geq 0 && \text{for } j \in J \\
\Leftrightarrow \quad & \alpha \geq \frac{\lambda \cdot (b_j - a_j)}{a_j + \lambda \cdot (b_j - a_j)} && \text{for } j \in J.
\end{aligned}$$

This proves the formula for $\rho(\mathcal{U}_\mathrm{B}^+)$. $\square$

Since $\lambda \leq 1/2$ we have $\xi^0 \leq 1/2 \cdot (a + b)$. Furthermore, for the translation factor we deduce:

Corollary 4.10. *Let $\mathcal{U}_B$ be as in Lemma 4.9. If there is $j \in J$ with $a_j = 0$, we have $\rho(\mathcal{U}_B^+) = 1$. Otherwise it holds that*

$$\operatorname*{arg\,max}_{j \in J} \frac{\lambda(b_j - a_j)}{a_j + \lambda(b_j - a_j)} = \operatorname*{arg\,max}_{j \in J} \frac{b_j}{a_j}.$$

Proof. If there is $j \in J$ with $a_j = 0$, the result directly follows from Lemma 4.9. Otherwise, fix $j \in J$ with $a_j > 0$. If $a_j = b_j$, region j is redundant for the translation factor. Thus, with $a_j \neq b_j$ we get

$$\frac{\lambda \cdot (b_j - a_j)}{a_j + \lambda \cdot (b_j - a_j)} = \frac{\lambda}{\lambda + \frac{1}{b_j/a_j - 1}}$$

and the claim holds as $\lambda > 0$ and $b_j/a_j > 1$. $\square$

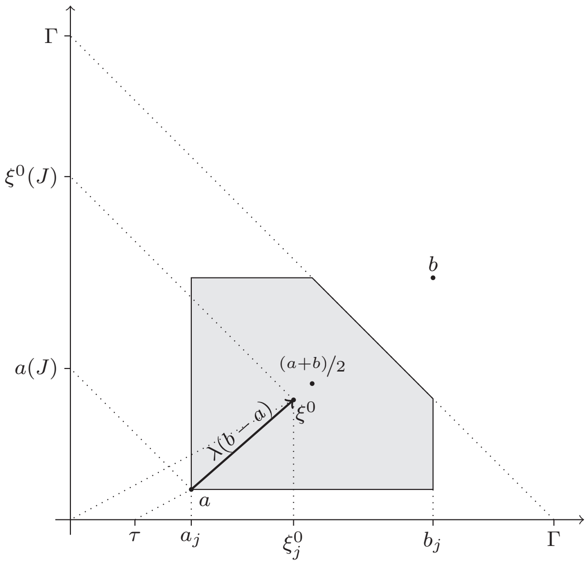


Figure 4.1.: Illustration of geometric factors for budgeted uncertainty $\mathcal{U}_\mathrm{B}$ with $\tau := a - \big(1 - \rho(\mathcal{U}_\mathrm{B}^+)\big) \cdot \xi^0$.

Note that the point of symmetry ξ^0 of $\mathcal{U}_\mathrm{B}^+$ is located on the first half of the line segment between a and b. It is exactly the scenario ξ where the ratios

$$\frac{\Gamma - \xi(J)}{\xi(J) - a(J)} \quad \text{and} \quad \frac{\xi_j - a_j}{b_j - \xi_j}$$

coincide for all $j \in J$. For the translation factor $\rho(\mathcal{U}_\mathrm{B}^+)$, we get that scenario a is moved as far as possible into direction $-\xi^0$ until the first entry attains the value zero. An illustration is shown in Figure 4.1.

For an interpretation of the approximation factor, suppose there is $j \in J$ with $a_j = 0$. Then, we have

$$1 + \frac{\rho(\mathcal{U}_\mathrm{B}^+)}{\mathrm{sym}(\mathcal{U}_\mathrm{B}^+)} = 1 + \frac{1}{\mathrm{sym}(\mathcal{U}_\mathrm{B}^+)} = 1 + \frac{b(J) - a(J)}{\Gamma - a(J)} = 2 + \frac{b(J) - \Gamma}{\Gamma - a(J)}. \quad (4.9)$$

Thus, the ratio (4.9) increases when increasing the lower or upper bounds a, b. Similarly, the ratio increases if Γ decreases. This also captures intuition as we actually decrease the degree of uncertainty in the adjustable model with decreasing Γ or increasing either a or b, while the strictly robust model assumes $\Gamma = b(J)$ by Assumption 3.60. In all cases, the uncertainty set becomes "less" symmetric. On the other hand, for $\Gamma \to b(J)$, we increase the degree of uncertainty and the adjustable model moves towards the strictly robust model. Here, the ratio approaches 2 from above which was the guarantee shown by [BG10] if $\mathcal{U}_{\mathrm{B}}^{+}$ is positive, cf. Example 4.5. Note that, unless $\Gamma = b(J)$, rounding up yields an actual factor of 3 and that, for $\Gamma = b(J)$, it holds that $z(\mathcal{U}_{\mathrm{B}}) = z_{\mathrm{rob}}(\mathcal{U}_{\mathrm{B}})$.

If $a > 0$, let $j^\star \in \arg\max_{j \in J} {}^{b_j}/_{a_j}$ and let $\omega := {}^{b_{j^\star}}/_{a_{j^\star}} > 1$, i.e., $b_j - a_j \leq (\omega - 1) \cdot a_j$ for all $j \in J$. According to the proof of Corollary 4.10 we get

$$\frac{\rho(\mathcal{U}_{\mathrm{B}}^{+})}{\mathrm{sym}(\mathcal{U}_{\mathrm{B}}^{+})} = \frac{\lambda}{\lambda + \frac{1}{\omega - 1}} \cdot \frac{1-\lambda}{\lambda} = \frac{1-\lambda}{\lambda + \frac{1}{\omega-1}}.$$

Thus, $\rho(\mathcal{U}_{\mathrm{B}}^{+}) < 1$ and the ratio improves compared to (4.9). Moreover, we can express λ by means of the symmetry factor $\mathrm{sym}(\mathcal{U}_{\mathrm{B}}^{+})$:

$$\lambda = \frac{1}{1 + \frac{1}{\mathrm{sym}(\mathcal{U}_{\mathrm{B}}^{+})}}.$$

Hence, the closer $\mathcal{U}_{\mathrm{B}}^{+}$ is to a symmetric set and the closer ω is to 1, the better our approximation ratio. In addition to the previous observations for $a \not> 0$, we get that

$$\frac{1-\lambda}{\lambda + \frac{1}{\omega-1}} \leq 1 \quad \Leftrightarrow \quad \lambda \geq \frac{1}{2} - \frac{1}{2(\omega - 1)},$$

so that the ratio 2 is also attained for $\lambda < {}^{1}/_{2}$, i.e., $\Gamma < b(J)$.

All in all, we have seen that an optimal strictly robust solution leads to approximations whose ratios depend only on the geometry of the uncertainty set. Exemplarily, for budgeted uncertainty, we have extensively studied these geometric properties. Unfortunately, we only obtain approximation algorithms if either $q = 1$ or $q = 2$ and if the scenario vector ξ^{wc} can be computed in polynomial time.

Performance of Affine Policy

After having analyzed the strictly robust policy in the context of approximations, we now turn to applying the affine policy as introduced in Section 4.1.1. Thus, we assume that, for a given instance $\mathcal{I}$ of Robust Min q-MSMC and a scenario $\xi \in \mathcal{U}$, the adjustable variables $y(\xi)$ are of the form $y(\xi) = W\xi + w$ for a matrix W and a vector w of appropriate dimensions. This means we concentrate on the quality of the optimal value $z_{\text{aff}}(\mathcal{U})$ to the following problem:

$$
\begin{array}{llrl}
\mathrm{P}_{\text{aff}}(\mathcal{U}) & \min\limits_{x,W,w} & \sum\limits_{i \in I} x_i & \\
& \text{s.t.} & \sum\limits_{i \in N(j)} y(\xi)_{ij} \geq \xi_j & \text{for } j \in J, \xi \in \mathcal{U} \\
& & \sum\limits_{j \in N(i)} y(\xi)_{ij} \leq q \cdot x_i & \text{for } i \in I, \xi \in \mathcal{U} \\
& & y(\xi) = W\xi + w & \text{for } \xi \in \mathcal{U} \\
& & y(\xi)_{ij} \geq 0 & \text{for } i \in I, j \in J, \xi \in \mathcal{U} \\
& & x_i \in \mathbb{N} & \text{for } i \in I \\
& & W \in \mathbb{R}^{|I||J| \times |J|} & \\
& & w \in \mathbb{R}^{|I||J|}. &
\end{array}
$$

Observe that we apply the identification of $\mathbb{R}^{|I| \times |J|}$ and $\mathbb{R}^{|I||J|}$ as mentioned in Section 2.1 on the variables $y(\xi)$ here. As the affine policy is a generalization of the static policy of the previous section, we directly get that $\mathrm{P}_{\text{aff}}(\mathcal{U})$ is feasible and has an optimal solution $(x^\star, W^\star, w^\star)$. We call $x^\star$ an *optimal affine solution* for $\mathcal{I}$. Obviously, for any solution (x, W, w) to $\mathrm{P}_{\text{aff}}(\mathcal{U})$, the vector x is feasible for $\mathcal{I}$ so that $x^\star$ is indeed a solution to $\mathcal{I}$. The results concerning the affine policy given in [Ben+04] are applicable to the LP-relaxation of Robust Min q-MSMC. But in general, we cannot assume that $y(\xi)$ affinely depends on the scenario ξ in an optimal solution as the subsequent example shows.

Example 4.11 (Suboptimality of Affine Policy)**.** Consider the following instance of Robust Min q-MSMC with $I := \{A, B\}$, $J := \{1, 2, 3, 4\}$, and

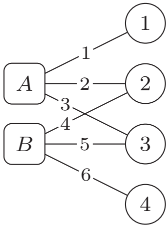


Figure 4.2.: Bipartite graph G corresponding to Example 4.11.

the uncertainty set

$$\mathcal{U} := \left\{ \begin{pmatrix} q \\ q \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} q \\ 0 \\ q \\ 0 \end{pmatrix}, \begin{pmatrix} q \\ 0 \\ 0 \\ q \end{pmatrix}, \begin{pmatrix} 0 \\ q \\ q \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ q \\ 0 \\ q \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ q \\ q \end{pmatrix} \right\}.$$

Furthermore, let $N_G(A) := \{1, 2, 3\}$ and $N_G(B) := \{2, 3, 4\}$ and let E be the corresponding edge set. The bipartite graph $G := (I \cup J, E)$ is illustrated in Figure 4.2. It can readily be seen that the optimal value of the instance is 2 which is attained by setting $x^\star_A = x^\star_B = 1$. Note that this optimal solution is unique. We show that, given $x^\star$, no solution for $y(\xi)$ is an affine function of ξ. To see this, we label the edges as depicted in Figure 4.2. To ensure legibility we omit $y(\xi)_{ij}$ if $i \notin N_G(j)$ and write $y(\xi) = (y(\xi)_1, \ldots, y(\xi)_6)^T$. Then, we must have

$$y \begin{pmatrix} q \\ q \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} q \\ 0 \\ 0 \\ q \\ 0 \\ 0 \end{pmatrix}, \; y \begin{pmatrix} 0 \\ 0 \\ q \\ q \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ q \\ 0 \\ 0 \\ q \end{pmatrix}, \; y \begin{pmatrix} q \\ 0 \\ q \\ 0 \end{pmatrix} = \begin{pmatrix} q \\ 0 \\ 0 \\ 0 \\ q \\ 0 \end{pmatrix}, \text{ and } y \begin{pmatrix} 0 \\ q \\ 0 \\ q \end{pmatrix} = \begin{pmatrix} 0 \\ q \\ 0 \\ 0 \\ 0 \\ q \end{pmatrix}.$$

Now, suppose $y(\xi) = W\xi + w$ for a matrix $W \in \mathbb{R}^{6\times 4}$ and a vector $w \in \mathbb{R}^6$. This leads to the following contradiction:

$$W \begin{pmatrix} q \\ q \\ q \\ q \end{pmatrix} + 2w = y \begin{pmatrix} q \\ q \\ 0 \\ 0 \end{pmatrix} + y \begin{pmatrix} 0 \\ 0 \\ q \\ q \end{pmatrix} \neq y \begin{pmatrix} q \\ 0 \\ q \\ 0 \end{pmatrix} + y \begin{pmatrix} 0 \\ q \\ 0 \\ q \end{pmatrix} = W \begin{pmatrix} q \\ q \\ q \\ q \end{pmatrix} + 2w.$$

Thus, the variable vector $y(\xi)$ does not affinely depend on ξ in an optimal solution to the constructed instance. Moreover, this result also holds for budgeted uncertainty as

$$\mathcal{U} \subseteq \left\{\xi \in \mathbb{N}^4 \colon 0 \leq \xi \leq q \cdot 1, \xi(J) \leq 2q\right\}$$

and $x^\star$ is also optimal for the extended uncertainty set. $\lhd$

Despite the previous example, in [BG11], it is shown that an optimal affine solution for $\mathcal{I}$ does not behave too badly compared to an optimal solution. The ratio of the corresponding solution values can be bounded by $3\sqrt{|I| + |J|}+1$. In our case, this bound can even be improved to $3\sqrt{|J|}+1$ as we only have $|J|$ uncertain parameters. Thus, we need to compute an optimal affine solution to approximate $z(\mathcal{U})$ within this ratio. Unfortunately, the computation of an optimal affine solution is APX-hard which can be shown by an L-reduction from Min Dominating Set(3) similar to Theorem 3.30. We formalize the adapted optimization problem and present the necessary adjustments of the proof.

Problem 4.12 (Robust Min q-MSMC with affine policy).
Instance: An instance of Robust Min q-MSMC.
Solution: A solution (x, W, w) to $\mathrm{P}_{\mathrm{aff}}(\mathcal{U})$.
Measure: The value $x(I)$.

Theorem 4.13. *For any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-MSMC with affine policy is* APX*-hard.*

Proof. Let an instance $\mathcal{I}$ of Min Dominating Set(3) be given, i.e., a simple graph $G = (V, E)$ with $V = \{1, \ldots, n\}$ for $n \in \mathbb{N}_{>0}$ and $\Delta_G \leq 3$. To construct the instance $\mathcal{I}'$ of Robust Min q-MSMC with affine policy, we refer to the proof of Theorem 3.30. Note that $\mathcal{U} = \{0, e_1, \ldots, e_n\}$ and let G' be the constructed bipartite graph.

Let $V' \subseteq V = I$ be an optimal solution to $\mathcal{I}$. Then, we set $x_i := 1$ for all $i \in V'$ and zero otherwise. By Theorem 3.30 we get that x is feasible for the corresponding instance of Robust Min q-MSMC. As $|\mathcal{U}| = n + 1$, we can compute a solution $\bar{y}$ to the (non-affine) adjustable variables $y(e_1), \ldots, y(e_n)$ in polynomial time using n Max Flow computations. We set

$$W := \left(\bar{y}(e_1), \ldots, \bar{y}(e_n)\right) \quad \text{and} \quad w := \bar{y}(0) := 0.$$

Hence, (x, W, w) is feasible for $\mathcal{I}'$ and $\mathrm{OPT}(\mathcal{I}') \leq \mathrm{OPT}(\mathcal{I})$.

Conversely, suppose that (x, W, w) is a solution to $\mathcal{I}'$. Then, x is feasible for the corresponding instance of Robust Min q-MSMC and we can assume that W is given as above as well as $w = 0$. Furthermore, by construction of $\mathcal{U}$ we can assume that $x_i \leq 1$ for $i \in I$. We define $V' := \{v \in V : x_v = 1\}$ and claim that V' is a dominating set for G. To this end, select a vertex $u \in V$. As $e_u \in \mathcal{U}$ and by feasibility of (x, W, w), we know that there is $v \in N_{G'}(u)$ with $x_v = 1$, i.e., $v \in V'$. By construction of G', either $v = u$ or the vertices u and v are adjacent in G. Therefore, $\mathrm{SOL}(\mathcal{I}, V') \leq \mathrm{SOL}(\mathcal{I}', (x, W, w))$ and $\mathrm{SOL}(\mathcal{I}, V') - \mathrm{OPT}(\mathcal{I}) \leq \mathrm{SOL}(\mathcal{I}', (x, W, w)) - \mathrm{OPT}(\mathcal{I}')$. In particular, we also have $\mathrm{OPT}(\mathcal{I}) \leq \mathrm{OPT}(\mathcal{I}')$. □

In the proof of Theorem 4.13, we repeatedly exploit the fact that the convex hull of the constructed uncertainty set $\mathcal{U} = \{0, e_1, \ldots, e_n\}$ is a simplex for which the affine policy is optimal, cf. [BG11]. Despite this hardness result of applying the affine policy, the $\mathcal{O}(\sqrt{|J|})$-ratio is appealing when utilizing an optimal affine solution as a start solution in the constraint generation method (if the instance size allows for a computation of the optimal affine solution).

Performance of Dominating Uncertainty Sets

As a final approximation concept for Min ARC we investigate the performance of dominating uncertainty sets. Considering $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$, the idea is to replace the uncertainty set $\mathcal{Z}$ with an "easier" uncertainty set $\mathcal{Z}'$ that additionally dominates $\mathcal{Z}$, i.e., for every $\zeta \in \mathcal{Z}$, there is $\zeta' \in \mathcal{Z}'$ with $\zeta \leq \zeta'$. Note that $\mathcal{Z}$ does not need to be a subset of $\mathcal{Z}'$ here.

Definition 4.14 (Dominating Solution)**.** Let an instance of Min ARC with uncertainty set $\mathcal{Z} \subseteq \mathbb{R}^m_{\geq 0}$ be given and let $\mathcal{Z}' \subseteq \mathbb{R}^m_{\geq 0}$ be an uncertainty set dominating $\mathcal{Z}$. Let (x', y') be feasible for $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z}')$. A solution (x, y) to $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$ with $x := x'$ and, for every $\zeta \in \mathcal{Z}$, $y(\zeta) := y'(\zeta')$, where $\zeta' \in \mathcal{Z}'$ dominates ζ, is called a *dominating solution* with respect to (x', y'). If (x', y') is optimal for $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z}')$, we call (x, y) an *optimal dominating solution*.

As implicitly given in Definition 4.14 it can readily be seen that a dominating solution is feasible for $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$. Further, the optimal value of $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z}')$ is an upper bound on the optimal value of $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$: For an optimal dominating solution (x, y) with respect to (x', y'), we have

$$z_{\text{arc}}(\mathcal{Z}) \leq c^T x + \max_{\zeta \in \mathcal{Z}} d^T y(\zeta) \leq c^T x' + \max_{\zeta' \in \mathcal{Z}'} d^T y'(\zeta') = z_{\text{arc}}(\mathcal{Z}'). \quad (4.10)$$

as $c, d \geq 0$ and by definition of (x, y).

For our purposes, let $p_2 = 0$ and assume without loss of generality that the first $r \in \mathbb{N}$ constraints in $\mathrm{P}_{\text{arc}}(\mathcal{Z})$ have an uncertain right-hand side. Then, the problem can be rewritten as

$$\begin{array}{llll}
\mathrm{P}'_{\text{arc}}(\mathcal{Z}_r) & \min\limits_{x,\, y(\zeta)} & c^T x + \max\limits_{\zeta \in \mathcal{Z}_r} d^T y(\zeta) & \\
& \text{s.t.} & Ax + By(\zeta) \geq \begin{pmatrix} \zeta \\ \bar{\zeta} \end{pmatrix} & \text{for } \zeta \in \mathcal{Z}_r \\
& & x, y(\zeta) \geq 0 & \text{for } \zeta \in \mathcal{Z}_r \\
& & x \in \mathbb{R}^{n_1 - p_1} \times \mathbb{N}^{p_1} & \\
& & y(\zeta) \in \mathbb{R}^{n_2} & \text{for } \zeta \in \mathcal{Z}_r,
\end{array}$$

with uncertainty set $\mathcal{Z}_r \subseteq \mathbb{R}^r_{\geq 0}$ and $\bar{\zeta} \in \mathbb{Q}^{m-r}_{\geq 0}$ fixed. Denote by $z'_{\text{arc}}(\mathcal{Z}_r)$ the optimal value of $\mathrm{P}'_{\text{arc}}(\mathcal{Z}_r)$.

Definition 4.15 (Prevailing Set [BG11])**.** Given $\mathrm{P}'_{\text{arc}}(\mathcal{Z}_r)$ with uncertainty set $\mathcal{Z}_r \subseteq \mathbb{R}^r_{\geq 0}$, a *prevailing set* $\mathcal{Z}_0$ corresponding to $\mathcal{Z}_r$ is given by

$$\mathcal{Z}_0 := \left\{2\beta, \lceil 2\sqrt{r} \rceil \cdot \beta^1, \ldots, \lceil 2\sqrt{r} \rceil \cdot \beta^r \right\},$$

where, for $j \in \{1, \ldots, r\}$, $\beta^j \in \arg\max_{\zeta \in \mathcal{Z}_r} \zeta_j$ and β is obtained from Algorithm 4.1. The vectors $\beta, \beta^1, \ldots, \beta^r$ are called *generating vectors* corresponding to $\mathcal{Z}_r$ and we say that $\mathcal{Z}_0$ is *generated* by $\beta, \beta^1, \ldots, \beta^r$.

Example 4.16. For an illustration of a prevailing set, consider the convex hull of a budgeted uncertainty set

$$\mathcal{Z} := \left\{\xi \in \mathbb{R}^2 \colon 0 \leq \xi \leq 4 \cdot \mathbb{1}, \xi_1 + \xi_2 \leq 7\right\}$$

with $r := 2$. We can choose $\beta^1 = (4, 0)^T$ and $\beta^2 = (0, 4)^T$. Algorithm 4.1 applied to $\mathcal{Z}$ may choose the scenario $u^1 := (3, 4)^T$ in the first iteration as

$$\frac{3}{4} + \frac{4}{4} = 1.75 > \sqrt{2}.$$

Algorithm 4.1 Computation of scenario β for $\mathcal{Z}_0$ [BG11].

Input: An uncertainty set $\mathcal{Z} \subseteq \mathbb{R}^r_{\geq 0}$.
Output: Scenario β for a prevailing set $\mathcal{Z}_0$.

$J_1 := \{1, \dots, r\}$, $J_2 := \varnothing$, $\beta := 0$
$\mu_j := \max_{\zeta \in \mathcal{Z}} \zeta_j$ for $j = 1, \dots, r$
while $\exists \zeta \in \mathcal{Z}$ **with** $\sum_{j \in J_1} \zeta_j/\mu_j > \sqrt{r}$ **do**
  $u \in \arg\max_{\zeta \in \mathcal{Z}} \sum_{j \in J_1} \zeta_j/\mu_j$
  $\beta := \beta + u$
  $J' := J_1$
  for $j \in J'$ **do**
    if $\beta_j \geq \mu_j$ **then**
      $J_1 := J_1 \setminus \{j\}$
      $J_2 := J_2 \cup \{j\}$
    end if
  end for
end while
return β

Thus, we get $J_1 = \{1\}$ and $J_2 = \{2\}$ and the algorithm stops. Hence, we have $\beta = u^1 = (3, 4)^T$ and a prevailing set $\mathcal{Z}_0$ corresponding to $\mathcal{Z}$ is given by

$$\mathcal{Z}_0 = \left\{ 2\begin{pmatrix}3\\4\end{pmatrix}, 3\begin{pmatrix}4\\0\end{pmatrix}, 3\begin{pmatrix}0\\4\end{pmatrix} \right\}.$$

The sets $\mathcal{Z}$ and $\mathcal{Z}_0^+ := \operatorname{conv}(\mathcal{Z}_0)$ are shown in Figure 4.3. $\triangleleft$

Lemma 4.17 ([BG11])**.** *Let $\mathcal{Z}_r$ and $\mathcal{Z}_0$ be as in Definition 4.15. Then, $\mathcal{Z}_0^+ := \operatorname{conv}(\mathcal{Z}_0)$ dominates $\mathcal{Z}_r$ and, given $\zeta \in \mathcal{Z}_r$ and $J_1 \subseteq \{1, \dots, r\}$ from Algorithm 4.1, a scenario $\zeta' \in \mathcal{Z}_0^+$ dominating ζ can be found in polynomial time.* $\square$

In particular, in [BG11], the authors show that, if $p_1 = p_2 = 0$ and $r = m$, the uncertainty set $\mathcal{Z}_0^+$ leads to a non-affine $4\sqrt{m}$-approximation for $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$ as

$$z_{\mathrm{arc}}(\mathcal{Z}_0^+) \leq 4\sqrt{m} \cdot z_{\mathrm{arc}}(\mathcal{Z}) \tag{4.11}$$

and by (4.10). In fact, the proof of (4.11) in [BG11] can be adapted to the case of only $r \leq m$ uncertain parameters and $p_1 \geq 0$, $p_2 = 0$, cf. Appendix A.

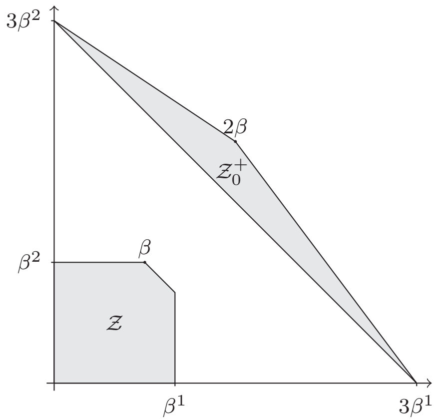


Figure 4.3.: Sets $\mathcal{Z}$ and $\mathcal{Z}_0^+$ corresponding to Example 4.16.

With Lemma A.1 we get that

$$z'_{\text{arc}}(\mathcal{Z}_0) \leq \left(4\sqrt{r}+1\right) \cdot z'_{\text{arc}}(\mathcal{Z}_r)$$

for a prevailing set $\mathcal{Z}_0$ corresponding to $\mathcal{Z}_r$. Given an optimal solution (x, y) to $\mathrm{P}'_{\text{arc}}(\mathcal{Z}_0)$, we augment (x, y) to an optimal solution (x, y') to $\mathrm{P}'_{\text{arc}}(\mathcal{Z}_0^+)$ by Lemma 2.10. By Lemma 4.17 an optimal dominating solution $(x, \bar{y})$ to $\mathrm{P}'_{\text{arc}}(\mathcal{Z}_r)$ corresponding to (x, y') yields a $(4\sqrt{r}+1)$-approximate solution to $\mathrm{P}'_{\text{arc}}(\mathcal{Z}_r)$ as

$$c^T x + d^T \bar{y}(\zeta) \leq z'_{\text{arc}}(\mathcal{Z}_0^+) = z'_{\text{arc}}(\mathcal{Z}_0) \leq \left(4\sqrt{r}+1\right) \cdot z'_{\text{arc}}(\mathcal{Z}_r)$$

for every $\zeta \in \mathcal{Z}_r$. Concerning Algorithm 4.1, in [BG11], it is shown that the number of iterations is bounded by $2\sqrt{r}$ implying that its running time mainly depends on the computation of Step 4. Let (x, y), (x, y'), and $(x, \bar{y})$ be as above. If

(a) a set of generating vectors corresponding to $\mathcal{Z}_r$,

(b) an optimal solution (x, y) to $\mathrm{P}'_{\text{arc}}(\mathcal{Z}_0)$, and,

(c) for every scenario $\zeta \in \mathcal{Z}_r$, a dominating scenario $\zeta' \in \mathcal{Z}_0^+$ and the corresponding second stage solution $\bar{y}(\zeta) = y'(\zeta')$ to $\mathrm{P}'_{\text{arc}}(\mathcal{Z}_r)$

can be accomplished in polynomial time, this gives an approximation algorithm for Min ARC with r uncertain parameters and $p_2 = 0$ (cf. Definition 2.9).

Considering a general instance $\mathcal{I}$ of Robust Min q-MSMC with bipartite graph $G = (I \cup J, E)$, $n := |I|$, $m := |J|$, and uncertainty set $\mathcal{U} \subseteq \mathbb{N}^m$, we have m uncertain and n certain parameters. Further, in addition to the parameters given in the proof of Lemma 4.1, we set $\mathcal{Z}_m := \mathcal{U}$ and $\bar{\zeta} := 0 \in \mathbb{N}^n$. Then, $\mathrm{P}'_{\mathrm{arc}}(\mathcal{Z}_m) = \mathrm{P}(\mathcal{U})$. Let $\mathcal{U}_0 \subseteq \mathbb{N}^m$ be a prevailing set corresponding to $\mathcal{U}$. The generating vectors $\beta, \beta^1, \ldots, \beta^m$ can be computed in polynomial time if, for any $c \in \mathbb{Q}^m_{\geq 0}$, $\max_{\xi \in \mathcal{U}} c^T \xi$ can be solved in polynomial time, i.e., the uncertainty set $\mathcal{U}$ has a polynomial time optimization oracle. Considering item (c) and an optimal solution (x, y) to $\mathrm{P}(\mathcal{U}_0)$, we can directly obtain our adjustable variables by computing maximum flows in the networks $H_G(q \cdot x, \xi)$ for $\xi \in \mathcal{U}$, cf. Theorem 3.8. Thus, item (b) remains to be investigated.

Problem 4.18 (Robust q-MSMC with dominating uncertainty)**.**
Instance: An instance of Robust q-MSMC with uncertainty set $\mathcal{U}$ and generating vectors $\beta, \beta^1, \ldots, \beta^{|J|}$ corresponding to $\mathcal{U}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for every subset $S \subseteq J$ and every scenario ξ in the prevailing set generated by $\beta, \beta^1, \ldots, \beta^{|J|}$, we have $q \cdot x(N(S)) \geq \xi(S)$?

We show that Problem 4.18 is NP-complete. Hence, most likely, we do not obtain an approximation algorithm for Robust Min q-MSMC with performance guarantee $4\sqrt{|J|}+1$ when applying this approach. We consider the $\{k\}$-Domination problem for a given fixed integer $k \in \mathbb{N}_{>0}$ [Gai+03]:

Problem 4.19 ($\{k\}$-Domination)**.**
Instance: A simple graph $G = (V, E)$ and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|V|}$ such that $x(V) \leq B$ and $x(N[v]) \geq k$ for $v \in V$?

In [Gai+03], Gairing et al. show that $\{k\}$-Domination is NP-complete for any fixed $k \in \mathbb{N}_{>0}$ by providing a polynomial time reduction from Set Cover. This reduction can be extended to an APX-hardness proof, cf. Appendix B. With an instance of Set Cover consisting of $N \in \mathbb{N}_{>0}$ items and $M \in \mathbb{N}_{>0}$ sets, they construct a graph with a total of $2 + M + N \cdot (3k + 1)$ vertices.

Following their proof, we observe that it also applies if k is polynomial in M and N as the construction remains a polynomial time reduction. Thus, we can show NP-completeness of the following problem:

Problem 4.20 (Poly-Domination)**.**
Instance: A simple graph $G = (V, E)$ with $n \in \mathbb{N}_{>0}$ vertices, a positive integer k that is $\mathcal{O}(n^2)$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|V|}$ such that $x(V) \leq B$ and $x(N[v]) \geq k$ for $v \in V$?

Theorem 4.21. *Poly-Domination is strongly* NP*-complete even if $k \geq 2\sqrt{n}$ and k is even.*

Proof. Given an instance of Set Cover with $N \in \mathbb{N}_{>0}$ items and $M \in \mathbb{N}_{>0}$ sets, we choose $k := 18N^2 + 8N + 2M + 4$. Note that k is even and it can be verified that

$$k^2 \geq 4 \cdot (2 + M + N \cdot (3k + 1)).$$

Thus, constructing the graph $G = (V, E)$ with $|V| = 2+M+N\cdot(3k+1)$ as in the proof of Theorem B.1, we additionally have the property that $k \geq 2\sqrt{|V|}$ and k is $\mathcal{O}(|V|^2)$. As k is polynomial in N and M, we can replicate this proof and obtain NP-hardness of Poly-Domination with Corollary B.2. Finally, Poly-Domination is contained in NP as k is polynomially bounded in the number of vertices of the given graph. □

Remark 4.22. The proof of Theorem 4.21 applies to any function $f : \mathbb{N} \to \mathbb{R}$ such that, for any $a, b \in \mathbb{N}_{>0}$, the inequality

$$k \geq f(a + b \cdot k)$$

has an integral solution $k \in \mathbb{N}_{>0}$ that is polynomial in a and b. Given any such fixed function f, we obtain NP-completeness of Poly-Domination even if $k \geq f(n)$, where $n \in \mathbb{N}_{>0}$ is the number of vertices of the graph. For instance, this holds for linear functions with slope less than 1.

To show NP-completeness of the initial Problem 4.18, we subsequently prove NP-completeness of the following, more specific, domination problem:

Problem 4.23 (Order-Domination).
Instance: A simple graph $G = (V, E)$ with $n \in \mathbb{N}_{>0}$ vertices and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|V|}$ such that $x(V) \leq B$ and $x(N[v]) \geq 2\sqrt{n}$ for $v \in V$?

Theorem 4.24. *Order-Domination is strongly* NP*-complete even if $n = p^2$ for some $p \in \mathbb{N}_{>0}$.*

Proof. Order-Domination is contained in NP: Given a value $x_v \in \mathbb{N}$ with $x_v \leq 4n$ for every vertex $v \in V$, we can check in polynomial time whether $(x(N[v]))^2 \geq 4n$ holds for every $v \in V$.

Further, NP-hardness follows by a polynomial time reduction from Poly-Domination with $k \geq 2\sqrt{n}$ and k even. Given a simple graph $G = (V, E)$ with $n \in \mathbb{N}_{>0}$ vertices, we add $l := {}^{k^2}/_4 - n$ singletons to G. Note that $l \in \mathbb{N}$ by our assumptions on k. Denote the new graph by $G' = (V', E)$ and observe that $k = 2\sqrt{n + l} = 2\sqrt{|V'|}$. The construction of G' can be accomplished in polynomial time as k is $\mathcal{O}(n^2)$. In the following, we assume that, in any solution $(x, x') \in \mathbb{N}^{n+l}$ of the Order-Domination problem on G', we have $x'_i = k$ for every newly added singleton i as this only improves our solution. Then, a vector $x \in \mathbb{N}^n$ is a solution to the Poly-Domination problem on G with value at most $B \in \mathbb{N}_{>0}$ if and only if $(x, x') \in \mathbb{N}^{n+l}$ is a solution to the Order-Domination problem on G' with value at most $B + l \cdot k$. $\square$

The proof of Theorem 4.24 reveals NP-completeness of Order-Domination even if the input graph G contains a quadratic number of vertices. We are now ready to show our desired complexity result.

Theorem 4.25. *For any fixed $q \in \mathbb{N}_{>0}$, Robust q-MSMC with dominating uncertainty is* NP*-complete.*

Proof. Let an instance $\mathcal{I}$ of Robust q-MSMC with dominating uncertainty with bipartite graph G and $m := |J|$ be given. Then, the corresponding prevailing set is

$$\mathcal{U}_0 := \left\{2\beta, \lceil 2\sqrt{m} \rceil \cdot \beta^1, \ldots, \lceil 2\sqrt{m} \rceil \cdot \beta^m \right\}.$$

To check feasibility of a given vector $\bar{x} \in \mathbb{N}^{|I|}$, it suffices to compute the maximum flow values in the networks $H_G(q \cdot \bar{x}, \xi)$ for $\xi \in \mathcal{U}_0$, cf. Theorem 3.36

and Corollary 3.37. This can be done in polynomial time, especially as we can assume a fixed precision of the square root $\sqrt{m}$ due to rounding. Furthermore, note that we can assume $\langle \bar{x} \rangle$ to be polynomially bounded in the encoding length of $\mathcal{I}$. Thus, Robust q-MSMC with dominating uncertainty is contained in NP.

We show NP-hardness by a polynomial time reduction from Order-Domination. Given a simple graph $G = (V, E)$ with $V = \{1, \dots, p^2\}$ for $p \in \mathbb{N}_{>0}$ and a positive integer $B \in \mathbb{N}_{>0}$, we construct the bipartite graph $G' = (I \cup J, E')$ as in the proof of Theorem 3.30. Then, $|I| = |J| = p^2$ and let

$$\mathcal{U} := \left\{ \xi \in \mathbb{N}^{p^2} : 0 \leq \xi \leq q \cdot 1, \xi(J) \leq qp \right\}.$$

Given p^2, the value p can be found in polynomial time as we can simply test all values in $\{1, \dots, p^2\}$ or apply binary search. To that end, note that the encoding length of G is at least p^2.

Considering Algorithm 4.1 applied to $\mathcal{U}$, we see that $\beta = 0$ as, for any scenario $\xi \in \mathcal{U}$,

$$\sum_{j \in J} \frac{\xi_j}{\mu_j} \leq \frac{qp}{q} = p = \sqrt{p^2}.$$

Thus, a prevailing set $\mathcal{U}_0$ is given by $\beta^j := q \cdot e_j$ for $j \in J$ and we get

$$\mathcal{U}_0 = \left\{ 2\beta, 2\sqrt{p^2} \cdot \beta^1, \dots, 2\sqrt{p^2} \cdot \beta^{p^2} \right\} = \left\{ 0, 2pq \cdot e_1, \dots, 2pq \cdot e_{p^2} \right\}.$$

The constructed instance of Robust q-MSMC with dominating uncertainty asks for $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ and

$$\begin{aligned}
& q \cdot x(N_{G'}(S)) \geq \max_{\xi \in \mathcal{U}_0} \xi(S) = 2pq && \text{for } S \subseteq J, S \neq \emptyset \\
\Leftrightarrow \quad & x(N_{G'}(S)) \geq 2p && \text{for } S \subseteq J, S \neq \emptyset \\
\Leftrightarrow \quad & x(N_{G'}(j)) \geq 2p && \text{for } j \in J.
\end{aligned}$$

Note that the last equivalence follows as the right-hand side does not depend on $S \subseteq J$. Rewriting this in terms of the graph G, we get

$$x(N_{G'}(j)) \geq 2p \text{ for } j \in J \quad \Leftrightarrow \quad x(N_G[v]) \geq 2p \text{ for } v \in V.$$

Thus, $x \in \mathbb{N}^{p^2}$ is a solution to the Order-Domination problem on G with value at most B if and only if x is a solution to the constructed instance of Robust q-MSMC with dominating uncertainty with value at most B. $\square$

Remark 4.26. In the proof of Theorem 4.25 we can also choose the uncertainty set $\mathcal{U} = \left\{q \cdot e_1, \ldots, q \cdot e_{p^2}\right\}$ that only contains p^2 many scenarios and, hence, is polynomial time enumerable. The choice of the budgeted uncertainty set will be of use in Section 4.2.3.

All in all, for now, we may only use the optimal solution to an instance of Robust Min q-MSMC with dominating uncertainty to obtain an approximate start solution to the corresponding instance of Robust Min q-MSMC for small instances. Due to Appendix A, the so found optimal dominating solution has an approximation ratio of $4\sqrt{|J|} + 1$, where J is the set of regions of the given instance. Nevertheless, in Section 4.2.3, we see that prevailing sets become useful for instances of Robust Min q-MSMC with an exponential number of scenarios.

4.2. Approximation based on a Set Cover Approach

In the previous section, we encountered hardness results concerning possible approximations for Robust Min q-MSMC. The solution policies present in literature mainly lead to NP-hard problems when applied to Robust Min q-MSMC. Yet, as mentioned in Section 3.2, Robust Min q-MSMC also admits for an interpretation as a robust Min Multiset Multicover problem. In this section, we aim to exploit this property to obtain an approximation algorithm for many classes of uncertainty sets.

4.2.1. Literature Review

We review results concerning approximations and approximation algorithms for Min Set Cover and its generalizations. For the definitions of these problems we refer to Section 2.4.

The research on approximation algorithms for the classical Min Set Cover problem starts with Johnson and Lovász, who introduce a greedy algorithm that always chooses the set containing the most newly covered elements [Joh74; Lov75]. Later, Chvátal generalizes this idea for the weighted Min Set Cover problem [Chv79]. They show that their algorithms provide an approximation guarantee of $\mathrm{H}(k)$, where $k \in \mathbb{N}$ is the maximum number of elements of a set, and also give examples for which the ratio is tight.

Later, an extension of the algorithm achieves a guarantee of $\mathrm{H}(k) - 1/6$, cf. [GHY93]. The general algorithm is adapted to the Min Set Multicover problem in [FW82]. For an instance of this problem with sets $C_1, \ldots, C_n$ and corresponding cost values $c_1, \ldots, c_n$, let $\beta_1 := \max_j |C_j|/c_j$ and $\beta_2 := \min_j c_j$. Fisher and Wolsey show an approximation guarantee of $1 + \ln(\beta_1 \cdot \beta_2)$ for the extended greedy, cf. [FW82].

Moreover, the authors in [Hoc82] and [BE85] present approximation algorithms for the weighted Min Set Cover problem with ratio f where $f \in \mathbb{N}$ is the maximum number of sets containing a common element. This result is generalized to Min Set Multicover in [HH86] and to Min CIP in [PC10]. In this case, the ratio corresponds to the maximum number of non-zero entries in any row of the constraint matrix A. Moreover, an LP-based approximation combining the values k and f is given in [SS12].

Results for approximating the Min Multiset Multicover problem are presented by Dobson [Dob82] as well as Vazirani [Vaz03]. Therein, it is shown that the greedy algorithm can be augmented and one reobtains the ratio of $\mathrm{H}(k)$. Note that the elements in a set are now counted multiple times.

Besides these sequential guarantees, there are several publications that provide approximation algorithms for Min CIP_∞ based on randomized rounding, e.g., [Sri01; KY05; Sri06; CHS16; CQ19]. These algorithms require an optimal solution to the LP-relaxation of the corresponding covering problem. For Robust Min q-MSMC, we see in Section 4.2.3 that its covering equivalent leads to an IP where the number of variables is potentially exponential in q and $|\mathcal{U}|$. Thus, these rounding-based algorithms are only applicable efficiently for $|\mathcal{U}|$ fixed.

Hence, for Robust Min q-MSMC, we now concentrate on the details of Dobson's greedy algorithm.

4.2.2. Dobson's Algorithm

In [Dob82], Dobson introduces a greedy algorithm for approximating the optimal value of instances of Min Multiset Multicover, cf. Section 2.4. Hence, he considers integer programming problems of the form

$$\min \left\{ c^T x \colon Ax \geq b, x \in \mathbb{N}^n \right\} \tag{4.12}$$

for $A \in \mathbb{N}^{m \times n}$, $b \in \mathbb{N}^m_{>0}$, and $c \in \mathbb{N}^n_{>0}$ with $A_{ij} \leq b_i$ for all $i \in \{1, \ldots, m\}$ and $j \in \{1, \ldots, n\}$. The procedure is a direct generalization of the approximation

Algorithm 4.2 Dobson's greedy algorithm to solve (4.12).

Input: $A \in \mathbb{N}^{m \times n}$, $b \in \mathbb{N}^m$, $c \in \mathbb{N}^n$, $A_{ij} \leq b_i$ for all $i \in \{1, \ldots, m\}$ and $j \in \{1, \ldots, n\}$.
Output: A solution $x \in \mathbb{N}^n$ to (4.12) with solution value $z \in \mathbb{N}$.

```
x := 0, z := 0
while Σ_{i=1}^m b_i ≥ 1 do
    k ∈ arg min_{j∈{1,...,n}} c_j / Σ_{i=1}^m A_ij
    x_k := x_k + 1
    b_i := b_i − A_ik for all i ∈ {1,...,m}
    A_ij := min{A_ij, b_i} for all i ∈ {1,...,m} and j ∈ {1,...,n}
    z := z + c_k
end while
return x, z
```

algorithms for Min Set Cover in [Chv79; Lov75; Joh74]. In these works, one iteratively chooses the set/column, which minimizes the cost over utility ratio, and increases its corresponding variable by one. In the case of (4.12), this leads to selecting a column $j \in \{1, \ldots, n\}$ which minimizes the ratio

$$\frac{c_j}{\sum_{i=1}^m A_{ij}}$$

in the first iteration. The algorithm is depicted in Algorithm 4.2. According to [Dob82], it runs in polynomial time with at most $n + m$ iterations using a slightly enhanced implementation. Further, the returned solution $x \in \mathbb{N}^n$ fulfills

$$c^T x \leq \mathrm{H}\left(\max_j \sum_{i=1}^m A_{ij}\right) \cdot c^T x^\star, \tag{4.13}$$

with $x^\star \in \mathbb{N}^n$ being an optimal solution to (4.12). The bound is tight due to the results in [Joh74; Chv79]. In fact, the ratio in (4.13) can even be strengthened as it also holds if $x^\star$ is replaced by an optimal solution x^{LP} to the LP-relaxation of (4.12). Hence, we additionally obtain a bound on the the integrality gap.

4.2.3. Application of Dobson's Algorithm

In this section, we see that even the robust version of Min q-MSMC can be interpreted as a Min Multiset Multicover problem. But, in comparison to Min q-MSMC, the input size of the related Min Multiset Multicover problem might be exponential in the input size of Robust Min q-MSMC, cf. Remark 3.2.

Throughout this section, let an instance $\mathcal{I}$ of Robust Min q-MSMC be given with bipartige graph $G = (V, E)$ and uncertainty set $\mathcal{U}$. Let $k := |\mathcal{U}|$ for $k \in \mathbb{N}$ so that $\mathcal{U} = \left\{\xi^1, \dots, \xi^k\right\}$ and, for $i \in I$, let $\mathcal{S}_i$ contain all multisubsets of $N(i)$ of size q. Consider an element S in

$$\mathcal{S}_i^k := \bigtimes_{l=1}^{k} \mathcal{S}_i,$$

i.e., $S = (S_1, \dots, S_k)$ with $S_l \in \mathcal{S}_i$ for $l \in \{1, \dots, k\}$. We interpret S as putting a supplier into location i who, for every l, covers the clients in the set S_l in scenario $\xi^l \in \mathcal{U}$. Note, that

$$|\mathcal{S}_i^k| = |\mathcal{S}_i|^k = \binom{|N(i)| + q - 1}{q}^k \leq |N(i)|^{qk}$$

and that, for two distinct locations i_1, i_2, it is possible that $\mathcal{S}_{i_1}^k \cap \mathcal{S}_{i_2}^k \neq \varnothing$. Now, let $x(S, i) \in \mathbb{N}$ be the frequency of choosing $S \in \mathcal{S}_i^k$. Then, Robust Min q-MSMC is equivalent to

$$\min_x \quad \sum_{i \in I} \sum_{S \in \mathcal{S}_i^k} x(S, i) \tag{4.14a}$$

$$\text{s.t.} \quad \sum_{i \in I} \sum_{S \in \mathcal{S}_i^k} \mathrm{m}(j, S_l) \cdot x(S, i) \geq \xi_j^l \quad \text{for } j \in J, l \in \{1, \dots, k\} \tag{4.14b}$$

$$x(S, i) \in \mathbb{N} \quad \text{for } i \in I, S \in \mathcal{S}_i^k. \tag{4.14c}$$

In this manner, we reformulate $\mathcal{I}$ into an instance $\mathcal{I}'$ of Min Multiset Multicover with $k|J|$ many constraints and at most $|I||J|^{qk}$ many variables. This means, the instance $\mathcal{I}'$ has a ground set of $k|J|$ many elements and at most $|I||J|^{qk}$ many sets, which is exponential in the input size of $\mathcal{I}$, unless k is fixed. Of course, practically, this reformulation appears to be pointless as we need to encode a tremendous amount of sets. Yet, it is of theoretical interest as we fulfill the requirements of [Dob82] of non-negative and integral data. Thus, if we reduce $\mathrm{m}(j, S_l)$ to ξ_j^l whenever necessary, we may apply the approximation procedure described in Section 4.2.2. Note that each column of the constraint matrix in (4.14) sums up to at most qk. Thus, the solution value returned by the greedy algorithm approximates the optimal value by a factor of $\mathrm{H}(qk)$.

Consider the case of fixed k, where both the number of constraints and the number of variables of (4.14) is polynomially bounded in the input size of $\mathcal{I}$. Here we obtain a constant factor approximation algorithm, i.e., Robust Min q-MSMC with discrete uncertainty is contained in APX, see also Section 3.4.1. On the other hand, in this special case, we can apply the randomized rounding algorithms mentioned in Subsection 4.2.1 and we may obtain improved results. For instance, with [CQ19] we can obtain an approximation algorithm with ratio $\ln k + \ln\ln k + \mathcal{O}(1)$ if $\xi_j^l \geq q$ for all $j \in J$, $l \in \{1, \ldots, k\}$.

For the general case, let us analyze Algorithm 4.2 applied to (4.14). Our aim is to ensure a running time polynomial in $|I|$, $|J|$, and k. Thus, we have to circumvent encoding the constraint matrix as well as the solution vector corresponding to (4.14). Furthermore, for $j \in J$ and $l \in \{1, \ldots, k\}$, we need to ensure implicitly that the entries of the matrix in the row corresponding to ξ_j^l do not exceed the value ξ_j^l.

For iteration $r \in \mathbb{N}_{>0}$ of the algorithm, let $\xi_j^{l,r}$ reflect the number of clients in region j in scenario ξ^l that still need to be covered at the beginning of iteration r. While $\sum_{l=1}^k \xi^{l,r}(J)$ is not zero yet, we seek a location $i \in I$ and an element $S = (S_1, \ldots, S_k) \in \mathcal{S}_i^k$ that together maximize the newly covered demand

$$\sum_{l=1}^{k} \sum_{j \in N(i)} \min\left\{\mathrm{m}(j, S_l), \xi_j^{l,r}\right\}. \tag{4.15}$$

Note that, when choosing $S \in \mathcal{S}_i^k$, the choice of S_l for scenario ξ^l is independent from the choices of $S_{l'}$ for $l' \neq l$, as the inner minimization in (4.15) only depends on the l^{th} coordinate of S, the region j, and the current iteration r. Thus, fixing a location $i \in I$, we get

$$\max_{S \in \mathcal{S}_i^k}\left\{\sum_{l=1}^{k} \sum_{j \in N(i)} \min\left\{\mathrm{m}(j, S_l), \xi_j^{l,r}\right\}\right\} = \sum_{l=1}^{k} \max_{T \in \mathcal{S}_i}\left\{\sum_{j \in N(i)} \min\left\{\mathrm{m}(j, T), \xi_j^{l,r}\right\}\right\}.$$

Lemma 4.27. *Let an iteration $r \in \mathbb{N}_{>0}$, a location $i \in I$, and a scenario $\xi^l \in \mathcal{U}$ be fixed. Then,*

$$\max_{T \in \mathcal{S}_i} \left\{ \sum_{j \in N(i)} \min \left\{ m(j,T), \xi_j^{l,r} \right\} \right\} = \min \left\{ q, \xi^{l,r}(N(i)) \right\} \tag{4.16}$$

and an optimal solution $T^\star \in \mathcal{S}_i$ can be computed in time $\mathcal{O}(|N(i)|)$.

Proof. Let $T \in \mathcal{S}_i$. It holds true that

$$\sum_{j \in N(i)} \min \left\{ \mathrm{m}(j,T), \xi_j^{l,r} \right\} \leq \sum_{j \in N(i)} \mathrm{m}(j,T) \leq q.$$

Therefore, we get that

$$\max_{T \in \mathcal{S}_i} \sum_{j \in N(i)} \min \left\{ \mathrm{m}(j,T), \xi_j^{l,r} \right\} \leq \min \left\{ q, \xi^{l,r}(N(i)) \right\}.$$

We define $T' := \varnothing$ and successively consider the regions $j \in N(i)$. For $j \in N(i)$, we add $\min\{q - |T'|, \xi_j^{l,r}\}$ many copies of region j to T' if $\min\{q - |T'|, \xi_j^{l,r}\} > 0$. If $q \leq \xi^{l,r}(N(i))$, we have $|T'| = q$. Thus, $T' \in \mathcal{S}_i$ and

$$\sum_{j \in N(i)} \min \left\{ \mathrm{m}(j,T'), \xi_j^{l,r} \right\} = \sum_{j \in N(i)} \mathrm{m}(j,T') = q.$$

On the other hand, if $q > \xi^{l,r}(N(i))$, we have $|T'| < q$. To enforce $|T'| = q$ we fix a region $j' \in N(i)$ and additionally add $q - |T'|$ many copies of region j' to T'. Then, $T' \in \mathcal{S}_i$ and we have

$$\sum_{j \in N(i)} \min \left\{ \mathrm{m}(j,T'), \xi_j^{l,r} \right\} = \xi^{l,r}(N(i)),$$

as T' contains at least $\xi_j^{l,r}$ many copies of region j for each $j \in N(i)$. It is easy to see that in both cases we can find the set T' in time $\mathcal{O}(|N(i)|)$. □

Corollary 4.28. *In iteration $r \in \mathbb{N}_{>0}$, the maximization problem*

$$\max_{i \in I} \max_{S \in \mathcal{S}_i^k} \left\{ \sum_{l=1}^{k} \sum_{j \in N(i)} \min \left\{ m(j, S_l), \xi_j^{l,r} \right\} \right\} \tag{4.17}$$

Algorithm 4.3 Approximation algorithm for Robust Min q-MSMC.

Input: An instance $\mathcal{I}$ of Robust Min q-MSMC.
Output: A solution $x \in \mathbb{N}^{|I|}$ to $\mathcal{I}$.

```
x := 0
while ∑_{l=1}^k ξ^l(J) ≥ 1 do
    Compute S = (S_1, ..., S_k) ∈ 𝒮_i^k for some i ∈ I maximizing (4.17).
    x_i := x_i + 1
    ξ_j^l := max{0, ξ_j^l − m(j, S_l)} for all j ∈ J, l ∈ {1, ..., k}
end while
return x
```

can be solved in time $\mathcal{O}(k|E|)$ with solution $S \in \mathcal{S}_{i'}^k$ for some $i' \in I$, where $S = (S_1, \ldots, S_k)$ and, for $l \in \{1, \ldots, k\}$, the set S_l is a solution to (4.16). □

Using Lemma 4.27 there is no need to execute Step 6 of Algorithm 4.2 if applied to an instance of Robust Min q-MSMC. Further, we adapt the algorithm slightly to completely avoid encoding the constraint matrix and the variables corresponding to (4.14). The resulting procedure is shown in Algorithm 4.3. The algorithm acts as applying Algorithm 4.2 to (4.14) with reducing the matrix entries appropriately in a preprocessing step. The only difference is that we do not store the chosen maximizer S in Step 3 of Algorithm 4.3 as we are only interested in the location from which the element originates. In fact, this enables us to avoid encoding the variables of (4.14). Further, we can improve the approximation ratio from $\mathrm{H}(qk)$ to

$$\mathrm{H}\left(\max_{i \in I} \sum_{l=1}^{k} \min\left\{q, \xi^l(N(i))\right\}\right),$$

as by Lemma 4.27 the maximum column sum of the constraint matrix (after adapting its entries) corresponds to

$$\max_{i \in I} \max_{S \in \mathcal{S}_i^k} \left\{ \sum_{l=1}^{k} \sum_{j \in N(i)} \min\left\{\mathrm{m}(j, S_l), \xi_j^l\right\} \right\} = \max_{i \in I} \sum_{l=1}^{k} \min\left\{q, \xi^l(N(i))\right\}.$$

By Corollary 4.28 we can bound the running time of one iteration of Algorithm 4.3 by $\mathcal{O}(k|E|)$. All in all, only the total number of iterations remains to be analyzed. Currently, the algorithm needs at least

$$\left\lceil \frac{\sum_{l=1}^{k} \xi^l(J)}{qk} \right\rceil$$

many iterations. Similar to the idea in [Dob82], we want to limit the number of iteration by a polynomial in $|I|$ and k. Suppose $x_{i'}$ is increased in iteration $r \in \mathbb{N}_{>0}$, i.e., location $i' \in I$ maximizes the value

$$\sum_{l=1}^{k} \min \left\{ q, \xi^{l,r}(N(i)) \right\}$$

over all $i \in I$. For a location $i \in I$, only the scenarios with $\xi^{l,r}(N(i)) > 0$ contribute to this sum. We define the *set of active scenarios* $\mathcal{U}'$ and a *minimum active scenario* $\xi^{l'}$ with respect to i' as follows:

$$\mathcal{U}' \coloneqq \left\{ \xi^l \in \mathcal{U} \colon \xi^{l,r}(N(i')) > 0 \right\} \quad \text{and} \quad \xi^{l'} \in \arg\min_{\xi \in \mathcal{U}'} \left\{ \xi^{l,r}(N(i')) \right\}.$$

If $\xi^{l',r}(N(i')) < q$, then the total demand of $N(i')$ in scenario $\xi^{l'}$ is covered after iteration r, i.e., $\xi^{l',r+1}(N(i')) = 0$. On the other hand, if $\xi^{l',r}(N(i')) \geq q$, then the total demand of *every* scenario $\xi^l \in \mathcal{U}'$ is decreased by q. Thus, for $\xi^l \in \mathcal{U}'$, we have

$$\xi^{l,r+1}(N(i')) = \xi^{l,r}(N(i')) - q \geq 0 \tag{4.18}$$

and $\xi^{l,r+1}(N(i')) = \xi^{l,r}(N(i')) = 0$ for $\xi^l \in \mathcal{U} \setminus \mathcal{U}'$. If $\xi^{l',r+1}(N(i')) \geq q$ still holds, then location i' also maximizes Step 3 of Algorithm 4.3 for iteration $r+1$ as, for $i \in I$, we have

$$\begin{aligned} q \cdot |\mathcal{U}'| = \sum_{l=1}^{k} \min \left\{ q, \xi^{l,r+1}(N(i')) \right\} &= \sum_{l=1}^{k} \min \left\{ q, \xi^{l,r}(N(i')) \right\} \\ \geq \sum_{l=1}^{k} \min \left\{ q, \xi^{l,r}(N(i)) \right\} &\geq \sum_{l=1}^{k} \min \left\{ q, \xi^{l,r+1}(N(i)) \right\}. \end{aligned}$$

Thus, if location i' is chosen, by (4.18) we can directly chose the location

$$\max \left\{ 1, \left\lfloor \frac{\xi^{l',r}(N(i'))}{q} \right\rfloor \right\}$$

many times. Then, if location i' is chosen again in the course of the algorithm, we know that there exists a scenario ξ^l such that the total demand in the neighborhood $N(i')$ for this scenario is covered. Thus, the total number of iterations is bounded by $2k|I|$. Using this implementation we get a total running time of $\mathcal{O}(k^2|I||E|)$ as the running time of one iteration is still dominated by Step 3. Altogether, we have proven the following theorem.

Theorem 4.29. *For any fixed $q \in \mathbb{N}_{>0}$, Algorithm 4.3 is an approximation algorithm for Robust Min q-MSMC with polynomial time enumeration uncertainty. Let $\mathcal{I}$ be such an instance. Algorithm 4.3 runs in time $\mathcal{O}(|\mathcal{U}|^2|I||E|)$ and outputs a solution x to $\mathcal{I}$ which satisfies*

$$x(I) \leq H\left(\max_{i \in I} \sum_{\xi \in \mathcal{U}} \min\left\{q, \xi(N(i))\right\}\right) \cdot z(\mathcal{U})$$

with $z(\mathcal{U})$ being the optimal value of $\mathcal{I}$. □

Observe that

$$\max_{i \in I} \sum_{\xi \in \mathcal{U}} \min\left\{q, \xi(N(i))\right\} \leq \min\left\{q \cdot |\mathcal{U}|, \sum_{\xi \in \mathcal{U}} \xi(J)\right\}.$$

Hence, the ratio is bounded from above by $1 + \ln \sum_{\xi \in \mathcal{U}} \xi(J)$ and by Corollary 3.32 it is essentially optimal up to additive constants in the factor. Furthermore, tightness follows from [Joh74] and the construction in the proof of Theorem 3.30, where we have

$$\max_{i \in I} \sum_{\xi \in \mathcal{U}} \min\left\{q,\ \xi(N(i))\right\} = \max_{i \in I} |N(i)|.$$

Recalling Section 4.1.2, we can make use of a prevailing set for the cases in which the uncertainty set is not polynomial time enumerable, cf. Definition 4.15. Let an instance $\mathcal{I}$ of Robust Min q-MSMC with $m := |J|$ and uncertainty set $\mathcal{U} \subseteq \mathbb{N}^m$ be given. Let $\beta, \beta^1, \ldots, \beta^m$ be some generating vectors corresponding to $\mathcal{U}$, i.e., $\beta^j \in \arg\max_{\xi \in \mathcal{U}} \xi_j$ for $j \in J$ and β is obtained from Algorithm 4.1 applied to $\mathcal{U}$. Consider the prevailing set $\mathcal{U}_0$ of *integral* scenarios

$$\mathcal{U}_0 := \left\{2\beta, \lceil 2\sqrt{m}\rceil \cdot \beta^1, \ldots, \lceil 2\sqrt{m}\rceil \cdot \beta^m\right\}.$$

In a first step, we replace $\mathcal{U}$ by $\mathcal{U}_0$ in the given instance $\mathcal{I}$ and approximate the optimal value $z(\mathcal{U}_0)$ using Algorithm 4.3. Then, we apply Lemma 2.10 and Lemma A.1 saying that

$$z(\mathcal{U}_0) = z(\mathrm{conv}(\mathcal{U}_0)) \leq \left(4\sqrt{m}+1\right) \cdot z(\mathcal{U}),$$

where $z(\mathcal{U})$ is the optimal value of $\mathcal{I}$. Denote by x_0 the output of Algorithm 4.3 applied to $\mathcal{U}_0$ and by $\alpha_{\mathcal{U}_0}$ the corresponding ratio from Theorem 4.29. We obtain

$$x_0(I) \leq \alpha_{\mathcal{U}_0} \cdot z(\mathcal{U}_0) \leq \alpha_{\mathcal{U}_0} \cdot \left(4\sqrt{m}+1\right) \cdot z(\mathcal{U}).$$

In this case, Algorithm 4.3 runs in time $\mathcal{O}(m^2|I||E|)$ provided the uncertainty set $\mathcal{U}_0$ is already given. Note that the vector x_0 is feasible for $\mathcal{I}$ as $\mathrm{conv}(\mathcal{U}_0)$ dominates $\mathcal{U}$ and by Lemma 2.10. It remains to specify the running time to obtain $\mathcal{U}_0$: The scenarios $\beta, \beta^1, \ldots, \beta^m$ can be computed in time $\mathcal{O}(m \cdot T(\mathcal{U}))$, where $T(\mathcal{U})$ is the time needed to solve $\max_{\xi \in \mathcal{U}} c^T \xi$ for any given vector $c \in \mathbb{Q}^m_{\geq 0}$ and $T(\mathcal{U})$ is $\Omega(m)$. Further, the final computation of the scenarios of $\mathcal{U}_0$ needs time $\mathcal{O}(m^2)$. Note that the desired precision of the square root can be assumed fix due to rounding. Altogether, we have:

Theorem 4.30. *For any fixed $q \in \mathbb{N}_{>0}$, there is an approximation algorithm for Robust Min q-MSMC with polynomial time optimization uncertainty. Let $\mathcal{I}$ be such an instance. The algorithm runs in time $\mathcal{O}(|J|^2|I||E| + |J| \cdot T(\mathcal{U}))$ and outputs a solution x to $\mathcal{I}$ which satisfies*

$$x(I) \leq \alpha_{\mathcal{U}_0} \cdot \left(4\sqrt{|J|}+1\right) \cdot z(\mathcal{U}).$$

for some prevailing set $\mathcal{U}_0$ corresponding to $\mathcal{U}$. □

With the notation of Theorem 4.30 we have

$$\alpha_{\mathcal{U}_0} = \mathrm{H}\left(\max_{i \in I} \sum_{\xi \in \mathcal{U}_0} \min\left\{q, \xi(N(i))\right\} \right) \leq \mathrm{H}(q \cdot (m+1)).$$

There are various classes of uncertainty sets for which Theorem 4.30 applies, e.g., budgeted uncertainty, Γ-uncertainty, or integral polyhedral uncertainty in general. Further, recall Theorem 4.25 showing that the computation of $z(\mathcal{U}_0)$ is NP-hard for budgeted uncertainty. Hence, unless P = NP, it is meaningful to apply the concept of domination together with Algorithm 4.3 for budgeted uncertainty. Last but not least, Theorem 4.30 also applies to instances with polynomial time enumerable uncertainty sets. In particular, for instances with $|J|$ small, this approach might lead to an approximation algorithm with faster running time and a comparatively small loss in the approximation guarantee.

Conclusion

In this chapter, we have studied approximations and approximation algorithms for Robust Min q-MSMC by applying two different approaches: the representation of Robust Min q-MSMC as an Min ARC problem and its representation as a Min Multiset Multicover problem.

We have seen that an optimal strictly robust solution is a 2-approximation for positive uncertainty sets. For interval uncertainty $\mathcal{U}_{\mathrm{I}}$, a solution of this type is even optimal by the results of Section 3.4.2. Therefore, for budgeted uncertainty $\mathcal{U}_{\mathrm{B}}$, an optimal strictly robust solution is optimal if $b(J) = \Gamma$ and provides a 2-approximate solution if the vector $1/2 \cdot (a+b)$ is contained in $\mathcal{U}_{\mathrm{B}}^{+}$. Moreover, we have determined the symmetry factor $\mathrm{sym}(\mathcal{U}_{\mathrm{B}}^{+})$ as well as the translation factor $\rho(\mathcal{U}_{\mathrm{B}}^{+})$ to apply a result of [BGS11] showing that

$$z_{\mathrm{rob}}(\mathcal{U}_{\mathrm{B}}) \leq \left\lceil 1 + \frac{\rho(\mathcal{U}_{\mathrm{B}}^{+})}{\mathrm{sym}(\mathcal{U}_{\mathrm{B}}^{+})} \right\rceil \cdot z(\mathcal{U}_{\mathrm{B}}).$$

An optimal strictly robust solution can be computed in polynomial time if $q = 1$ or $q = 2$ and if the vector ξ^{wc} with $\xi_j^{\mathrm{wc}} := \max_{\xi \in \mathcal{U}} \xi_j$ for $j \in J$ can be computed in polynomial time. Hence, we obtain approximation algorithms for Robust Min 1-MSMC and Robust Min 2-MSMC with interval or budgeted uncertainty. Considering the affine policy, we have shown that it leads to suboptimal solutions and that computing an optimal affine solution is NP-hard. Similarly, replacing the uncertainty set $\mathcal{U}$ with a corresponding prevailing set $\mathcal{U}_0$ also leads to NP-hard problems. Therefore, these policies are most likely not applicable efficiently in practice.

Furthermore, we have provided an approximation algorithm for Robust Min q-MSMC with polynomial time enumeration uncertainty that runs in time $\mathcal{O}(|\mathcal{U}|^2|I||E|)$ and outputs a solution which guarantees a ratio of

$$\alpha_{\mathcal{U}} := \mathrm{H}\left(\max_{i\in I}\sum_{\xi\in\mathcal{U}}\min\left\{q,\xi(N(i))\right\}\right),$$

that is essentially best possible up to additive constants in the factor.

Moreover, we presented an approximation algorithm for Robust Min q-MSMC with polynomial time optimization uncertainty that has a running time of $\mathcal{O}(|J|^2|I||E| + |J|\cdot T(\mathcal{U}))$ and outputs a solution guaranteeing a ratio of $\alpha_{\mathcal{U}_0}\cdot\left(4\sqrt{|J|}+1\right)$, where $\mathcal{U}_0$ is a prevailing set corresponding to the given uncertainty set $\mathcal{U}$ and $\alpha_{\mathcal{U}_0}\leq \mathrm{H}(q\cdot(|J|+1))$.

5. Including Behavior Patterns of Clients

In the previous chapters, we have dedicated our analysis to the problem Robust Min q-MSMC in which the clients have no influence on the choice of the location they are served by. Thus, they are completely dependent on the decision maker and their own interests are not taken into account in any way. Motivated by this imbalance, we introduce (partly) independent clients in this chapter who are able to enforce their preferences. We realize this empowerment of the clients by specifying different behavior patterns for the clients depending on their ability to influence the solution. For each such pattern, we analyze the non-robust as well as the robust variant of the emerging problem. In particular, we highlight the implications for different specific classes of uncertainty sets. As the different behavior patterns lead to structurally different optimization problems, we include related literature into the respective sections.

In Section 5.1, we begin with the most independent form of a behavior pattern in which each client is allowed to determine the location by which they want to be covered. As the decision maker does not know the choices of the clients in advance, they must implement a solution that is capable of handling all possible distributions of clients. With this set-up, the optimal solution can be stated with low expenditure. Hence, we narrow the power of the clients in Section 5.2 by limiting the set of available locations. That means, in a first step, the decision maker decides on a set of available locations. The independent clients need to adapt to these changes as they can only be served by the locations chosen by the decision maker. But still, multiple locations need to prepare for the demand of a single region. Hence, in Section 5.3, we aim to decrease the responsibilities of the locations. For the clients of a region, we demand a common preference order over their adjacent locations. Now, the decision maker decides on a set of available locations and each client is served by their most preferred location.

Some parts of this chapter also appear in [Büs+21; Com21; Str21]. Specifically, in [Büs+21], a combination of steerable and unsteerable clients is considered in the context of mobile medical units. This publication extends the Robust Min q-MSMC problem in a practical application by including clients with preferences, cf. Sections 5.3. Altogether, the content of this chapter is joint work together with Christina Büsing, Martin Comis, and Manuel Streicher. Based on this cooperation, both Martin Comis and Manuel Streicher also have chapters in their theses that associate with the findings presented here, see [Com21; Str21].

5.1. Min q-Free Clients

We begin with the consideration of completely free clients in a non-robust setting. That means, the clients themselves can exactly determine the location by which they want to be served. Moreover, the decision maker is not aware of this decision in advance. The only information for the decision maker is whether the clients of a region consider a location as a possible service location or not. As in the previous chapters, the number of clients within a region is represented as the demand of the region. Thus, the decision maker needs to provide sufficiently many suppliers in the locations for every possible combination of choices of the clients. Formally, for a fixed integer $q \in \mathbb{N}_{>0}$, this leads to the following problem:

Problem 5.1 (q-Free Clients (q-FC)).
Instance: Finite sets I, J with $I \cap J = \varnothing$, a demand value $d_j \in \mathbb{N}_{>0}$ for each $j \in J$, a bipartite graph $G = (I \cup J, E)$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for all $y \in \mathbb{N}^{|I|\times|J|}$ with $\sum_{i\in N(j)} y_{ij} = d_j$ for $j \in J$, it holds that

$$\sum_{j\in N(i)} y_{ij} \leq q \cdot x_i \text{ for } i \in I?$$

We refer to the optimization version of q-FC as Min q-Free Clients (Min q-FC). From the above definition, the value $y_{ij} \in \mathbb{N}$ can be interpreted as the number of clients in region j that want to be served by location i. Thus, a matrix $y \in \mathbb{N}^{|I|\times|J|}$ with the above properties represents one possible combination of choices of the clients. Then, a solution $x \in \mathbb{N}^{|I|}$ is in that sense robust against these choices as it provides enough suppliers in each location, no matter how the clients decide. Further, the instance $\mathcal{I}$ leads to a corresponding instance $\mathcal{I}'$ of Min q-MSMC as the input data coincides and the feasibility condition for $\mathcal{I}'$ carries over, cf. Observation 3.5. Therefore, the feasibility of a given instance can be verified in time $\mathcal{O}(|J|)$ and, from now on, we only consider feasible instances. The subsequent theorem shows that finding an optimal solution for $\mathcal{I}$ can be accomplished in linear time.

Theorem 5.2. *For any fixed $q \in \mathbb{N}_{>0}$, the optimal solution $x^\star$ to an instance of Min q-FC is given by*

$$x_i^\star := \left\lceil \frac{d(N(i))}{q} \right\rceil$$

for every location $i \in I$ and can be computed in time $\mathcal{O}(|I| + |E|)$.

Proof. Fix a location $i \in I$. For every region $j \notin N(i)$, choose an arbitrary location $i_j \in N(j)$ and consider the matrix $y^i \in \mathbb{N}^{|I|\times|J|}$ defined as follows: For $i' \in I$ and $j \in J$, let

$$y^i_{i'j} := \begin{cases} d_j, & \text{if } j \in N(i) \text{ and } i' = i, \\ d_j, & \text{if } j \notin N(i) \text{ and } i' = i_j, \\ 0, & \text{else.} \end{cases} \tag{5.1}$$

Then, $\sum_{i' \in N(j)} y^i_{i'j} = d_j$ for every $j \in J$ and $\sum_{j \in N(i)} y^i_{ij} = d(N(i))$. Thus, any solution x must satisfy

$$x_i \geq \left\lceil \frac{d(N(i))}{q} \right\rceil = x_i^\star \tag{5.2}$$

for every location i. Now, let $y \in \mathbb{N}^{|I|\times|J|}$ with $\sum_{i \in N(j)} y_{ij} = d_j$ for all $j \in J$ be given. For every region j and every location $i \in N(j)$, we must have $y_{ij} \leq d_j$. Then, $\sum_{j \in N(i)} y_{ij} \leq d(N(i)) \leq q \cdot x_i^\star$ for every $i \in I$ implying feasibility of $x^\star$. Optimality of $x^\star$ is then given by (5.2) and we see that $x^\star$ is the only optimal solution. The running time follows by initializing $x^\star$ and considering each edge exactly once. □

The interpretation of the optimal solution $x^\star$ is evident: Each location prepares for the worst-case scenario, meaning the greatest possible number of clients they need to serve.

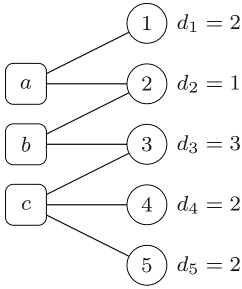


Figure 5.1.: Bipartite graph G corresponding to Example 5.3.

Example 5.3. Recall Example 3.4 where $q = 2$. The constructed bipartite graph G is given once again in Figure 5.1. In the new setting, the given vector $\bar{x} = (\bar{x}_a, \bar{x}_b, \bar{x}_c)^T = (1, 2, 2)^T$ is not feasible anymore as location a might need to serve three clients at once. Similarly, locations b and c have to prepare for at most four and seven clients, respectively. Thus, the optimal solution $x^\star$ of this instance of q-FC is $x_a^\star := 2$, $x_b^\star := 2$, and $x_c^\star := 4$ and in total we need three additional suppliers compared to the optimal value of 5 of the corresponding instance of q-MSMC. $\triangleleft$

5.1.1. Including Uncertainty

As this non-robust variant of free clients can be solved in linear time, we now include some uncertainty in the demand of each region. Similar to Robust q-MSMC, we introduce an uncertainty set $\mathcal{U}$ of possible demand scenarios. We will see that this substantially changes the complexity of the problem.

Problem 5.4 (Robust q-Free Clients (Robust q-FC)).
Instance: Finite sets I, J with $I \cap J = \varnothing$, an uncertainty set $\mathcal{U} \subseteq \mathbb{N}^{|J|}$, a bipartite graph $G = (I \cup J, E)$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for all $\xi \in \mathcal{U}$ and for all $y \in \mathbb{N}^{|I| \times |J|}$ with $\sum_{i \in N(j)} y_{ij} = \xi_j$ for $j \in J$, it holds that

$$\sum_{j \in N(i)} y_{ij} \leq q \cdot x_i \text{ for } i \in I?$$

Now, for a scenario $\xi \in \mathcal{U}$, the value $y_{ij} \in \mathbb{N}$ represents the number of clients of region j that want to be served by location i in case scenario ξ occurs. Thus, in this robust situation, the decision maker has to prepare for any combination of choices of the clients no matter which scenario reveals. When looking for a solution with minimum value of $x(I)$, we call the problem Robust Min q-Free Clients (Robust Min q-FC). Again, we restrict ourselves to feasible instances of Robust Min q-Free Clients since feasibility can be checked in time $\mathcal{O}(|J|)$, cf. Observation 3.28. Let $\mathcal{I}$ be such an instance. By the same argumentation as in the proof of Theorem 5.2, we get that the unique optimal solution $x^\star$ to $\mathcal{I}$ is given by

$$x_i^\star := \left\lceil \frac{\max_{\xi \in \mathcal{U}} \xi(N(i))}{q} \right\rceil \tag{5.3}$$

for every location $i \in I$ with the analogous interpretation. By (5.3) computing $x_i^\star$ for some $i \in I$ requires to solve an instance of Max Robust Sum. This leads to the subsequent hardness result.

Theorem 5.5. *For any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-FC is* NP*-hard.*

Proof. Let an instance of Robust Sum be given, i.e., an uncertainty set $\mathcal{U} \subseteq \mathbb{N}^n$, a set $S \subseteq \{1, \ldots, n\}$, and an integer $B \in \mathbb{N}_{>0}$. By Lemma 3.22 we know that this problem is NP-complete. In fact, we can even assume that B is a multiple of q by Remark 3.23. Thus, let $B = q \cdot B'$ for some $B' \in \mathbb{N}_{>0}$. We construct two instances of Robust Min q-FC. To that end, we set $I := \{0\}$, $J := S$, and $J' := S \cup \{n+1\}$ and define the complete bipartite graphs $G = (I \cup J, E)$ and $G' = (I \cup J', E')$. Let $\mathcal{U}' := \{(\xi, 1) : \xi \in \mathcal{U}\} \subseteq \mathbb{N}^{n+1}$. Now, let $\mathcal{I}$ and $\mathcal{I}'$ denote the instances of Robust Min q-FC with bipartite graphs G and G' and uncertainty sets $\mathcal{U}$ and $\mathcal{U}'$, respectively. Further, let $\bar{x}_0$ and $\bar{x}_0'$ denote the optimal solutions to $\mathcal{I}$ and $\mathcal{I}'$, respectively. Note that $\bar{x}_0 \leq \bar{x}_0' \leq \bar{x}_0 + 1$. Then, we get

$$q \cdot (\bar{x}_0 - 1) < \max_{\xi \in \mathcal{U}} \xi(S) \leq q \cdot \bar{x}_0$$

and

$$q \cdot (\bar{x}_0' - 1) < \max_{\xi \in \mathcal{U}} \xi(S) + 1 \leq q \cdot \bar{x}_0'.$$

Thus, if $B' \leq \bar{x}_0 - 1$, we know that the given instance of Robust Sum is a yes-instance. If $B' > \bar{x}_0$, the instance is a no-instance. Further, if

$B' = \bar{x}_0 = \bar{x}'_0$, the instance is a no-instance. Finally, if $B' = \bar{x}_0 = \bar{x}'_0 - 1$, the instance is a yes-instance. Thus, having an oracle for Robust Min q-FC, we can decide whether the instance of Robust Sum is a yes-instance or not. □

Observe that we did not show APX-hardness in Theorem 5.5 as the reduction requires optimal solutions to the constructed instances of Robust Min q-FC and not only the transfer of solutions. Certainly, computing $x^\star$ by (5.3) mainly depends on the structure of the uncertainty set. In general, we cannot rely on such a structure so that we propose the following integer program for Robust Min q-FC:

$$
\begin{aligned}
\mathrm{P}_{\mathrm{FC}}(\mathcal{U}) \quad & \min_{x} \quad \sum_{i \in I} x_i && && (5.4\mathrm{a})\\
& \text{s.t.} \quad \sum_{j \in N(i)} \xi_j \le q \cdot x_i && \text{for } i \in I, \xi \in \mathcal{U} && (5.4\mathrm{b})\\
& \qquad\qquad\quad x_i \in \mathbb{N} && \text{for } i \in I. && (5.4\mathrm{c})
\end{aligned}
$$

The correctness of formulation $\mathrm{P}_{\mathrm{FC}}(\mathcal{U})$ is immediate. If Max Robust Sum can be solved efficiently on $\mathcal{U}$, the constraints (5.4b) can be replaced with $\max_{\xi \in \mathcal{U}} \xi(N(i)) \le q \cdot x_i$ for $i \in I$ to regain a compact formulation. If this is not the case, we can apply constraint generation starting with an initially empty uncertainty set $\mathcal{U}'$, cf. Section 2.5. If $\bar{x} \in \mathbb{N}^{|I|}$ is optimal for $\mathrm{P}_{\mathrm{FC}}(\mathcal{U}')$, we need to check whether there exists a scenario $\xi \in \mathcal{U}$ and a location $i \in I$ such that $\xi(N(i)) > q \cdot \bar{x}_i$. Thus, we need to solve $|I|$ many instances of Max Robust Sum which is, in general, a hard task due to Lemma 3.22.

If, e.g., the uncertainty set $\mathcal{U}$ is polynomial time enumerable, the instance of Robust Min q-FC can be solved in polynomial time. For example, this applies to discrete uncertainty:

Theorem 5.6. *For any fixed $q \in \mathbb{N}_{>0}$, the optimal solution $x^\star$ to an instance of Robust Min q-FC with discrete uncertainty is given by*

$$x_i^\star := \left\lceil \frac{\max_{\xi \in \mathcal{U}} \xi(N(i))}{q} \right\rceil$$

for $i \in I$ and can be computed in time $\mathcal{O}(|I| + |E|)$. □

Also for interval, budgeted, and Γ-uncertainty introduced in Section 3.4 we obtain positive results as Max Robust Sum restricted to theses classes of uncertainty sets can be solved efficiently by Lemma 3.62 and Lemma 3.76. Moreover, we can directly state the optimal solution.

Theorem 5.7. *For any fixed $q \in \mathbb{N}_{>0}$, the optimal solution $x^\star$ to an instance of Robust Min q-FC with interval uncertainty is given by*

$$x_i^\star := \left\lceil \frac{b(N(i))}{q} \right\rceil$$

for $i \in I$ and can be computed in time $\mathcal{O}(|I| + |E|)$. □

Theorem 5.8. *For any fixed $q \in \mathbb{N}_{>0}$, the optimal solution $x^\star$ to an instance of Robust Min q-FC with budgeted uncertainty is given by*

$$x_i^\star := \left\lceil \frac{\min\{b(N(i)), \Gamma - a(J \setminus N(i))\}}{q} \right\rceil$$

for $i \in I$ and can be computed in time $\mathcal{O}(|I| + |E|)$. □

Theorem 5.9. *For any fixed $q \in \mathbb{N}_{>0}$, the optimal solution $x^\star$ to an instance of Robust Min q-FC with Γ-uncertainty is given by*

$$x_i^\star := \left\lceil \frac{a(N(i)) + \max\{\hat{a}(T) \colon T \subseteq N(i), |T| \leq \Gamma\}}{q} \right\rceil$$

for $i \in I$ and can be computed in time $\mathcal{O}(|I| + |E|)$.

Proof. For the running time, we refer to the proof of Lemma 3.76. □

On the contrary, Robust Sum is NP-complete for multi-budgeted as well as ellipsoidal uncertainty by Lemma 3.67 and Theorem 3.72. In both cases, this remains true if we restrict to instances whose bound is a multiple of q for some fixed $q \in \mathbb{N}_{>0}$, cf. Remark 3.74. For that reason, Robust Min q-FC restricted to these classes of uncertainty sets is still NP-hard. In both cases, the proof is analogous to that of Theorem 5.5.

Theorem 5.10. *For any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-FC with multi-budgeted uncertainty is* NP*-hard.* □

Theorem 5.11. *For any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-FC with ellipsoidal uncertainty is* NP*-hard.* □

As given by (5.3), the clients completely determine the optimal solution to Robust Min q-FC while the locations do not have any scope of action. To change this situation slightly we allow locations to be closed in the following. We see that this admission drastically increases the complexity of the problem already for the non-robust variant.

5.2. Min q-Adapting Clients

In this section, we consider free clients as in Section 5.1. But now, the decision maker is allowed to select only a subset of the locations to be opened whereas, in Section 5.1, at any time, every location needs to prepare for clients. Thus, the set of available locations for each client is restricted and the clients have to adapt to these circumstances. Naturally, we require at least one open location for every client. This leads to the following decision problem for a fixed positive integer $q \in \mathbb{N}_{>0}$.

Problem 5.12 (q-Adapting Clients (q-AC))**.**
Instance: Finite sets I, J with $I \cap J = \varnothing$, a demand value $d_j \in \mathbb{N}_{>0}$ for each $j \in J$, a bipartite graph $G = (I \cup J, E)$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for every $j \in J$, there is $i \in N(j)$ with $x_i \geq 1$ and, for all $y \in \mathbb{N}^{|I| \times |J|}$ with $\sum_{i \in N(j)\colon x_i \geq 1} y_{ij} = d_j$ for $j \in J$, it holds that

$$\sum_{j \in N(i)} y_{ij} \leq q \cdot x_i \text{ for } i \in I \text{ with } x_i \geq 1?$$

The problem of minimizing $x(I)$ in instances of q-AC is named Min q-Adapting Clients (Min q-AC). Let $\mathcal{I}$ be an instance of this problem. Then, as in Section 5.1 the instance $\mathcal{I}$ has a feasible solution if and only if we have $N(j) \neq \varnothing$ for every region $j \in J$. As this condition can be checked in time $\mathcal{O}(|J|)$, we restrict our analysis to feasible instances. Let $I' := \{i \in I\colon x_i \geq 1\}$ be the set of *open* locations for a given solution x to $\mathcal{I}$. We call a location $i \notin I'$ *closed*. Similar to Section 5.1, for $i \in I'$ and $j \in J$, the value $y_{ij} \in \mathbb{N}$ can be interpreted as the number of clients in region j that want to be served by location $i \in I'$. For every $i \in I'$, consider the vector y^i which is defined analogously to (5.1) with the adaption that, for $j \notin N(i)$, the location i_j is now chosen from $N(j) \cap I'$. Then, we must have $\sum_{j \in N(i)} y^i_{ij} = d(N(i)) \leq q \cdot x_i$ since x is feasible. As in the proof of Theorem 5.2 setting

$$x'_i = \left\lceil \frac{d(N(i))}{q} \right\rceil \tag{5.5}$$

if $i \in I'$ and zero otherwise leads to another solution x' with $x'(I) \leq x(I)$. Given $I' \subseteq I$, we can compute the solution x' in time $\mathcal{O}(|I| + |E|)$. Thus, an optimal solution to the instance $\mathcal{I}$ is determined only by the choice of open locations.

Definition 5.13 (Feasible Locations)**.** Let an instance of q-AC be given. A subset $I' \subseteq I$ such that, for every $j \in J$, there is $i \in N(j) \cap I'$ is called *feasible.*

Let x be feasible for an instance of q-AC. The corresponding set of open locations is denoted by $I_x := \{i \in I \colon x_i \geq 1\}$. Then, I_x is feasible. Let us consider a small example.

Example 5.14. We reconsider Example 3.4 in this new setting. Observe that locations a and c need to be opened as otherwise regions 1 and 5 cannot be covered. As $\{a, c\}$ is feasible, we directly get that $x^\star_a := 2$, $x^\star_b := 0$, and $x^\star_c := 4$ is an optimal solution to this instance of Min q-Adapting Clients. Compared to the optimal values of 5 and 8 of the corresponding instances of Min q-MSMC and Min q-FC, respectively, we see that this variant leads to a compromise solution, cf. Example 5.3. ◁

For a feasible set $I' \subseteq I$, the corresponding solution defined by (5.5) is called the *solution induced by* I'. Due to the previous argumentation, it suffices to restrict our considerations to feasible sets of locations and their induced solutions. Hence, Min q-AC can be formulated as the following integer program.

$$\mathrm{P}_{\mathrm{AC}}(d) \quad \min_{z} \quad \sum_{i \in I} \left\lceil \frac{d(N(i))}{q} \right\rceil \cdot z_i \tag{5.6a}$$

$$\text{s.t.} \quad \sum_{i \in N(j)} z_i \geq 1 \quad \text{for } j \in J \tag{5.6b}$$

$$z_i \in \mathbb{N} \quad \text{for } i \in I. \tag{5.6c}$$

Due to (5.6b) and the objective function, we can assume that $z_i \in \mathbb{B}$ for every $i \in I$. Hence, the variable z_i indicates whether location $i \in I$ is contained in the set of open locations or not. For a solution $\bar{z} \in \mathbb{B}^{|I|}$ to $\mathrm{P_{AC}}(d)$, the set $\{i \in I\colon \bar{z}_i = 1\}$ is feasible by (5.6b). In any case, the vector $z' := 1$ is another solution and we get that $\mathrm{P_{AC}}(d)$ has a finite optimal solution [Sch98; Mey74]. Additionally, this program reveals that we are actually dealing with a special weighted Set Cover problem: We need to find a feasible set of locations I' such that

$$\sum_{i \in I'} \left\lceil \frac{d(N(i))}{q} \right\rceil$$

is minimum, cf. Section 2.4. Here, the weight of a set $N(i)$ for $i \in I$ solely depends on the elements contained in the set. Observe that, if $d(N(i)) \leq q$ for every $i \in I$, we obtain an instance of the classical Min Set Cover problem and all known solution techniques for this problem are applicable. This close relation leads to the subsequent result. Recall that Min 3-Dimensional Cover is APX-complete by Theorem 3.55.

Theorem 5.15. *For any fixed $q \in \mathbb{N}_{>0}$, Min q-AC is* APX*-hard.*

Proof. Let an instance $\mathcal{I}$ of Min 3-Dimensional Cover be given, i.e., a collection $\mathcal{C} \subseteq W \times X \times Y$ with disjoint sets W, X, and Y. We construct a bipartite graph $G = (I \cup J, E)$ analogously to the proof of Theorem 3.57: Let $I := \mathcal{C}$, $J := W \cup X \cup Y$ and $N((w, x, y)) := \{w, x, y\}$ for all $(w, x, y) \in \mathcal{C}$. Furthermore, we set $d_j := q$ for every $j \in J$. Note that, for every $i \in I$,

$$\left\lceil \frac{d(N(i))}{q} \right\rceil = 3.$$

This constitutes our instance $\mathcal{I}'$ of Min q-AC. Given an optimal cover $\mathcal{C}'$ for $\mathcal{I}$, we set $z_i := 1$ if $i \in \mathcal{C}'$ and zero otherwise. Then, $x := 3z$ is feasible for $\mathcal{I}'$ and we have $\mathrm{OPT}(\mathcal{I}') \leq 3 \cdot \mathrm{OPT}(\mathcal{I})$. On the other hand, given a solution x to $\mathcal{I}'$, we can assume that x is of the form $3z$ for some $z \in \mathbb{B}^{|I|}$. Then, by feasibility of x, the vector z encodes a solution to $\mathcal{I}$ and we get $3 \cdot \mathrm{SOL}(\mathcal{I}, z) \leq \mathrm{SOL}(\mathcal{I}', x)$ so that $3 \cdot (\mathrm{SOL}(\mathcal{I}, z) - \mathrm{OPT}(\mathcal{I})) \leq \mathrm{SOL}(\mathcal{I}', x) - \mathrm{OPT}(\mathcal{I}')$. In particular, this also gives $3 \cdot \mathrm{OPT}(\mathcal{I}) \leq \mathrm{OPT}(\mathcal{I}')$. □

Corollary 5.16. *For any fixed $q \in \mathbb{N}_{>0}$, q-AC is strongly* NP*-complete.*

Proof. Given a vector $x \in \mathbb{N}^{|I|}$ with $x_i \leq d(N(i))$ for $i \in I$, we can check in polynomial time whether I_x is feasible and whether $q \cdot x_i \geq d(N(i))$ for every $i \in I_x$. Thus, the problem is contained in NP. NP-hardness follows from the proof of Theorem 5.15 as a solution to $\mathcal{I}$ with value $B \in \mathbb{N}$ leads to a solution to $\mathcal{I}'$ with value $3B$ in polynomial time and vice versa. $\square$

5.2.1. Including Uncertainty

The above complexity results show that a polynomial time algorithm for Min q-AC is very unlikely. Unfortunately, this does not change when including robustness. For the sake of completeness, we nevertheless state the problem and the consequent findings.

Problem 5.17 (Robust q-Adapting Clients (Robust q-AC)).
Instance: Finite sets I, J with $I \cap J = \varnothing$, an uncertainty set $\mathcal{U} \subseteq \mathbb{N}^{|J|}$, a bipartite graph $G = (I \cup J, E)$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for every $j \in J$, there is $i \in N(j)$ with $x_i \geq 1$ and, for all $\xi \in \mathcal{U}$ and for all $y \in \mathbb{N}^{|I| \times |J|}$ with $\sum_{i \in N(j)\colon x_i \geq 1} y_{ij} = \xi_j$ for $j \in J$, it holds that

$$\sum_{j \in N(i)} y_{ij} \leq q \cdot x_i \text{ for } i \in I \text{ with } x_i \geq 1?$$

Again, for a solution $x \in \mathbb{N}^{|I|}$ and a scenario $\xi \in \mathcal{U}$, the value $y_{ij} \in \mathbb{N}$ for $i \in I_x$ and $j \in J$ describes the number of clients of region j that want to be served by location i in case scenario ξ occurs. By the same argumentation as for the non-robust case, we restrict ourselves to feasible instances. Then, an optimal solution to an instance of the corresponding minimization problem Robust Min q-Adapting Clients (Robust Min q-AC) is given by a feasible set of locations $I' \subseteq I$ such that

$$\sum_{i \in I'} \left\lceil \frac{\max_{\xi \in \mathcal{U}} \xi(N(i))}{q} \right\rceil$$

is minimum. Thus, the *(robust) solution induced by* I' is given by

$$x_i := \left\lceil \frac{\max_{\xi \in \mathcal{U}} \xi(N(i))}{q} \right\rceil$$

if $i \in I'$ and zero otherwise and as before it suffices to only consider solutions of this type. By Theorem 5.15 and Corollary 5.16 we directly get the subsequent result.

Theorem 5.18. *For any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-AC is* APX*-hard and its decision version is strongly* NP*-hard.* □

As in the proof of Theorem 5.5 we get that computing the solution induced by some feasible set $I' \subseteq I$ is NP-hard. For completeness, we also state the integer program corresponding to Robust Min q-AC.

$$
\begin{aligned}
\mathrm{P}_{\mathrm{AC}}(\mathcal{U}) \quad & \min_{x,\,z} \quad \sum_{i \in I} x_i && && \text{(5.7a)}\\
& \text{s.t.} \quad \sum_{i \in N(j)} z_i \geq 1 && \text{for } j \in J && \text{(5.7b)}\\
& \quad z_i \cdot \sum_{j \in N(i)} \xi_j \leq q \cdot x_i && \text{for } i \in I, \xi \in \mathcal{U} && \text{(5.7c)}\\
& \quad x_i, z_i \in \mathbb{N} && \text{for } i \in I. && \text{(5.7d)}
\end{aligned}
$$

Again, we can assume $z \in \mathbb{B}^{|I|}$ and the variable vector z encodes the chosen feasible set of locations. If, for a solution $(\bar{x}, \bar{z})$ and some $i \in I$, $\bar{z}_i = 0$, we can assume that also $\bar{x}_i = 0$ as (5.7c) is redundant for i and every scenario $\xi \in \mathcal{U}$ and we aim to minimize $\bar{x}(I)$. On the other hand, if $\bar{z}_i = 1$ for some $i \in I$, we must have $\xi(N(i)) \leq q \cdot \bar{x}_i$ for every $\xi \in \mathcal{U}$. Therefore, in an optimal solution $(x^\star, z^\star)$ to $\mathrm{P}_{\mathrm{AC}}(\mathcal{U})$, the vector $x^\star$ is the solution induced by $\{i \in I \colon z_i^\star = 1\}$ and $x^\star$ is optimal for the given instance of Robust Min q-AC. This shows the correctness of the formulation.

As for Robust Min q-Free Clients, the constraints (5.7c) can be replaced by $z_i \cdot \max_{\xi \in \mathcal{U}} \xi(N(i)) \leq q \cdot x_i$ for $i \in I$. Hence, this gives a polynomial number of constraints if $\max_{\xi \in \mathcal{U}} \xi(N(i))$ can be solved efficiently for every $i \in I$. Otherwise, we can apply constraint generation to the constraints (5.7c) starting with a (potentially empty) subset $\mathcal{U}'$ of the scenarios, cf. Section 2.5. Let $(\bar{x}, \bar{z})$ be feasible for $\mathrm{P}_{\mathrm{AC}}(\mathcal{U}')$. In the separation step, we seek a scenario $\xi \in \mathcal{U}$ and a location $i \in I$ with $\bar{z}_i = 1$ such that $\xi(N(i)) > q \cdot \bar{x}_i$. As in Section 5.1, this leads to solving instances of Max Robust Sum.

Every class of uncertainty sets we consider in this thesis can be forced to contain only one single scenario, cf. Section 2.6. Hence, Theorem 5.18 also applies to Robust Min q-AC restricted to instances of a specific class of uncertainty sets. At least, for discrete, interval, budgeted, and Γ-uncertainty, given a feasible set $I' \subseteq I$, we can obtain the corresponding induced solution in polynomial time, cf. Section 5.1. Furthermore, for these classes, formulation (5.7) can be condensed to obtain a compact formulation as mentioned above. Hence, in these cases, we even obtain that the corresponding decision problems are contained in NP and, thus, are NP-complete with Theorem 5.18.

Altogether, ensuring this degree of freedom to the clients while passing only little information on the clients' behavior to the decision maker leads to solutions in which various locations need to provide suppliers for the demand of a single region. In the following section, we consider another compromise behavior pattern in which, in any solution, there is, for every region, exactly one responsible location.

5.3. Min q-Ordered Clients

In this section, we showcase a behavior pattern of the clients that increases the level of information for the decision maker and decreases the responsibilities of the locations. So, we aim for a trade-off between the self-determination of the clients and the provision of supply by the locations. We assume that every region has a globally known preference order over its set of adjacent locations. Clients within a region share the preference order and the region's cardinality sets up its demand. To formalize this requirement, let us define a general preference order over a finite set.

Definition 5.19 (Preference Order)**.** Given a finite set S, a *preference order* over S is a bijection $\sigma\colon \{1, \dots, |S|\} \to S$.

Hence, a preference order σ over S induces a *linear order* of the elements of S [Sch02]. We interpret the preference order as $\sigma(1)$ being the most preferred element and $\sigma(|S|)$ being the least preferred element. For better readibility, the function σ is often given as a tuple $(\sigma(1), \dots, \sigma(|S|)) \in S^{|S|}$. Observe that the preference order is *strict* as σ is a bijection, i.e., for any two distinct elements of S, we can decide which element is preferred over which.

Thus, having a bipartite graph $G = (I \cup J, E)$ and, for every region $j \in J$, a preference order $\sigma_j\colon \{1, \dots, |N(j)|\} \to N(j)$, we assume that the decision maker selects a feasible set of locations $I' \subseteq I$ in a first step, cf.

Definition 5.13. In a subsequent step, the clients of each region demand to be served by their most preferred location in I'. As the orders are strict, the actions of the clients are unique and the decision maker cannot influence their behavior. Such two-stage problems are a special type of bilevel optimization problems, cf. [VKK09]. For a fixed integer $q \in \mathbb{N}_{>0}$, this leads to the following problem definition.

Problem 5.20 (q-Ordered Clients (q-OC))**.**
Instance: Finite sets I, J with $I \cap J = \varnothing$, a demand value $d_j \in \mathbb{N}_{>0}$ for each $j \in J$, a bipartite graph $G = (I \cup J, E)$, a preference order σ_j over $N(j)$ for each $j \in J$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for every $j \in J$, there is $i \in N(j)$ with $x_i \geq 1$ and, for $y \in \mathbb{B}^{|I| \times |J|}$ defined by

$$y_{ij} := \begin{cases} 1, & \text{if } i \in \arg\min \left\{ \sigma_j^{-1}(i') \colon i' \in N(j) \wedge x_{i'} \geq 1 \right\}, \\ 0, & \text{otherwise,} \end{cases}$$

for $i \in I$, $j \in J$, it holds true that $\sum_{j \in N(i)} d_j \cdot y_{ij} \leq q \cdot x_i$ for $i \in I$?

The optimization version of this problem is referred to as Min q-Ordered Clients (Min q-OC) and, again, feasibility of an instance is given as in Observation 3.5. Hence, from now on, we only consider feasible instances. Let $x \in \mathbb{N}^{|I|}$ be feasible for an instance of q-OC with corresponding feasible set of locations $I_x := \{i \in I \colon x_i \geq 1\}$. Then, fixing $i \in I$ and $j \in J$, the entry y_{ij} in Problem 5.20 is set to 1 if the location i is contained in $N(j) \cap I_x$ and i is the most preferred location of region j among the open locations I_x. Otherwise, the entry is set to zero. Note that this choice is unique so that $\sum_{i \in I} y_{ij} = 1$ for every $j \in J$. Further, the definition of y depends on I_x only and not on the specific values of x. Therefore, we define:

Definition 5.21 (Responsibility, Active Neighborhood)**.** Let an instance of q-OC be given and let $I' \subseteq I$ be feasible.

(a) For a region $j \in J$, the unique location $i \in N(j) \cap I'$ with $\sigma_j^{-1}(i) \leq \sigma_j^{-1}(i')$ for all $i' \in N(j) \cap I'$ is called *responsible* for region j with respect to I'.

(b) For a location $i \in I$, the set of regions location i is responsible for (with respect to I') is called *active neighborhood* of location i and denoted by $N'(i)$, i.e., for $i \in I'$, we have

$$N'(i) := \left\{ j \in N(i) \colon \sigma_j^{-1}(i) \leq \sigma_j^{-1}(i') \text{ for all } i' \in N(j) \cap I' \right\}$$

and $N'(i) := \varnothing$ for $i \in I \setminus I'$.

Observe that $\dot{\bigcup}_{i \in I} N'(i) = J$. Using Definition 5.21 we obtain an equivalent reformulation of the matrix y in Problem 5.20.

Definition 5.22 (Assignment Matrix)**.** Let $I' \subseteq I$ be feasible for an instance of q-OC. The *assignment matrix* $y(I') \in \mathbb{B}^{|I| \times |J|}$ is given by

$$y(I')_{ij} := \begin{cases} 1, & \text{if location } i \text{ is responsible for region } j \text{ with respect to } I', \\ 0, & \text{otherwise,} \end{cases}$$

for $i \in I, j \in J$.

Thus, the matrix y in Problem 5.20 corresponds to the assignment matrix $y(I_x)$. In the following, we show that it suffices to consider feasible subsets of the locations similar to Section 5.2.

Lemma 5.23. *Let an instance of Min q-OC be given and let $I' \subseteq I$ be feasible. Define $x' \in \mathbb{N}^{|I|}$ by*

$$x'_i := \left\lceil \frac{d(N'(i))}{q} \right\rceil \tag{5.8}$$

for $i \in I$. Then, the vector x' is feasible and $x'(I) \leq x(I)$ for every solution x with $y(I_x) = y(I')$.

Proof. Consider the assignment matrix $y(I')$ and note that, by definition of x', we have $I_{x'} \subseteq I'$ and $y(I') = y(I_{x'})$. For every location $i \in I$, we have

$$\sum_{j \in N(i)} d_j \cdot y(I')_{ij} = d(N'(i)) \leq q \cdot x'_i$$

implying feasibility of x'. Now, let $x \in \mathbb{N}^{|I|}$ be feasible with $y(I_x) = y(I')$. Then, we have that $d(N'(i)) \leq q \cdot x_i$ for every $i \in I$. Hence, $x'(I) \leq x(I)$ by definition of x'. $\square$

As in Section 5.2 we call x' defined by (5.8) the *solution induced by* I'. Observe that we have $I_{x'} \subseteq I'$ since a location in I' might not be responsible for any adjacent region. All in all, by Lemma 5.23 solving an instance $\mathcal{I}$ of Min q-OC reduces to computing a feasible set $I' \subseteq I$ such that

$$\sum_{i \in I} x'_i = \sum_{i \in I'} \left\lceil \frac{d(N'(i))}{q} \right\rceil \tag{5.9}$$

is minimum. Observe the slight difference to (5.5) due to the active neighborhood $N'(i)$. Given $I' \subseteq I$, the corresponding induced solution value can be computed in time $\mathcal{O}(|I| + |E|)$. Thus, from now on we restrict ourselves to feasible subsets of the locations and their induced solutions. As in Section 5.2, the whole set of locations I is trivially feasible so that $\mathcal{I}$ has a finite optimal solution. Let us consider some examples.

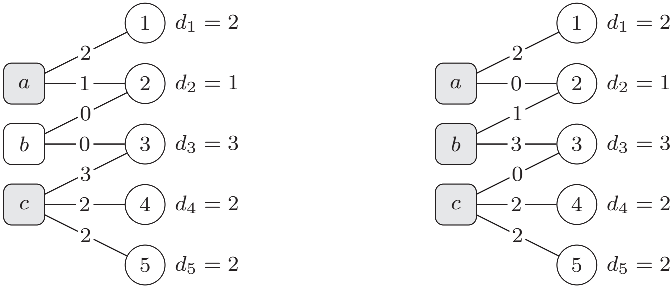
Example 5.24. Recall Example 3.4 where $q = 2$ and $d = (2, 1, 3, 2, 2)^T$. To obtain an instance of q-OC we define the following preference orders: $\sigma_1 := (a)$, $\sigma_2 := (b, a)$, $\sigma_3 := (b, c)$, and $\sigma_4 := \sigma_5 := (c)$. As in Example 5.14 fix the feasible set $\{a, c\} \subseteq I$ which leads to the assignment matrix

$$y(\{a, c\}) = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \end{pmatrix},$$

where the first, second, and third row correspond to the locations a, b, and c, respectively. Then, the value $d_j \cdot y(\{a, c\})_{ij}$ is given as an edge label on edge $[i, j]$ in Figure 5.2a. Therefore, any solution x with corresponding feasible set $\{a, c\}$ needs to satisfy $x_a \geq 2$ and $x_c \geq 4$ so that the best possible solution value under these assumptions is 6. On the other hand, Figure 5.2b shows the value $d_j \cdot y(I)_{ij}$ on edge $[i, j]$ in case of opening all locations. Here, we have

$$y(I) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

and we can set $x'_a := 1$, $x'_b := 2$, and $x'_c := 2$ to obtain a solution x' with value 5. Note that this is also the optimal value of the corresponding instance of Min q-MSMC where we disregard the preference orders. $\triangleleft$



(a) Open locations a and c. (b) Open all locations.

Figure 5.2.: Bipartite graph G with two possible choices of open locations corresponding to Example 5.24.

We have seen that opening the least possible number of locations does not necessarily lead to an optimal solution. Due to the preference orders the task is to open a feasible set of locations $I' \subseteq I$ such that

$$\begin{aligned}\sum_{i \in I'} \left\lceil \frac{d(N'(i))}{q} \right\rceil &= \sum_{i \in I'} \frac{d(N'(i))}{q} + \sum_{i \in I'} \left(\left\lceil \frac{d(N'(i))}{q} \right\rceil - \frac{d(N'(i))}{q} \right) \\ &= \frac{d(J)}{q} + \sum_{i \in I'} \left(\left\lceil \frac{d(N'(i))}{q} \right\rceil - \frac{d(N'(i))}{q} \right)\end{aligned}$$

is minimized. Thus, we aim to minimize the total sum of rounding errors and opening a new location might change the assignment matrix in a way that these errors decrease. In a second example, we see an instance of Min q-OC in which the optimal value of its corresponding instance of Min q-MSMC is strictly better.

Example 5.25. Consider the setting in Example 5.24 and reverse the preference order of region 2, i.e., $\sigma_2 := (a, b)$. Analogously to Example 5.24, opening locations a and c leads to an induced solution with solution value 6. On the other hand, when opening all locations the corresponding induced solution is given by $x_a := x_b := x_c := 2$ with solution value 6 as well. Hence, the optimal value of this instance is 6. By Example 3.4 the optimal value of the corresponding instance of Min q-MSMC is 5. $\triangleleft$

With our new interpretation of the problem we now argue why instances of Min q-OC with unit demands are of particular interest. To that end, let $T(n)$ be the time needed to solve an instance of Min q-OC with unit demands and encoding length $n \in \mathbb{N}$. Naturally, we assume that $T\colon \mathbb{N} \to \mathbb{N}$ is monotonically increasing.

Lemma 5.26. *Let an instance $\mathcal{I}$ of Min q-OC be given with encoding length $n \in \mathbb{N}$. The optimal solution to $\mathcal{I}$ can be found in time $\mathcal{O}(|E| + T(qn))$.*

Proof. Consider the demand values $d_j \in \mathbb{N}_{>0}$ of the given instance $\mathcal{I}$. For every $j \in J$, there exist unique values $s_j \in \mathbb{N}$ and $r_j \in \{1, \dots, q\}$ such that $d_j = s_j \cdot q + r_j$. These values can be computed in time $\mathcal{O}(|J|)$. Let $I' \subseteq I$ be feasible. We reconsider the solution value of the solution induced by I':

$$\begin{aligned}
\sum_{i \in I'} \left\lceil \frac{d(N'(i))}{q} \right\rceil &= \sum_{i \in I'} \left\lceil \frac{\sum_{j \in N'(i)} s_j \cdot q + r_j}{q} \right\rceil \\
&= \sum_{i \in I'} \left(s(N'(i)) + \left\lceil \frac{r(N'(i))}{q} \right\rceil \right) \\
&= s(J) + \sum_{i \in I'} \left\lceil \frac{r(N'(i))}{q} \right\rceil .
\end{aligned} \tag{5.10}$$

As $s(J)$ does not depend on I', we can find a solution to $\mathcal{I}$ by considering the corresponding instance $\mathcal{I}'$ in which the demand value $d_j \in \mathbb{N}_{>0}$ is replaced by $r_j \in \mathbb{N}_{>0}$ for every $j \in J$. Given an optimal solution x' to $\mathcal{I}'$ we simply add, for every region j, s_j many suppliers to the location i which is responsible for j with respect to $I_{x'}$. This can be done in $\mathcal{O}(|E|)$. By (5.10) the so constructed solution is optimal for $\mathcal{I}$. To obtain an instance with unit demands, we simply copy every region $j \in J$ exactly r_j many times and connect each copy with $N(j)$. Hence, each region and each edge needs to be copied at most q times. The encoding length of this instance is at most qn and every optimal solution to it is also optimal for $\mathcal{I}'$ and vice versa. The running time follows as $|J| \leq |E|$. □

Corollary 5.27. *If $T(n)$ is $\mathcal{O}(n^k)$ for some fixed $k \in \mathbb{N}$, an instance of Min q-OC with encoding length $n \in \mathbb{N}$ can be solved in time $\mathcal{O}(n^k)$.*

Proof. As q is fixed, $T(qn)$ is $\mathcal{O}((qn)^k) = \mathcal{O}(n^k)$. Further, for the edge set E of the given bipartite graph we have $|E| \leq n$. Thus, the result follows from Lemma 5.26. □

Corollary 5.27 implies that a polynomial time algorithm for an instance of Min q-OC with unit demands leads to a polynomial time algorithm for general instances. Thus, we now focus on the existence of such algorithms by starting with the case $q = 1$. As the following algorithm is as simple as it can get, we refrain from only considering unit demands and directly handle the general case.

Proposition 5.28. *Min 1-OC can be solved in time $\mathcal{O}(|I| + |J|)$.*

Proof. In every solution to an instance of Min 1-OC, each client needs their own supplier. Thus, we initialize $x := 0$ and, for every region $j \in J$, we add d_j suppliers to location $\sigma_j(1)$. This leads to an optimal solution in time $\mathcal{O}(|I| + |J|)$. ☐

Unfortunately, polynomial time solvability for cases other than $q = 1$ is very unlikely as we see in the following. First of all, we show that there exists a constant factor approximation algorithm for Min q-OC.

Lemma 5.29. *For any fixed $q \in \mathbb{N}_{>0}$, Min q-OC is contained in* APX.

Proof. Let $I' \subseteq I$ be feasible for an instance $\mathcal{I}$ of Min q-OC with induced solution x'. Then, we get

$$\begin{aligned} x'(I) &\leq \sum_{j \in J} \left\lceil \frac{d_j}{q} \right\rceil \leq \frac{d(J)}{q} + \sum_{j \in J} \frac{q-1}{q} = \frac{d(J)}{q} + \frac{q-1}{q} \cdot |J| \\ &\leq \mathrm{OPT}(\mathcal{I}) + (q-1) \cdot \mathrm{OPT}(\mathcal{I}) = q \cdot \mathrm{OPT}(\mathcal{I}) \end{aligned}$$

as $d_j \geq 1$ for all $j \in J$ and hence $\mathrm{OPT}(\mathcal{I}) \geq {}^{|J|}/_{q}$. Thus, the solution $\bar{x}$ induced by the feasible set $\{\sigma_j(1) \colon j \in J\}$ leads to a q-approximation. As $\bar{x}$ can be computed in time $\mathcal{O}(|I| + |E|)$ and q is fixed the claim follows. ☐

Theorem 5.30. *Min 2-Ordered Clients is* APX*-complete even for unit demands.*

Proof. By Lemma 5.29 we get that Min 2-Ordered Clients can be approximated within a ratio of 2. To show APX-hardness, we present an L-reduction from the APX-complete problem Max 3-SAT(3), where each variable appears at most three times and each clause has at most three literals. Therefore, recall Problem 3.53 and Assumption 3.54. We start by constructing an instance of Min 2-Ordered Clients containing regions with

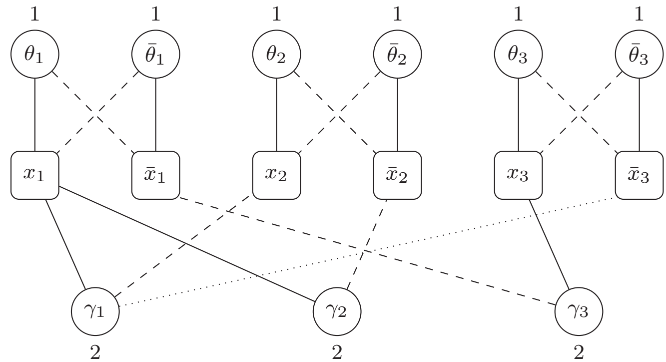


Figure 5.3.: Construction of instance $\mathcal{I}'$ in the proof of Theorem 5.30 for $C_1 = X_1 \vee X_2 \vee \overline{X}_3$, $C_2 = X_1 \vee \overline{X}_2$, and $C_3 = X_3 \vee \overline{X}_1$. A solid edge is preferred to a dashed edge which itself is preferred to a dotted edge. The demand of each region is shown as a vertex label.

demand 2. Nevertheless, at the end of the proof, we see how to adapt this instance to obtain an instance with unit demands. Thus, given an instance $\mathcal{I}$ of Max 3-SAT(3) with $m \in \mathbb{N}_{>0}$ clauses $C_1, \ldots, C_m$ and $n \in \mathbb{N}_{>0}$ variables $X_1, \ldots, X_n$, we define the following Min 2-Ordered Clients instance $\mathcal{I}'$:

Procedure 5.31. For each literal $L \in \left\{X_i, \overline{X}_i \colon i \in \{1, \ldots, n\}\right\}$, there is a location, i.e., $I \coloneqq \{x_1, \bar{x}_1, \ldots, x_n, \bar{x}_n\}$. As with literals, for a location $l \in I$, we write $\bar{l}$ for its corresponding "negated" location. For each $i \in \{1, \ldots, n\}$, there are two regions θ_i and $\bar{\theta}_i$ which are both adjacent to x_i and $\bar{x}_i$. Region θ_i prefers x_i to $\bar{x}_i$ while region $\bar{\theta}_i$ prefers $\bar{x}_i$ to x_i. Furthermore, for each clause C_j, there is a region γ_j which is adjacent to every location l whose corresponding literal is contained in C_j. Thus, the regions are

$$J \coloneqq J_\theta \cup J_{\bar{\theta}} \cup J_\gamma \coloneqq \{\theta_1, \ldots, \theta_n\} \cup \left\{\bar{\theta}_1, \ldots, \bar{\theta}_n\right\} \cup \{\gamma_1, \ldots, \gamma_m\}.$$

If $C_j = L_1 \vee L_2 \vee L_2$ with corresponding locations $l_1, l_2, l_3 \in I$, we use the same order for the preference order of region γ_j, i.e., $\sigma_{\gamma_j} = (l_1, l_2, l_3)$. We proceed analogously if $|C_j| \in \{1, 2\}$.

The demand of the regions in $J_\theta \cup J_{\bar{\theta}}$ is set to 1 while the regions in J_γ have demand 2. In total, constructing the instance $\mathcal{I}'$ can be done in polynomial time. An example of the constructed instance is illustrated in Figure 5.3.

Now, given a truth assignment f for $\mathcal{I}$ fulfilling exactly $r \in \mathbb{N}$ clauses, we obtain a solution to $\mathcal{I}'$ with the following procedure.

Procedure 5.32. We start by opening a location $l \in I$ if the corresponding literal L is evaluated to TRUE. Further, for each unsatisfied clause C_j, we open the location l corresponding to the first literal appearing in the clause.

Hence, each region has an open location in its neighborhood and the set of open locations is feasible. By the choice of the preference orders, every open location must have one supplier who serves one or both clients of the regions in $J_\theta \cup J_{\bar{\theta}}$, i.e., these sum up to at most $n + m - r$ many. Furthermore, for every region $\gamma_j \in J_\gamma$, there is one supplier in some adjacent location covering its demand of 2. Thus, the induced solution value is bounded by $n + (m - r) + m = n + 2m - r$. By Assumption 3.54 we obtain

$$\begin{aligned} \mathrm{OPT}(\mathcal{I}') &\leq n + 2m - \mathrm{OPT}(\mathcal{I}) \qquad (5.11) \\ &\overset{(3.17)}{\leq} \frac{3}{2}m + 2m - \mathrm{OPT}(\mathcal{I}) = \frac{7}{2}m - \mathrm{OPT}(\mathcal{I}) \\ &\overset{(3.16)}{\leq} 7 \cdot \mathrm{OPT}(\mathcal{I}) - \mathrm{OPT}(\mathcal{I}) = 6 \cdot \mathrm{OPT}(\mathcal{I}). \end{aligned}$$

On the other hand, given a solution $I' \subseteq I$ to the instance $\mathcal{I}'$ of Min 2-Ordered Clients, we obtain a truth assignment f as follows:

Procedure 5.33. For a variable X_i, we set $f(X_i) := $ TRUE if the corresponding location x_i is open *and* location $\bar{x}_i$ is closed. If location x_i is closed but $\bar{x}_i$ is open, we set $f(X_i) :=$ FALSE. If there is a variable X_i such that both locations x_i and $\bar{x}_i$ are open, we set $f(X_i) :=$ TRUE if $|N(x_i)| \geq |N(\bar{x}_i)|$ and $f(X_i) :=$ FALSE otherwise.

This means, if $x_i \in I'$ and $\bar{x}_i \in I'$, the variable X_i is TRUE if the corresponding literal X_i appears at least as often in the given formula as its negation $\overline{X}_i$, otherwise the variable X_i is set to FALSE. By construction we have, for every $i \in \{1, \ldots, n\}$, $x_i \in I'$ or $\bar{x}_i \in I'$ so that the definition of f is complete. Let $r \in \mathbb{N}$ be the number of variables such that both locations x_i and $\bar{x}_i$ are open. Then, we have $\mathrm{SOL}(\mathcal{I}', I') \geq n + m + r$. As I' is feasible for $\mathcal{I}'$, we get that, for every clause C_j, there is at least one open location l

such that the corresponding literal L is contained in C_j. If location $\bar{l}$ is closed, then clause C_j is satisfied by the constructed truth assignment f. If location $\bar{l}$ is open, then clause C_j is satisfied if L appears more often than $\overline{L}$ in the given formula. If L appears at most as often as $\overline{L}$, then L appears exactly once by our Assumption 3.54. Thus, C_j is the only clause containing L or $\overline{L}$ that is potentially not satisfied by f. Therefore, we have $\mathrm{SOL}(\mathcal{I}, f) \geq m - r$ as whenever both x_i and $\bar{x}_i$ are open and we choose one of the corresponding literals to be TRUE by the prescribed rule, at most once clause containing X_i or $\overline{X_i}$ is not satisfied. Altogether, we get

$$\begin{aligned}
\mathrm{OPT}(\mathcal{I}) - \mathrm{SOL}(\mathcal{I}, f) &\leq \mathrm{OPT}(\mathcal{I}) - m + r \\
&= n + m + r - (n + 2m - \mathrm{OPT}(\mathcal{I})) \\
&\leq \mathrm{SOL}(\mathcal{I}', I') - (n + 2m - \mathrm{OPT}(\mathcal{I})) \\
&\overset{(5.11)}{\leq} \mathrm{SOL}(\mathcal{I}', I') - \mathrm{OPT}(\mathcal{I}').
\end{aligned}$$

Finally, by replacing each γ_j for $j \in J$ in the construction with two of its copies with demand 1 each we obtain the desired result. $\square$

Corollary 5.34. *2-Ordered Clients is strongly* NP*-complete even for unit demands.*

Proof. Given a subset $I' \subseteq I$, it can be checked in polynomial time whether I' is feasible and whether the solution value of the corresponding induced solution x' satisfies the given bound B. Observe that, for $i \in I$, x'_i is bounded by $d(N(i))$. Further, the proof of Theorem 5.30 shows that a solution to $\mathcal{I}$ with at least $r \in \mathbb{N}$ satisfied clauses leads to a solution to $\mathcal{I}'$ with at most $n + 2m - r$ suppliers in polynomial time and vice versa. $\square$

In particular, Theorem 5.30 and Corollary 5.34 show that including preference orders for the regions leads to a new problem with different structural properties compared to Min q-MSMC. Last but not least, we briefly analyze the case $q \geq 3$. Here, we can recover previous proofs.

Theorem 5.35. *For any fixed $q \geq 3$, Min q-OC is* APX*-complete and its decision version is strongly* NP*-complete even for unit demands.*

Proof. Due to Lemma 5.29 and the proof of Corollary 5.34 we get that Min q-OC is contained in APX and q-OC is in NP. Let $q \geq 3$ be fixed. In the proof of Theorem 3.14, we can define a preference order σ_s for every

$s \in S$, e.g., by the order of increasing indices i_C for $i_C \in N(s)$. Then, we can recover the remainder of the proof as, for any feasible $I' \subseteq I$ and any location i_C, we have $|N'(i_C)| \leq |N(i_C)| \leq 3 \leq q$. This yields APX-hardness of Min q-OC and NP-hardness of q-OC follows directly. $\square$

Observe that the reduction used in the proof of Theorem 5.35 reveals that, if $q \geq d(N(i))$ for every $i \in I$, Min q-OC is a Min Set Cover(q) problem as the preference orders are not relevant then. Hence, as for Min q-AC, all known solution methods for Min Set Cover can be applied in this case.

In general, to solve an instance $\mathcal{I}$ of Min q-OC with $q \geq 2$ we propose a mixed integer program. In literature, preference orders are a common feature in facility location problems [HP87; VK10; Cán+07; Koc11]. We refer to [HP87] for an overview of possible formulations of preference orders. Consider the following program for $\mathcal{I}$:

$$
\begin{aligned}
\mathrm{P_{OC}}(d) \quad \min_{w,x,y} \quad & \sum_{i \in I} x_i && && (5.12\text{a}) \\
\text{s.t.} \quad & \sum_{i \in N(j)} w_i \geq 1 && \text{for } j \in J && (5.12\text{b}) \\
& \sum_{j \in N(i)} d_j \cdot y_{ij} \leq q \cdot x_i && \text{for } i \in I && (5.12\text{c}) \\
& w_i - \sum_{k=1}^{\sigma_j^{-1}(i)-1} w_{\sigma_j(k)} \leq y_{ij} && \text{for } j \in J, i \in N(j) && (5.12\text{d}) \\
& y_{ij} \geq 0 && \text{for } i \in I, j \in J && (5.12\text{e}) \\
& w_i, x_i \in \mathbb{N} && \text{for } i \in I. && (5.12\text{f})
\end{aligned}
$$

We show that there exists an optimal solution $(w^\star, x^\star, y^\star)$ to $\mathrm{P_{OC}}(d)$ with $w^\star \in \mathbb{B}^{|I|}$ such that the variables can be interpreted in the following way: $I^\star := \{i \in I \colon w_i^\star = 1\}$ is feasible with induced solution $x^\star$ which is optimal for $\mathcal{I}$ and $y^\star = y(I^\star)$. With this interpretation, we see that (5.12b) enforces feasibility of $I^\star$ and (5.12c) together with the optimality of the considered solution ensure that $x^\star$ is induced by $I^\star$. The constraints (5.12d) are subject of the next lemma.

Lemma 5.36. *Let (w, x, y) be feasible for $P_{OC}(d)$. Then, there is a solution (w', x', y') with $w' \in \mathbb{B}^{|I|}$, $x'(I) \leq x(I)$, and the property that, for*

$I' \coloneqq \{i \in I \colon w'_i = 1\}$, x' is the corresponding induced solution and $y' = y(I')$. Given (w, x, y), the vectors w', x', and y' can be computed in time $\mathcal{O}(|I||J|)$.

Proof. First of all, consider the vector $w \in \mathbb{N}^{|I|}$ and fix $j \in J$. By (5.12b) there is some $i \in N(j)$ with $w_i \geq 1$. Let

$$i_j \in \arg\min \left\{\sigma_j^{-1}(i) \colon i \in N(j) \wedge w_i \geq 1\right\}$$

and note that i_j is unique. Thus, by (5.12d) and (5.12e) it holds true that $y_{i_j j} \geq w_{i_j} \geq 1$ and $y_{ij} \geq 0$ for $i \in I \setminus \{i_j\}$. We set $I' \coloneqq \{i_j \colon j \in J\}$ and $w'_i = 1$ if $i \in I'$ and zero otherwise. Then, (w', x, y) is feasible for $\mathrm{P_{OC}}(d)$ and with (5.12b) we get that I' is feasible. Consider a region $j \in J$ and a location $i \in N(j)$. If location i is responsible for region j with respect to I', it holds true that

$$w'_i = 1 \quad \text{and} \quad \sum_{k=1}^{\sigma_j^{-1}(i)-1} w'_{\sigma_j(k)} = 0.$$

Otherwise there is some location $i' \in N(j)$ with $w'_{i'} = 1$ and $\sigma_j^{-1}(i') < \sigma_j^{-1}(i)$ and location i is not responsible for j. Hence, (5.12d) reduces to $y_{ij} \geq 1$. If location i is not responsible for region j with respect to I' we have

$$w'_i = 0 \quad \text{or} \quad \sum_{k=1}^{\sigma_j^{-1}(i)-1} w'_{\sigma_j(k)} \geq 1.$$

In every case, (5.12d) is redundant and (5.12e) enforces $y_{ij} \geq 0$. Thus, setting $y' \coloneqq y(I')$ leads to a solution (w', x, y') with $y' \leq y$. Moreover, by (5.12c), for every location $i \in I$, it holds true that

$$d(N'(i)) = \sum_{j \in N(i)} d_j \cdot y'_{ij} \leq \sum_{j \in N(i)} d_j \cdot y_{ij} \leq q \cdot x_i.$$

Thus, setting x' to be the solution induced by I' leads to a solution (w', x', y') with $x'(I) \leq x(I)$. The vector w' can be computed in time $\mathcal{O}(|E|)$. As x' can be computed from w' in time $\mathcal{O}(|E|)$ and y' can be computed from w' in time $\mathcal{O}(|I||J|)$, we obtain the claimed running time. □

Corollary 5.37. *Let an instance $\mathcal{I}$ of Min q-OC be given. A vector $x \in \mathbb{N}^{|I|}$ is optimal for $\mathcal{I}$ if and only if there are $w \in \mathbb{N}^{|I|}$ and $y \in \mathbb{R}_{\geq 0}^{|I| \times |J|}$ such that (w, x, y) is optimal for $P_{OC}(d)$.*

Proof. Let $x' \in \mathbb{N}^{|I|}$ be optimal for $\mathcal{I}$. With no loss of generality, we can assume that x' is the induced solution of a feasible set $I' \subseteq I$. Define, for every $i \in I$, $w'_i := 1$ if $i \in I'$ and zero otherwise, as well as, for every $i \in I$ and every $j \in J$, $y'_{ij} := 1$ if location i is responsible for region j with respect to I' and zero otherwise. Analogously to the proof of Lemma 5.36, we get that (w', x', y') is feasible for $\mathrm{P_{OC}}(d)$. Now, let (w, x, y) be some arbitrary solution to $\mathrm{P_{OC}}(d)$. By Lemma 5.36 we can assume that $w \in \mathbb{B}^{|I|}$ and that $I'' := \{i \in I \colon w_i = 1\}$ is feasible for $\mathcal{I}$ with induced solution x. By optimality of x' we must have $x'(I) \leq x(I)$. Thus, the solution (w', x', y') is optimal for $\mathrm{P_{OC}}(d)$.

Now, let (w', x', y') be optimal for $\mathrm{P_{OC}}(d)$. By Lemma 5.36 we can assume that $w' \in \mathbb{B}^{|I|}$, $I' := \{i \in I \colon w'_i = 1\}$ is feasible for $\mathcal{I}$ with induced solution x', and $y' = y(I')$. Let $I'' \subseteq I$ be an arbitrary solution to $\mathcal{I}$ with induced solution x''. Setting w'' and y'' analogously to above leads to a solution (w'', x'', y'') to $\mathrm{P_{OC}}(d)$ and we must have $x'(I) \leq x''(I)$ by optimality of (w', x', y'). Hence, the solution x' is optimal for $\mathcal{I}$. $\square$

All in all, we have shown that $\mathrm{P_{OC}}(d)$ is a valid formulation for Min q-OC. Thus, with the help of an MIP-solver we are now able to compute optimal solutions to instances of Min q-OC.

5.3.1. Including Uncertainty

After the study of Min q-OC we now introduce robustness by considering uncertain demands of the regions. Introducing an uncertainty set to capture these uncertainties leads to the following problem for some fixed $q \in \mathbb{N}_{>0}$:

Problem 5.38 (Robust q-Ordered Clients (Robust q-OC)).
Instance: Finite sets I, J with $I \cap J = \varnothing$, an uncertainty set $\mathcal{U} \subseteq \mathbb{N}^{|J|}$, a bipartite graph $G = (I \cup J, E)$, a preference order σ_j over $N(j)$ for each $j \in J$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for every $j \in J$, there is $i \in I$ with $x_i \geq 1$ and the assignment matrix $y := y(I_x) \in \mathbb{B}^{|I| \times |J|}$ satisfies

$$\sum_{j \in N(i)} \xi_j \cdot y_{ij} \leq q \cdot x_i \text{ for } i \in I, \xi \in \mathcal{U}?$$

For the sake of completeness, let Robust Min q-Ordered Clients (Robust Min q-OC) be the minimization problem corresponding to Robust q-OC. As before we restrict our considerations to feasible instances of these problems. In comparison to Problem 5.20, now the locations need to provide suppliers that are capable of covering the demand no matter which scenario occurs. Observe that, given an instance $\mathcal{I}$ of Robust Min q-Ordered Clients, we cannot assume that $\xi > 0$ for every $\xi \in \mathcal{U}$ compared to the non-robust case. Nevertheless, we assume that, for every region $j \in J$, there is a scenario ξ with $\xi_j > 0$, cf. Section 2.6. Therefore, we must have

$$\mathrm{OPT}(\mathcal{I}) \geq \frac{|J|}{q \cdot |\mathcal{U}|} \tag{5.13}$$

since there exists a scenario with at least ${}^{|J|}/_{|\mathcal{U}|}$ many non-zero entries. As for the non-robust variant we can show that concentrating on feasible subsets $I' \subseteq I$ suffices.

Lemma 5.39. *Let an instance of Robust Min q-OC be given and let $I' \subseteq I$ be feasible. Define $x' \in \mathbb{N}^{|I|}$ by*

$$x'_i := \left\lceil \frac{\max_{\xi \in \mathcal{U}} \xi(N'(i))}{q} \right\rceil \tag{5.14}$$

for $i \in I$. Then, the vector x' is feasible and $x'(I) \leq x(I)$ for every solution x with $y(I_x) = y(I')$.

Proof. Fix $\xi \in \mathcal{U}$ and consider the assignment matrix $y(I') = y(I_{x'})$. Then, for every location $i \in I$, we have $\sum_{j \in N(i)} \xi_j \cdot y(I')_{ij} = \xi(N'(i)) \leq q \cdot x'_i$. As this holds for every $\xi \in \mathcal{U}$, we get that x' is feasible. Now, let $x \in \mathbb{N}^{|I|}$ be feasible with $y(I_x) = y(I')$. Then, we have $\xi(N'(i)) \leq q \cdot x_i$ for every $i \in I$ and every $\xi \in \mathcal{U}$. Hence, $x'(I) \leq x(I)$ by definition of x'. $\square$

We refer to x' defined by (5.14) as the *(robust) solution induced by I'*. Again, we have $I_{x'} \subseteq I'$ and solving an instance $\mathcal{I}$ of Robust Min q-OC reduces to finding a feasible set of locations $I' \subseteq I$ such that

$$\sum_{i \in I} x'_i = \sum_{i \in I'} \left\lceil \frac{\max_{\xi \in \mathcal{U}} \xi(N'(i))}{q} \right\rceil$$

is minimum. As the uncertainty set $\mathcal{U}$ is finite and I itself is feasible, we know that the instance $\mathcal{I}$ has a finite optimal solution.

Observation 5.40. After having introduced all behavior patterns we briefly note that the optimal values of corresponding instances are related. Let $\mathcal{I}_{\text{OC}}$ be an instance of Robust Min q-OC and let $\mathcal{I}$, $\mathcal{I}_{\text{FC}}$, and $\mathcal{I}_{\text{AC}}$ be the corresponding instances of Robust Min q-MSMC, Robust Min q-FC, and Robust Min q-AC, respectively, where we disregard the preference orders. Then, we have

$$\text{OPT}(\mathcal{I}) \leq \text{OPT}(\mathcal{I}_{\text{OC}}) \leq \text{OPT}(\mathcal{I}_{\text{AC}}) \leq \text{OPT}(\mathcal{I}_{\text{FC}}).$$

This relation is informally already intended due to the motivation of the behavior patterns in the beginning of each section. More formally, it can readily be seen that the optimal solution to $\mathcal{I}_{\text{FC}}$ is feasible for $\mathcal{I}_{\text{AC}}$ as the whole set of locations I is feasible. Hence, $\text{OPT}(\mathcal{I}_{\text{AC}}) \leq \text{OPT}(\mathcal{I}_{\text{FC}})$. Moreover, we can assume that an optimal solution x^{AC} to $\mathcal{I}_{\text{AC}}$ is induced by some feasible set $I^{\text{AC}} \subseteq I$. Then, the set I^{AC} leads to an induced solution x for $\mathcal{I}_{\text{OC}}$ with $x(I) \leq \text{OPT}(\mathcal{I}_{\text{AC}})$ as, for each location $i \in I$, the number of necessary suppliers is bounded from above by x_i^{AC}. Therefore, $\text{OPT}(\mathcal{I}_{\text{OC}}) \leq \text{OPT}(\mathcal{I}_{\text{AC}})$. Finally, an optimal feasible set $I^{\text{OC}} \subseteq I$ with induced solution x^{OC} for $\mathcal{I}_{\text{OC}}$ leads to a feasible solution $(x^{\text{OC}}, \bar{y})$ for $\mathcal{I}$ by setting $\bar{y}(\xi)_{ij} := \xi_j \cdot y(I^{\text{OC}})_{ij}$ for $i \in I$, $j \in J$, and $\xi \in \mathcal{U}$. Hence, $\text{OPT}(\mathcal{I}) \leq \text{OPT}(\mathcal{I}_{\text{OC}})$.

In contrast to the non-robust version, computing the induced solution from a given feasible set $I' \subseteq I$ corresponds to solving instances of Max Robust Sum and we cannot expect to do this in polynomial time, cf. Theorem 5.5. Therefore, we obtain the subsequent result which essentially makes use of the same ideas as the proof of Theorem 3.30.

Theorem 5.41. *For any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-OC is* APX*-hard and its decision version is strongly* NP*-hard.*

Proof. We show that Min Dominating Set(3) L-reduces to Robust Min q-OC. Let an instance $\mathcal{I}$ of Min Dominating Set(3) be given, i.e., a simple graph $G = (V, E)$ with $V = \{1, \ldots, n\}$ for $n \in \mathbb{N}_{>0}$ and $\Delta_G \leq 3$. To construct an instance $\mathcal{I}'$ of Robust Min q-OC we refer to the proof of Theorem 3.30. Then, we have $I = V$, $J = \{n+1, \ldots, 2n\}$, and $G' = (I \cup J, E')$. Further, we get $\mathcal{U} = \{0, e_1, \ldots, e_n\} \subseteq \mathbb{N}^n$. For the preference orders we let region $j \in J$ prefer $i_1 \in N_{G'}(j)$ over $i_2 \in N_{G'}(j)$ if $i_1 < i_2$.

Let $V' \subseteq V = I$ be an optimal solution to $\mathcal{I}$. Define $I' := V' \subseteq I$. As V' is a dominating set in G and by construction of the bipartite graph G', we get that, for every $j \in J$, there is at least one location $i \in N_{G'}(j) \cap I'$. Then, the induced solution x' is given by

$$x'_i = \left\lceil \frac{\max_{\xi \in \mathcal{U}} \xi(N'(i))}{q} \right\rceil \leq \left\lceil \frac{1}{q} \right\rceil = 1 \tag{5.15}$$

for $i \in I$. Note that $x'_i = 0$ for $i \notin I'$. Thus, setting $x_i := 1$ for $i \in I'$ and zero otherwise yields a solution to $\mathcal{I}'$ so that $\mathrm{OPT}(\mathcal{I}') \leq x(I) = \mathrm{OPT}(\mathcal{I})$.

Now, let $x \in \mathbb{N}^{|I|}$ be a solution to $\mathcal{I}'$. By Lemma 5.39 we can assume that x is induced by some feasible set $I' \subseteq I$ and analogously to (5.15) it holds that $x_i \leq 1$ for every $i \in I$. We define $V' := \{v \in V : x_v = 1\}$ and claim that V' is a dominating set for G. To that end, select a vertex $u \in V$. As $e_u \in \mathcal{U}$ and by feasibility of x, we know that there is $v \in N_{G'}(u)$ with $x_v = 1$, i.e., $v \in V'$. By construction of G', either $v = u$ or the vertices u and v are adjacent in G. Therefore, $\mathrm{SOL}(\mathcal{I}, V') \leq \mathrm{SOL}(\mathcal{I}', x)$ and $\mathrm{SOL}(\mathcal{I}, V') - \mathrm{OPT}(\mathcal{I}) \leq \mathrm{SOL}(\mathcal{I}', x) - \mathrm{OPT}(\mathcal{I}')$. □

Using the inapproximability result of Min Dominating Set due to [DS14; Mos15], we obtain analogously to Corollary 3.32:

Corollary 5.42. *For any fixed $q \in \mathbb{N}_{>0}$ and for any fixed $\varepsilon > 0$, Robust Min q-OC cannot be approximated within a factor of $(1 - \varepsilon)\ln |J|$ or $(1 - \varepsilon)\ln \sum_{\xi \in \mathcal{U}} \xi(J)$ unless $\mathrm{P} = \mathrm{NP}$.* □

By the previous results an integer programming formulation is meaningful to be able to compute a solution to an instance of Robust Min q-OC. Hence, we extend $\mathrm{P_{OC}}(d)$ and obtain:

$$\begin{aligned}
\mathrm{P_{OC}}(\mathcal{U}) \quad \min_{w,x,y} \quad & \sum_{i \in I} x_i && && (5.16a)\\
\text{s.t.} \quad & \sum_{i \in N(j)} w_i \geq 1 && \text{for } j \in J && (5.16b)\\
& \sum_{j \in N(i)} \xi_j \cdot y_{ij} \leq q \cdot x_i && \text{for } i \in I, \xi \in \mathcal{U} && (5.16c)\\
& w_i - \sum_{k=1}^{\sigma_j^{-1}(i)-1} w_{\sigma_j(k)} \leq y_{ij} && \text{for } j \in J, i \in N(j) && (5.16d)\\
& y_{ij} \geq 0 && \text{for } i \in I, j \in J && (5.16e)\\
& w_i, x_i \in \mathbb{N} && \text{for } i \in I. && (5.16f)
\end{aligned}$$

Analogously to Lemma 5.36 and Corollary 5.37 it follows:

Lemma 5.43. *Let (w, x, y) be feasible for $P_{OC}(\mathcal{U})$. Then, there is a solution (w', x', y') with $w' \in \mathbb{B}^{|I|}$, $x'(I) \leq x(I)$, and the property that, for $I' := \{i \in I \colon w'_i = 1\}$, x' is the corresponding induced solution and $y' = y(I')$.* □

Corollary 5.44. *Let an instance $\mathcal{I}$ of Robust Min q-OC be given. A vector $x \in \mathbb{N}^{|I|}$ is optimal for $\mathcal{I}$ if and only if there are $w \in \mathbb{N}^{|I|}$ and $y \in \mathbb{R}_{\geq 0}^{|I| \times |J|}$ such that (w, x, y) is optimal for $P_{OC}(\mathcal{U})$.* □

Observe that, in contrast to $\mathrm{P_{FC}}(\mathcal{U})$ and $\mathrm{P_{AC}}(\mathcal{U})$, we cannot aggregate the constraints (5.16c) directly here. To solve $\mathrm{P_{OC}}(\mathcal{U})$ applying constraint generation for the constraints (5.16c) is therefore a meaningful method, cf. Section 2.5. Here we iteratively solve $\mathrm{P_{OC}}(\mathcal{U}')$ for a smaller uncertainty set $\mathcal{U}' \subseteq \mathcal{U}$, which might be empty in the first iteration. Let (w', x', y') be feasible for $\mathrm{P_{OC}}(\mathcal{U}')$ with induced feasible set $I' \subseteq I$. In the separation step, we seek a scenario $\xi \in \mathcal{U}$ and a location $i \in I$ with $\xi(N'(i)) > q \cdot x'_i$. Thus, again we need to solve $|I|$ instances of Max Robust Sum and the complexity of the separation step is particularly determined by the structure of $\mathcal{U}$.

Let us briefly consider a case for which we can reobtain a compact formulation. To that end, let $(\bar{w}, \bar{x}, \bar{y})$ be feasible for $\mathrm{P_{OC}}(\mathcal{U})$. By (5.16c) we get that, for $i \in I$,

$$\max_{\xi \in \mathcal{U}} \left\{ \sum_{j \in N(i)} \xi_j \cdot \bar{y}_{ij} \right\} \leq q \cdot \bar{x}_i. \tag{5.17}$$

Thus, if the left-hand side of (5.17) can be reformulated, we might get an easier formulation for Robust Min q-OC that can be solved directly using some IP-solver. For instance, for interval uncertainty, where $\mathcal{U}_\mathrm{I} = \left\{\xi \in \mathbb{N}^{|J|} \colon a \leq \xi \leq b\right\}$, it can readily be seen that (5.16c) is equivalent to $\sum_{j \in N(i)} b_j \cdot y_{ij} \leq q \cdot x_i$ for $i \in I$ and the total number of constraints of $\mathrm{P_{OC}}(\mathcal{U}_\mathrm{I})$ is polynomially bounded.

Now, more generally, suppose the uncertainty set $\mathcal{U}$ under consideration is given by $\mathcal{U} = \left\{\xi \in \mathbb{N}^{|J|} \colon A\xi \leq b\right\}$ for a matrix $A \in \mathbb{Q}^{l \times |J|}$ and a vector $b \in \mathbb{Q}^l$ with $l \in \mathbb{N}_{>0}$ and an integral underlying polytope $\{\xi \colon A\xi \leq b, \xi \geq 0\}$. That means, the uncertainty set $\mathcal{U}$ is integral polyhedral. Then, by LP-duality we get

$$\begin{aligned}\max\left\{c^T\xi\colon A\xi \le b, \xi \in \mathbb{N}^{|J|}\right\} &= \max\left\{c^T\xi\colon A\xi \le b, \xi \ge 0\right\} \\ &= \min\left\{b^T\pi\colon A^T\pi \ge c, \pi \ge 0\right\}\end{aligned} \tag{5.18}$$

for a vector $c \in \mathbb{Q}^{|J|}$. Note that all problems have a finite optimal solution as $|\mathcal{U}| < \infty$ and by duality, cf. [Mey74]. Applying (5.18) to (5.17) gives, for $i \in I$,

$$\min\left\{b^T\pi\colon A^T\pi \ge \bar{y}_i, \pi \ge 0\right\} \le q \cdot \bar{x}_i \tag{5.19}$$

where $\bar{y}_i \in \mathbb{R}^{|J|}$ with $(\bar{y}_i)_j := \bar{y}_{ij}$ for $j \in J$. Note that we can assume $\bar{y}_{ij} = 0$ for $j \notin N(i)$. If a vector $\bar{\pi} \in \mathbb{R}^l$ is feasible for the left-hand side of (5.19) with $b^T\bar{\pi} \le q \cdot \bar{x}_i$, we directly get that (5.19) is fulfilled as for the optimal solution $\pi^\star \in \mathbb{R}^l$ we have $b^T\pi^\star \le b^T\bar{\pi}$, cf. [BS04]. Therefore, (5.19) is satisfied if and only if there is a solution $\pi \in \mathbb{R}^l$ to

$$\begin{aligned} b^T\pi &\le q \cdot \bar{x}_i \\ A^T\pi &\ge \bar{y}_i \\ \pi &\ge 0. \end{aligned}$$

Observe that our initial solution $(\bar{w}, \bar{x}, \bar{y})$ only appears on the right-hand side here and thus, for every $i \in I$, we can replace (5.16c) with these constraints. We get a variable vector $\pi^i \in \mathbb{R}^l$ for every location i:

$$\min_{w,x,y,\pi} \quad \sum_{i\in I} x_i \tag{5.20a}$$

$$\text{s.t.} \quad \sum_{i\in N(j)} w_i \ge 1 \qquad \text{for } j \in J \tag{5.20b}$$

$$b^T\pi^i \le q \cdot x_i \qquad \text{for } i \in I \tag{5.20c}$$

$$A^T\pi^i \ge y_i \qquad \text{for } i \in I \tag{5.20d}$$

$$w_i - \sum_{k=1}^{\sigma_j^{-1}(i)-1} w_{\sigma_j(k)} \le y_{ij} \qquad \text{for } j \in J, i \in N(j) \tag{5.20e}$$

$$y_{ij} \ge 0 \qquad \text{for } i \in I, j \in J \tag{5.20f}$$

$$w_i, x_i \in \mathbb{N} \qquad \text{for } i \in I \tag{5.20g}$$

$$\pi^i \ge 0 \qquad \text{for } i \in I. \tag{5.20h}$$

Again, we denote by y_i the vector with $(y_i)_j := y_{ij}$ for $j \in J$ and $\pi := (\pi^1, \ldots, \pi^r)$ with $r := |I|$.

Theorem 5.45. *Let an instance $\mathcal{I}$ of Robust Min q-OC be given with $\mathcal{U} = P(A, b) \cap \mathbb{N}^{|J|}$ for an integral polytope $P(A, b) \subseteq \mathbb{R}^{|J|}_{\geq 0}$ with $A \in \mathbb{Q}^{l \times |J|}$, $b \in \mathbb{Q}^l$ for $l \in \mathbb{N}_{>0}$. A vector $x \in \mathbb{N}^{|I|}$ is optimal for $\mathcal{I}$ if and only if there are $w \in \mathbb{N}^{|I|}$, $y \in \mathbb{R}^{|I| \times |J|}_{\geq 0}$, and $\pi^i \in \mathbb{R}^l_{\geq 0}$ for $i \in I$ such that (w, x, y, π) is optimal for* (5.20)*.*

Proof. By Corollary 5.44 and the preceding argumentation it suffices to show that (w, x, y) is feasible for $\mathrm{P_{OC}}(\mathcal{U})$ if and only if, for every $i \in I$, there is $\pi^i \in \mathbb{R}^l$ such that (w, x, y, π) is feasible for (5.20). Let (w, x, y) be feasible for $\mathrm{P_{OC}}(\mathcal{U})$. Without loss of generality, we can assume that, for every $i \in I$ and $j \in J$, $y_{ij} = 0$ if $j \notin N(i)$. Denote by π^i an optimal solution to $\min \left\{ b^T\pi \colon A^T\pi \geq y_i, \pi \geq 0 \right\}$. Then, for every $i \in I$,

$$b^T\pi^i = \max_{\xi \in \mathcal{U}} \sum_{j \in N(i)} \xi_j \cdot y_{ij} \leq q \cdot x_i.$$

Thus, (w, x, y, π) is feasible for (5.20). On the other hand, let (w, x, y, π) be feasible for (5.20). Then, for every $i \in I$,

$$\max_{\xi \in \mathcal{U}} \sum_{j \in N(i)} \xi_j \cdot y_{ij} \leq b^T\pi^i \leq q \cdot x_i$$

so that (w, x, y) is feasible for $\mathrm{P_{OC}}(\mathcal{U})$. □

Analogously, the above consideration is also valid for general integral polyhedral uncertainty sets, cf. Page 30, but omitted for convenience.

Corollary 5.46. *For any fixed $q \in \mathbb{N}_{>0}$, Robust q-OC with integral polyhedral uncertainty is strongly* NP*-complete.*

Proof. NP-hardness follows from the proof of Theorem 5.41 as the constructed uncertainty set is integral polyhedral. Furthermore, given a solution to (5.20) as a certificate, we can verify a yes-instance in polynomial time. Observe that we can assume the encoding length of the solution to be polynomially bounded in the encoding length of the given instance [Sch98]. Hence, the problem is contained in NP. □

As in the previous sections, we now aim to increase the level of detail by considering specific classes of uncertainty sets.

	$\lvert\mathcal{U}\rvert = 1$	$\lvert\mathcal{U}\rvert = k \geq 2$
$q = 1$	linear time	APX-complete
$q \geq 2$	APX-complete	APX-complete

Table 5.1.: Complexity analysis of Robust Min q-OC for a fixed number of scenarios $k \in \mathbb{N}_{>0}$.

5.3.2. Specific Classes of Uncertainty Sets

In this section, we analyze Robust Min q-OC restricted to instances for which the uncertainty sets belong to a specific class. Hence, we fix an instance of Robust Min q-OC and we successively assume that its uncertainty set $\mathcal{U}$ belongs to some particular class of uncertainty sets. An overview and general assumptions on these classes can be found in Section 2.6.

Discrete Uncertainty

We start by considering discrete uncertainty, i.e., $\mathcal{U} = \left\{\xi^1, \ldots, \xi^k\right\}$ for some fixed $k \in \mathbb{N}_{>0}$. It is easy to see that we can separate in polynomial time here and $\mathrm{P}_{\mathrm{OC}}(\mathcal{U})$ is a compact formulation. Furthermore, given a subset $I' \subseteq I$, the corresponding induced solution x' can be computed in time $\mathcal{O}(|I| + |E|)$. The complexity results for $k = 1$ are subject of Section 5.3. Therefore, here we concentrate on $k \geq 2$. Table 5.1 summarizes our results. First of all, we show that Robust Min q-OC with discrete uncertainty is contained in APX.

Lemma 5.47. *For any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-OC with discrete uncertainty is contained in* APX.

Proof. Similar to the proof of Lemma 5.29 we show that every induced solution to an instance $\mathcal{I}$ of Robust Min q-OC with $|\mathcal{U}| = k$ for some fixed $k \in \mathbb{N}_{>0}$ leads to a constant factor approximation. Let $I' \subseteq I$ be feasible with corresponding induced solution x' and let

$$\xi^i \in \arg\max_{\xi \in \mathcal{U}} \xi(N'(i))$$

for $i \in I'$. Further, denote by $i_j \in I'$ the location responsible for $j \in J$. Note that $q \cdot \mathrm{OPT}(\mathcal{I}) \geq \xi^i(J)$ for every $i \in I'$ and $\dot{\bigcup}_{i \in I} N'(i) = J$. With (5.13) we have

$$
\begin{aligned}
x'(I) &= \sum_{i \in I'} \left\lceil \frac{\xi^i(N'(i))}{q} \right\rceil \leq \sum_{j \in J} \left\lceil \frac{\xi_j^{i_j}}{q} \right\rceil \\
&\leq \sum_{j \in J} \left(\frac{\xi_j^{i_j}}{q} + \frac{q-1}{q} \right) \leq k \cdot \mathrm{OPT}(\mathcal{I}) + \frac{q-1}{q} \cdot |J| \\
&\overset{(5.13)}{\leq} (k + k \cdot (q-1)) \cdot \mathrm{OPT}(\mathcal{I}) = qk \cdot \mathrm{OPT}(\mathcal{I}).
\end{aligned}
$$

Hence, the feasible set $\{\sigma_j(1) \colon j \in J\} \subseteq I$ leads to a qk-approximation. As the corresponding induced solution can be computed in polynomial time and k is fixed, we obtain the desired result. $\square$

For $k = 1$, the approximation algorithm of Lemma 5.47 reproduces the q-approximation algorithm of Lemma 5.29. When additionally using the construction in the proof of Theorem 5.30 we are now able to complete the right column of Table 5.1.

Theorem 5.48. *For any fixed $k \geq 2$, Robust Min 1-OC with discrete uncertainty and $|\mathcal{U}| = k$ is* APX*-complete.*

Proof. The membership in APX is given by Lemma 5.47. To show APX-hardness, we present an L-reduction from Max 3-SAT(3), where each variable appears at most three times and each clause has at most three literals, cf. Problem 3.53 and Assumption 3.54. We start with the case $k = 2$. Thus, given an instance $\mathcal{I}$ of Max 3-SAT(3) with $m \in \mathbb{N}_{>0}$ clauses $C_1, \ldots, C_m$ and $n \in \mathbb{N}_{>0}$ variables $X_1, \ldots, X_n$, we apply Procedure 5.31 to obtain the bipartite graph $G = (I \cup J, E)$ with locations $I = \{x_1, \bar{x}_1, \ldots, x_n, \bar{x}_n\}$, regions $J = J_\theta \cup J_{\bar{\theta}} \cup J_\gamma$, and the preference orders of the regions. The uncertainty set $\mathcal{U}$ consists of two scenarios ξ^1 and ξ^2 with

$$
\xi_j^1 := \begin{cases} 1, & \text{if } j \in J_\theta \cup J_\gamma, \\ 0, & \text{else,} \end{cases} \qquad \text{and} \qquad \xi_j^2 := \begin{cases} 1, & \text{if } j \in J_{\bar{\theta}} \cup J_\gamma, \\ 0, & \text{else.} \end{cases}
$$

Note that $\xi^1(J) = \xi^2(J) = n+m$. This polynomial time construction results in our instance $\mathcal{I}'$ of Robust Min q-OC with $\mathcal{U} = \{\xi^1, \xi^2\}$. An example of the constructed instance is illustrated in Figure 5.4.

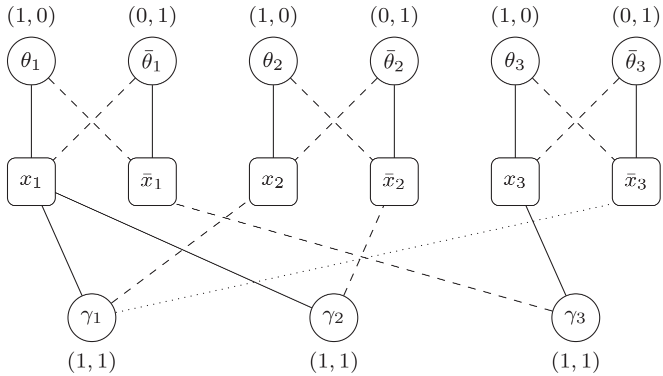


Figure 5.4.: Construction of instance $\mathcal{I}'$ in the proof of Theorem 5.48 for $C_1 = X_1 \vee X_2 \vee \overline{X}_3$, $C_2 = X_1 \vee \overline{X}_2$, and $C_3 = X_3 \vee \overline{X}_1$. A solid edge is preferred to a dashed edge which itself is preferred to a dotted edge. The first and second entry of the vertex label of region j represent ξ_j^1 and ξ_j^2, respectively.

Now, given a truth assignment f for $\mathcal{I}$ fulfilling exactly $r \in \mathbb{N}$ clauses, we obtain a solution $I' \subseteq I$ to $\mathcal{I}'$ by applying Procedure 5.32. Then, by the choice of the scenarios and the preference orders, in every open location $i \in I'$, there is one supplier who covers the demand of the regions in J_θ and $J_{\bar{\theta}}$, i.e., these sum up to at most $n+m-r$ many. Furthermore, for every region $\gamma_j \in J_\gamma$, there is one supplier in some adjacent location covering its demand in both scenarios. Thus, the induced solution value is bounded by $n + 2m - r$ and as in the proof of Theorem 5.30 we obtain

$$\mathrm{OPT}(\mathcal{I}') \leq 6 \cdot \mathrm{OPT}(\mathcal{I}).$$

On the other hand, given a solution $I' \subseteq I$ to $\mathcal{I}'$, we obtain a truth assignment f for $\mathcal{I}$ by applying Procedure 5.33. Again, we obtain $\mathrm{SOL}(\mathcal{I}', I') \geq n+m+r$ for $r \in \mathbb{N}$ being the number of variables such that both locations x_i and $\bar{x}_i$ are open. Analogously to Theorem 5.30 we get $\mathrm{SOL}(\mathcal{I}, f) \geq m - r$ and this gives

$$\mathrm{OPT}(\mathcal{I}) - \mathrm{SOL}(\mathcal{I}, f) \leq \mathrm{SOL}(\mathcal{I}', I') - \mathrm{OPT}(\mathcal{I}').$$

For $k \geq 3$, we include dummy locations, dummy regions, and dummy scenarios as in the proof of Theorem 3.57. Observe that for a dummy region there is only one feasible preference order. Now, as $\mathrm{OPT}(\mathcal{I}') \leq 6 \cdot \mathrm{OPT}(\mathcal{I}) + (k-2) \leq (k+4) \cdot \mathrm{OPT}(\mathcal{I})$, the proof goes along the same lines as for $k = 2$. □

Corollary 5.49. *For any fixed $k \geq 2$, Robust 1-OC with discrete uncertainty and $|\mathcal{U}| = k$ is strongly NP-complete.*

Proof. Robust 1-OC with discrete uncertainty is contained in NP. By the proof of Theorem 5.48 we get that a solution to $\mathcal{I}$ with at least $r \in \mathbb{N}$ satisfied clauses leads to a solution to $\mathcal{I}'$ with at most $n + 2m - r + \max\{0, k-2\}$ suppliers in polynomial time and vice versa. □

Corollary 5.50. *For any fixed $q, k \in \mathbb{N}_{>0}$ with $qk \geq 2$, Robust Min q-OC with discrete uncertainty and $|\mathcal{U}| = k$ is APX-complete and its decision version is strongly NP-complete.*

Proof. For $q = 1$, the result follows from Theorem 5.48 and Corollary 5.49. For $q = 2$, the result follows from Theorem 5.30 and Corollary 5.34 with the help of dummies as in Theorem 5.48 and Corollary 5.49. For $q \geq 3$, the result follows from Theorem 5.35 also with the help of dummies as in Theorem 3.57 and Corollary 3.58. □

All in all, we have seen that the value of the product $q \cdot |\mathcal{U}|$ is crucial for the polynomial time solvability of the problem. This is also the case for Robust Min q-MSMC with discrete uncertainty, cf. Section 3.4.1. Yet, for Robust Min q-OC with discrete uncertainty, we can only guarantee a polynomial time algorithm for $q = 1$ and $|\mathcal{U}| = 1$. For $q \cdot |\mathcal{U}| \geq 2$, the problem is already APX-complete. For Robust Min q-MSMC this border is crossed as recently as $q \cdot |\mathcal{U}| \geq 3$.

Interval Uncertainty

If we consider interval uncertainty sets, we have $\mathcal{U} = \left\{\xi \in \mathbb{N}^{|J|} \colon a \leq \xi \leq b\right\}$ for vectors $a, b \in \mathbb{N}^{|J|}$ with $a \leq b$. It can readily be seen that it suffices to consider the worst-case scenario b so that $\mathrm{P_{OC}}(\mathcal{U}) = \mathrm{P_{OC}}(b)$ and the complexity results of Min q-OC carry over. Hence, Robust Min q-OC with interval uncertainty is solvable in time $\mathcal{O}(|I| + |J|)$ for $q = 1$ and is APX-complete otherwise. As for the non-robust case, given a feasible set of locations, the corresponding induced solution can be computed in time $\mathcal{O}(|I| + |E|)$.

Budgeted and Γ-Uncertainty

Let us first analyze budgeted uncertainty where, in addition to interval uncertainty, we aim to prevent the global worst-case scenario by imposing an upper bound on the total sum of clients. Thus, we look at the uncertainty set $\mathcal{U} = \left\{\xi \in \mathbb{N}^{|J|} \colon a \le \xi \le b,\ \xi(J) \le \Gamma\right\}$ for some $\Gamma \in \mathbb{N}$, cf. Assumption 3.60. Consider an instance of Robust Min q-OC with budgeted uncertainty and a feasible set $I' \subseteq I$. For the corresponding induced solution x', we have

$$x'_i = \left\lceil \frac{\max_{\xi \in \mathcal{U}} \xi(N'(i))}{q} \right\rceil = \left\lceil \frac{\min\left\{b(N'(i)), \Gamma - a(J \setminus N'(i))\right\}}{q} \right\rceil$$

for $i \in I$ by Lemma 3.62. Hence, x' can be computed in time $\mathcal{O}(|I| + |E|)$.

Similarly, for Γ-uncertainty, we allow some demands to shift to their upper bound. Recall that $\mathcal{U} = \left\{\xi \in \mathbb{N}^{|J|} \colon a \le \xi \le a + \hat{a},\ |\left\{k \colon \xi_k \ne a_k\right\}| \le \Gamma\right\}$ with $a, \hat{a} \in \mathbb{N}^{|J|}$ and $\Gamma \in \{0, 1, \ldots, |J|\}$. Now, the corresponding induced solution x' is given by

$$x'_i = \left\lceil \frac{a(N'(i)) + \max\left\{\hat{a}(T) \colon T \subseteq N'(i), |T| \le \Gamma\right\}}{q} \right\rceil$$

for $i \in I$ by Lemma 3.76. Therefore, computing x' needs time $\mathcal{O}(|I| + |E|)$, cf. Theorem 5.9.

In the proof of Theorem 5.41, we construct the uncertainty set

$$\{0, e_1, \ldots, e_n\} \subseteq \mathbb{N}^n,$$

which is clearly a budgeted as well as a Γ-uncertainty set. Furthermore, budgeted and Γ-uncertainty are both integral polyhedral, cf. Page 31. Hence, as in Corollary 5.46 we get that Robust q-OC with budgeted or Γ-uncertainty is contained in NP.

Theorem 5.51. *For any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-OC with budgeted uncertainty is* APX*-hard and its decision version is strongly* NP*-complete.* □

Theorem 5.52. *For any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-OC with Γ-uncertainty is* APX*-hard and its decision version is strongly* NP*-complete.* □

Let $\mathcal{U}_\mathrm{B}$ and $\mathcal{U}_\Gamma$ be a budgeted and a Γ-uncertainty set, respectively. We aim to regain a compact formulation for $\mathrm{P_{OC}}(\mathcal{U}_\mathrm{B})$ and $\mathrm{P_{OC}}(\mathcal{U}_\Gamma)$. Thus, with slight abuse of notation, applying LP-duality for $c \in \mathbb{Q}^{|J|}$ gives

$$\begin{array}{llcllr} \max\limits_{\xi} & c^T\xi & = & \min\limits_{\alpha,\beta,\gamma} & b^T\beta - a^T\alpha + \Gamma\cdot\gamma & \\ \text{s.t.} & \xi \in \mathcal{U}_\mathrm{B} & & \text{s.t.} & \beta_j - \alpha_j + \gamma \geq c_j & \text{for } j \in J \\ & & & & \alpha,\beta,\gamma \geq 0, & \end{array}$$

and

$$\begin{array}{llcllr} \max\limits_{\xi} & c^T\xi & = & \min\limits_{\alpha,\beta,\gamma,\delta} & a^T\beta - a^T\alpha + \Gamma\cdot\gamma + 1^T\delta & \\ \text{s.t.} & \xi \in \mathcal{U}_\Gamma & & \text{s.t.} & \beta_j - \alpha_j \geq c_j & \text{for } j \in J \\ & & & & \delta_j + \gamma - \hat{a}_j\cdot\beta_j \geq 0 & \text{for } j \in J \\ & & & & \alpha,\beta,\gamma,\delta \geq 0, & \end{array}$$

cf. Page 31. For example, when applying Theorem 5.45 to an instance of Robust Min q-OC with budgeted uncertainty, we obtain the following mixed integer formulation with additional variable vectors $\alpha^i, \beta^i \in \mathbb{R}^{|J|}$ and $\gamma^i \in \mathbb{R}$ for $i \in I$:

$$\min_{w,x,y,\alpha,\beta,\gamma} \quad \sum_{i\in I} x_i \tag{5.21a}$$

$$\text{s.t.} \quad \sum_{i\in N(j)} w_i \geq 1 \qquad \text{for } j \in J \tag{5.21b}$$

$$b^T\beta^i - a^T\alpha^i + \Gamma\cdot\gamma^i \leq q\cdot x_i \qquad \text{for } i \in I \tag{5.21c}$$

$$\beta^i - \alpha^i + \gamma^i \geq y_i \qquad \text{for } i \in I \tag{5.21d}$$

$$w_i - \sum_{k=1}^{\sigma_j^{-1}(i)-1} w_{\sigma_j(k)} \leq y_{ij} \qquad \text{for } j \in J, i \in N(j) \tag{5.21e}$$

$$y_{ij} \geq 0 \qquad \text{for } i \in I, j \in J \tag{5.21f}$$

$$\alpha^i, \beta^i, \gamma^i \geq 0 \qquad \text{for } i \in I \tag{5.21g}$$

$$w_i, x_i \in \mathbb{N} \qquad \text{for } i \in I, \tag{5.21h}$$

where $\alpha = (\alpha^1, \dots, \alpha^r)$, $\beta = (\beta^1, \dots, \beta^r)$, and $\gamma = (\gamma^1, \dots, \gamma^r)$ for $I \coloneqq \{1, \dots, r\}$ with $r \in \mathbb{N}_{>0}$. Furthermore, we have $(y_i)_j \coloneqq y_{ij}$ for $i \in I$ and $j \in J$.

Multi-budgeted Uncertainty

With multi-budgeted uncertainty we consider uncertainty sets of the form $\mathcal{U} = \{\xi \in \mathbb{N}^{|J|} : a_S \leq \xi(S) \leq b_S \text{ for } S \in \mathcal{S}\}$ where $\mathcal{S} \subseteq 2^J$ and $a_S, b_S \in \mathbb{N}$ with $a_S \leq b_S$ for every $S \in \mathcal{S}$. As this class constitutes a generalization of budgeted uncertainty, we directly obtain:

Theorem 5.53. *For any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-OC with multi-budgeted uncertainty is* APX*-hard and its decision version is strongly* NP*-hard.* □

In contrast to the previous classes of uncertainty sets, we know that Max Robust Sum is NP-complete for multi-budgeted uncertainty, cf. Lemma 3.67. Thus, analogously to the proof of Theorem 5.5, we have:

Lemma 5.54. *Given an instance of Robust Min q-OC with multi-budgeted uncertainty and a feasible set $I' \subseteq I$, computing the corresponding induced solution x' is* NP*-hard.* □

Ellipsoidal Uncertainty

Last but not least, we focus on ellipsoidal uncertainty sets, i.e., a scenario $\xi \in \mathcal{U}$ fulfills $(\xi - a)^T A^{-1}(\xi - a) \leq 1$ for a vector $a \in \mathbb{N}^{|J|}$ and a positive definite matrix $A \in \mathbb{Q}^{|J| \times |J|}$. Recall, that we assume that the underlying ellipsoid $\mathrm{E}(A, a)$ is part of the non-negative orthant. By Theorem 3.72 we know that Robust Sum with ellipsoidal uncertainty is NP-complete. Hence, with Remark 3.74 we get:

Lemma 5.55. *Given an instance of Robust Min q-OC with ellipsoidal uncertainty and a feasible set $I' \subseteq I$, computing the corresponding induced solution x' is* NP*-hard.* □

Furthermore, for $q \geq 2$, we can apply Theorem 5.30, Corollary 5.34, and Theorem 5.35 and analogously to the proof of Theorem 3.68 we get:

Theorem 5.56. *For any fixed $q \geq 2$, Robust Min q-OC with ellipsoidal uncertainty is* APX*-hard and its decision version is strongly* NP*-hard.* □

Thus, we stay with the remaining case $q = 1$. Observe that the proof of Theorem 3.70 is build on the fact that the assignment of suppliers to regions varies over the scenarios. Thus, when including any preference order in this construction, the utilized arguments no longer apply. Nevertheless, there is also the possibility to use the construction of Theorem 5.48. Choosing an appropriate uncertainty set leads to the following result.

Theorem 5.57. *For any fixed $q \in \mathbb{N}_{>0}$, Robust Min q-OC with ellipsoidal uncertainty is* APX*-hard and its decision version is strongly* NP*-hard.*

Proof. It remains to prove the case $q = 1$. To show APX-hardness we provide an L-reduction from Max 3-SAT(3), where each variable appears at most three times and each clause has at most three literals, cf. Problem 3.53 and Assumption 3.54. Thus, given an instance $\mathcal{I}$ of Max 3-SAT(3) with $m \in \mathbb{N}_{>0}$ clauses $C_1, \dots, C_m$ and $n \in \mathbb{N}_{>0}$ variables $X_1, \dots, X_n$, we construct the bipartite graph $G = (I \cup J, E)$ of Procedure 5.31 with locations $I = \{x_1, \bar{x}_1, \dots, x_n, \bar{x}_n\}$ and regions $J = J_\theta \cup J_{\bar{\theta}} \cup J_\gamma$. We also copy the preference orders of the regions from the proof of Theorem 5.30. The ellipsoidal uncertainty set $\mathcal{U}$ is defined as follows: For $p := 2n + m$ we set

$$\mathcal{U} := \left\{ (\varphi, \bar{\varphi}, \tau) \in \mathbb{N}^p \colon \sum_{i=1}^{n} \left((\varphi_i - 1)^2 + (\bar{\varphi}_i - 1)^2 \right) + \sum_{j=1}^{m} 2\,(\tau_j - 1)^2 \leq 1 \right\}.$$

For a scenario $(\varphi, \bar{\varphi}, \tau) \in \mathcal{U}$ and $i \in \{1, \dots, n\}$, φ_i represents the demand of region θ_i and $\bar{\varphi}_i$ represents the demand of region $\bar{\theta}_i$ and, for $j \in \{1, \dots, m\}$, τ_j represents the demand of region γ_j. We have $\mathcal{U} = \mathrm{E}(A^{-1}, 1) \cap \mathbb{N}^p$ with A being a diagonal matrix with p columns and rows having value 1 on the diagonal for the first $2n$ columns and value 2 otherwise. Thus, the matrix A is positive definite and $\mathrm{E}(A^{-1}, 1) \subseteq \mathbb{R}^p_{\geq 0}$. Taking a closer look at the definition of $\mathcal{U}$, we see that it suffices to focus on the scenarios $(1, 1, 1) + (e_k, 0)$ with $k \in \{1, \dots, 2n\}$. All in all, this polynomial time construction results in our instance $\mathcal{I}'$ of Robust Min 1-OC with ellipsoidal uncertainty.

Now, given a truth assignment f for $\mathcal{I}$ fulfilling exactly $r \in \mathbb{N}$ clauses, we obtain a solution $I' \subseteq I$ to $\mathcal{I}'$ by applying Procedure 5.32. Consider a location x_i. If $\bar{x}_i$ is closed, we need three suppliers in location x_i to cover the demands of the regions θ_i and $\bar{\theta}_i$ by the choice of the scenarios. If $\bar{x}_i$ is open, we need two suppliers in x_i as well as two suppliers in $\bar{x}_i$ to cover these demands by the construction of the preference orders. In total, these suppliers sum up to at most $3n + m - r$ many. Furthermore, for every region $\gamma_j \in J_\gamma$, there is one additional supplier in some adjacent location covering its demand in all scenarios. Thus, the induced solution value is bounded by

$3n+2m-r$. This gives

$$\begin{aligned}\mathrm{OPT}(\mathcal{I}') &\leq 3n+2m-\mathrm{OPT}(\mathcal{I}) \qquad (5.22)\\ &\overset{(3.17)}{\leq} \frac{9}{2}m+2m-\mathrm{OPT}(\mathcal{I}) = \frac{13}{2}m-\mathrm{OPT}(\mathcal{I}) \overset{(3.16)}{\leq} 12\cdot\mathrm{OPT}(\mathcal{I}).\end{aligned}$$

On the other hand, given a solution $I' \subseteq I$ to $\mathcal{I}'$, we obtain a truth assignment f for $\mathcal{I}$ by applying Procedure 5.33. Then, we obtain $\mathrm{SOL}(\mathcal{I}', I') \geq 3n+m+r$ for $r \in \mathbb{N}$ being the number of variables such that both locations x_i and $\bar{x}_i$ are open. Furthermore, we get $\mathrm{SOL}(\mathcal{I}, f) \geq m-r$ as in the proof of Theorem 5.30 and with (5.22) we infer

$$\begin{aligned}\mathrm{OPT}(\mathcal{I})-\mathrm{SOL}(\mathcal{I}, f) &\leq \mathrm{OPT}(\mathcal{I})-m+r\\ &= 3n+m+r-(3n+2m-\mathrm{OPT}(\mathcal{I}))\\ &\leq \mathrm{SOL}(\mathcal{I}', I')-\mathrm{OPT}(\mathcal{I}').\end{aligned}$$

NP-hardness of Robust 1-OC follows as the proof reveals that a solution to $\mathcal{I}$ with at least $r \in \mathbb{N}$ satisfied clauses leads to a solution to $\mathcal{I}'$ with at most $3n+2m-r$ many suppliers in polynomial time and vice versa. □

This concludes our study of specific classes of uncertainty sets. For most of the results, we were able to adapt and extend previous ideas and findings on the problems Robust Min q-MSMC, Robust Min q-FC, and Max Robust Sum. Yet, each of these adaptions needed particular care.

5.3.3. Approximating Robust Min q-Ordered Clients

We have seen that, unless $q=1$ and $|\mathcal{U}|=1$, Robust Min q-Ordered Clients is a hard problem even when restricted to specific classes of uncertainty sets. For this reason, we briefly analyze ideas to approximate this problem in this section.

For Min q-OC we already have a q-approximation algorithm due to Lemma 5.29 that simply opens, for every region, its most preferred location. Using the same procedure, we get a qk-approximation algorithm for Robust Min q-OC where $|\mathcal{U}|=k$ for $k \in \mathbb{N}_{>0}$, cf. Lemma 5.47. Thus, we get a constant factor approximation algorithm in case of discrete uncertainty.

As mentioned on Page 167, if for an instance of Min q-OC it holds true that $q \geq d(N(i))$ for every $i \in I$, the problem reduces to a Min Set Cover(q)

problem. Hence, present approximation algorithms for this problem might be of use here, cf. Section 4.2.1.

Definition 5.58. Let an instance $\mathcal{I}$ of Robust Min q-OC be given. We say that the Min Set Cover instance with $S := J$ and $\mathcal{C} := \{N(i) \colon i \in I\}$ *corresponds* to $\mathcal{I}$.

For an instance $\mathcal{I}$ of Robust Min q-OC, every solution to the corresponding Min Set Cover instance $\mathcal{I}_{\mathrm{SC}}$ is given by a subset $I' \subseteq I$ such that $\{N(i) \colon i \in I'\}$ is a cover for J, i.e. I' is feasible. Conversely, let $x \in \mathbb{N}^{|I|}$ be feasible for $\mathcal{I}$. Any region $j \in J$ is adjacent to some $i \in N(j)$ with $x_i \geq 1$. Thus, the set I_x is a cover for $\mathcal{I}_{\mathrm{SC}}$ and we have $\mathrm{OPT}(\mathcal{I}_{\mathrm{SC}}) \leq \mathrm{OPT}(\mathcal{I})$. Further, to utilize $\mathcal{I}_{\mathrm{SC}}$ we derive a new upper bound on the value of an induced solution similar to the proof of Lemma 5.29.

Lemma 5.59. *Let $I' \subseteq I$ be feasible for an instance $\mathcal{I}$ of Robust Min q-OC with induced solution x'. Then, it holds true that*

$$x'(I) \leq |\mathcal{U}| \cdot \mathrm{OPT}(\mathcal{I}) + \frac{q-1}{q} \cdot |I'|.$$

Proof. We have

$$x'(I) = \sum_{i \in I'} \left\lceil \frac{\max_{\xi \in \mathcal{U}} \xi(N'(i))}{q} \right\rceil \leq \sum_{i \in I'} \left(\frac{\max_{\xi \in \mathcal{U}} \xi(N'(i))}{q} + \frac{q-1}{q} \right).$$

For each $i \in I'$, let $\xi^i \in \arg\max_{\xi \in \mathcal{U}} \xi(N'(i))$ and set $\mathcal{U}' := \left\{\xi^i \colon i \in I'\right\}$. As $\{N'(i) \colon i \in I'\}$ is a partition of J, we have

$$\sum_{i \in I'} \frac{\max_{\xi \in \mathcal{U}} \xi(N'(i))}{q} = \sum_{i \in I'} \frac{\xi^i(N'(i))}{q} \leq \sum_{\xi \in \mathcal{U}'} \frac{\xi(J)}{q} \leq |\mathcal{U}| \cdot \mathrm{OPT}(\mathcal{I}).$$

Combining these observations yields the desired result. □

Corollary 5.60. *Let an instance $\mathcal{I}$ of Robust Min q-OC be given. If, for $c \geq 1$, we have a c-approximation algorithm for Min Set Cover, there is a solution x for $\mathcal{I}$ with*

$$x(I) \leq \left(|\mathcal{U}| + c \cdot \frac{q-1}{q} \right) \cdot \mathrm{OPT}(\mathcal{I}).$$

If induced solutions for $\mathcal{I}$ can be computed in polynomial time, this leads to an approximation algorithm.

Proof. Let $I' \subseteq I$ be a c-approximation for the Min Set Cover instance $\mathcal{I}_{\mathrm{SC}}$ corresponding to $\mathcal{I}$. Let x' be the solution induced by I'. With Lemma 5.59 we get

$$\begin{aligned} x'(I) &\leq |\mathcal{U}| \cdot \mathrm{OPT}(\mathcal{I}) + \frac{q-1}{q} \cdot |I'| \\ &\leq |\mathcal{U}| \cdot \mathrm{OPT}(\mathcal{I}) + \frac{q-1}{q} \cdot c \cdot \mathrm{OPT}(\mathcal{I}_{\mathrm{SC}}) \\ &\leq |\mathcal{U}| \cdot \mathrm{OPT}(\mathcal{I}) + \frac{q-1}{q} \cdot c \cdot \mathrm{OPT}(\mathcal{I}) \\ &= \left(|\mathcal{U}| + c \cdot \frac{q-1}{q}\right) \cdot \mathrm{OPT}(\mathcal{I}) \end{aligned}$$

as $\mathrm{OPT}(\mathcal{I}_{\mathrm{SC}}) \leq \mathrm{OPT}(\mathcal{I})$. □

Observe that, for an instance $\mathcal{I}$ of Robust Min q-OC and a feasible set of locations $I' \subseteq I$, we can assume that $|I'| \leq |J|$. Otherwise, we can remove locations from I' without changing the induced solution value. Hence, for the corresponding induced solution x', we have $x'(I) \leq \min\{|I|, |J|\} \cdot \mathrm{OPT}(\mathcal{I})$. All in all, using the classical Greedy approximation algorithm for Min Set Cover with approximation ratio $\mathrm{H}(k)$, where $k \in \mathbb{N}$ is the maximum number of elements of a set, we find a feasible set $I' \subseteq I$ such that the corresponding induced solution value approximates $\mathrm{OPT}(\mathcal{I})$ within a ratio of

$$\min\left\{|I|, |J|, |\mathcal{U}| + \mathrm{H}\left(\max_{i \in I} |N(i)|\right) \cdot \frac{q-1}{q}\right\}. \tag{5.23}$$

For discrete, interval, budgeted, and Γ-uncertainty, we can compute induced solutions in polynomial time. Hence, for these cases, we even obtain approximation algorithms. Moreover, if $\mathrm{H}(\max_{i \in I} |N(i)|) < q \cdot |\mathcal{U}|$, this algorithm improves the previous $q|\mathcal{U}|$-approximation.

Conclusion

In this chapter, we considered three different behavior patterns for clients in the setting of a Robust Min q-MSMC problem:

(a) Completely free clients which independently choose a location by which they want to be served (Section 5.1),

(b) free clients that need to adapt to a selection of available locations (Section 5.2), and

(c) clients which reveal a ranking over their adjacent locations and want to be served by their most preferred available location (Section 5.3).

For each of these patterns, a solution to a given instance is also feasible for the corresponding instance of Robust Min q-MSMC. We investigated the complexities of the problems for the non-robust as well as the robust variant and analyzed the restrictions to various classes of uncertainty sets. Our findings are summarized in Table 5.2, Table 5.3, and Table 5.4. While an optimal solution to an instance of Robust Min q-FC is easy to formulate theoretically (although not easy to compute), for Robust Min q-AC and Robust Min q-OC, the concept of induced solutions is of importance. We showed that, for instances of Robust Min q-AC and Robust Min q-OC, it suffices to analyze solutions which are induced by a feasible subset of the locations. Such a feasible set corresponds to a cover of the regions. Furthermore, we presented mixed integer programming formulations for the hard problem variants. For integral polyhedral uncertainty sets, these programs can be reformulated to obtain a compact formulation. In case of a non-compact formulation, constraint generation is a meaningful solution method. Here, the separation problems correspond to instances of Max Robust Sum.

Uncertainty	Robust Min q-FC
General	NP-hard
Discrete	PO
Interval	PO
Budgeted	PO
Multi-budgeted	NP-hard
Ellipsoidal	NP-hard
Γ	PO

Table 5.2.: Summary of results for Robust Min q-FC. All results hold for any fixed $q \in \mathbb{N}_{>0}$.

Uncertainty	Robust Min q-AC	Induced Solution
General	APX-hard	NP-hard
Discrete	APX-hard	PO
Interval	APX-hard	PO
Budgeted	APX-hard	PO
Multi-budgeted	APX-hard	NP-hard
Ellipsoidal	APX-hard	NP-hard
Γ	APX-hard	PO

Table 5.3.: Summary of results for Robust Min q-AC and related problems. All results hold for any fixed $q \in \mathbb{N}_{>0}$.

Uncertainty	Robust Min q-OC	Induced Solution
General	APX-hard	NP-hard
Discrete		
$q \cdot \lvert\mathcal{U}\rvert \leq 1$	PO	PO
$q \cdot \lvert\mathcal{U}\rvert \geq 2$	APX-complete	PO
Interval		
$q \leq 1$	PO	PO
$q \geq 2$	APX-complete	PO
Budgeted	APX-hard	PO
Multi-budgeted	APX-hard	NP-hard
Ellipsoidal	APX-hard	NP-hard
Γ	APX-hard	PO

Table 5.4.: Summary of results for Robust Min q-OC and related problems. If not stated otherwise, the results hold for any fixed $q \in \mathbb{N}_{>0}$.

6. Conclusion and Future Research

This thesis comprises a study of the optimization problem Robust Min q-MSMC. In particular, in the case of discrete uncertainty, we were able to exactly determine the transition from polynomial time solvable instances to hard instances in terms of the fixed value $q \in \mathbb{N}_{>0}$ and the number of present scenarios $|\mathcal{U}|$. For instances with $q \cdot |\mathcal{U}| \leq 2$, we provided strongly polynomial time algorithms depending on the exact values of q and $|\mathcal{U}|$. To show APX-hardness for the remaining instances, the Min 3-Dimensional Cover problem played a particular role as it was shown to be APX-complete. Moreover, the equivalence of the two integer programming formulations $\mathrm{P}(\mathcal{U})$ and $\mathrm{P_s}(\mathcal{U})$, originating from the famous Max-Flow-Min-Cut Theorem, lead to the co-existence of a strictly robust and an adjustable robust formulation for instances of Robust Min q-MSMC. Additionally, it lead to the analysis of the corresponding separation problem as both formulations are non-compact. Surprisingly, even for classes of uncertainty sets that admit for a polynomial time optimization oracle, e.g., budgeted or Γ-uncertainty, verifying the feasibility of a given tentative solution is co-NP-complete. On the contrary, this problem can be solved easily in the non-robust case revealing one of the additional challenges of including uncertainty. Hence, the further identification of classes of uncertainty sets which allow to identify a solution in polynomial time is of great interest. For practical purposes, the development of heuristic approaches for the separation problems might have a great impact on the constraint generation process. Another interesting version of Robust Min q-MSMC may be the consideration of an *absolute* assignment as in the case of Robust Min q-OC, i.e, in each scenario, the demand of a region has to be covered by just one location.

As the structure of the given uncertainty set highly influences the complexity of Robust Min q-MSMC, we developed an approximation algorithm for instances whose uncertainty sets have a polynomial time optimization oracle. Here, we neatly combined results on approximation algorithms for Min Multiset Multicover and Min ARC and excessively exploited the structure of our problem. The difficulty was to ensure a polynomial running time despite the generality of the given uncertainty set. The implicit set structure in Robust Min q-MSMC was the key to overcome this challenge. We were able to bound the performance ratio by $\mathrm{H}(q \cdot |\mathcal{U}|) \cdot \mathcal{O}(\sqrt{|J|})$ in the general case, which can be improved for polynomial time enumerable uncertainty sets. In the latter case, the presented ratio is tight and also optimal – up to additive constants in the ratio – unless P = NP. Thus, the question of tightness of the general ratio remains open. Moreover, approximations or approximation algorithms for other classes of uncertainty sets are still unknown. For instance, our algorithms do not apply to ellipsoidal uncertainty.

To increase the influence of the regions and their clients, we introduced further variants of Robust Min q-MSMC by imposing behavior patterns on the clients. This lead to the problems Robust Min q-FC, Robust Min q-AC, and Robust Min q-OC. Here, we theoretically deduced the unique optimal solution to an instance of Robust Min q-FC and we showed that, for instances of Robust Min q-AC and Robust Min q-OC, it suffices to restrict our analysis to feasible subsets of the locations. These results revealed these problems to be substantially different to Robust Min q-MSMC. Yet, despite this structural outcome for Robust Min q-AC, we proved APX-hardness of the problem even for the single scenario case. In further contrast to Robust Min q-MSMC, we only provided a linear time algorithm for instances of Robust Min q-OC with $q = |\mathcal{U}| = 1$. For general instances, we showed that the problem is APX-hard. Moreover, we analyzed the complexity of Robust Min q-FC, Robust Min q-AC, and Robust Min q-OC when restricted to specific classes of uncertainty sets. We saw that instances of Robust Min q-FC with polynomial time optimization uncertainty can be solved efficiently. This result pertained to discrete, interval, budgeted, and Γ-uncertainty. Moreover, the above mentioned linear time algorithm also applied to instances of Robust Min 1-OC with interval uncertainty. All remaining restrictions were shown to be NP-hard, APX-hard, or even APX-complete. Hence, an extensive analysis of possible approximation algorithms for these variants of Robust Min q-MSMC is meaningful. In particular, these could help in the solution process for large-scale instances. Moreover, the introduction of further variants of Robust Min q-MSMC leading to simpler problems in terms of complexity is appealing.

Besides these theoretical research aspects, new interesting directions are possible due to the mentioned practical application of Robust Min q-MSMC, cf. Example 3.27. For instance, a differentiation between types of regions may be meaningful to account for rural and urban areas. Hence, covering an emergency in a rural area may consume more capacity of an emergency ambulance than covering an urban emergency. This could lead to a different handling of the given value $q \in \mathbb{N}_{>0}$. Moreover, the combination of different sources of uncertainty, similar to [Büs+21], might reveal further potential applications in locating emergency service facilities.

Altogether, with the introduction of Robust Min q-MSMC we have opened a new interesting field of research in robust covering problems which is, by all means, not exhausted yet.

A. Approximating Min Adjustable Robust Covering

In this chapter, we consider the Min ARC problem introduced in Section 2.6.1:

$$\begin{array}{llll} \mathrm{P}_{\mathrm{arc}}(\mathcal{Z}) & \min\limits_{x,\,y(\zeta)} & c^T x + \max\limits_{\zeta \in \mathcal{Z}} d^T y(\zeta) & \\ & \text{s.t.} & Ax + By(\zeta) \geq \zeta & \text{for } \zeta \in \mathcal{Z} \\ & & x, y(\zeta) \geq 0 & \text{for } \zeta \in \mathcal{Z} \\ & & x \in \mathbb{R}^{n_1 - p_1} \times \mathbb{N}^{p_1} & \\ & & y(\zeta) \in \mathbb{R}^{n_2 - p_2} \times \mathbb{N}^{p_2} & \text{for } \zeta \in \mathcal{Z}, \end{array}$$

with $A \in \mathbb{Q}^{m \times n_1}$, $B \in \mathbb{Q}^{m \times n_2}$, $c \in \mathbb{Q}^{n_1}_{\geq 0}$, $d \in \mathbb{Q}^{n_2}_{\geq 0}$, non-negative integers $p_1 \leq n_1$, $p_2 \leq n_2$, and an uncertainty set $\mathcal{Z} \subseteq \mathbb{R}^m_{\geq 0}$. As before we assume that the optimal value $z_{\mathrm{arc}}(\mathcal{Z})$ of $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$ exists. Further, for $j \in \{1, \ldots, m\}$, let

$$\beta^j \in \arg\max_{\zeta \in \mathcal{Z}} \zeta_j .$$

In [BG11], Bertsimas and Goyal show that the uncertainty set

$$\tilde{\mathcal{Z}} := \operatorname{conv}\bigl(2\beta, 2\sqrt{m} \cdot \beta^1, \ldots, 2\sqrt{m} \cdot \beta^m\bigr), \tag{A.1}$$

where scenario β is obtained from Algorithm 4.1, dominates $\mathcal{Z}$. Moreover, if $p_1 = p_2 = 0$, it holds true that $z_{\mathrm{arc}}\bigl(\tilde{\mathcal{Z}}\bigr) \leq 4\sqrt{m} \cdot z_{\mathrm{arc}}(\mathcal{Z})$.

In this section, we provide an adaption of $\tilde{\mathcal{Z}}$ to obtain a ratio of $(4\sqrt{r} + 1)$ if, for $r \in \mathbb{N}_{>0}$, only $r \leq m$ parameters are uncertain, $p_1 \geq 0$, and $p_2 = 0$. Thus, let $\mathcal{Z}_r \subseteq \mathbb{R}^r_{\geq 0}$ for $r \leq m$ be non-empty and compact. Then, we can rewrite $\mathrm{P}_{\mathrm{arc}}(\mathcal{Z})$ as follows:

$$\begin{array}{llll}
\mathrm{P}'_{\mathrm{arc}}(\mathcal{Z}_r) & \min\limits_{x,\, y(\zeta)} & c^T x + \max\limits_{\zeta \in \mathcal{Z}_r} d^T y(\zeta) & \\
& \text{s.t.} & Ax + By(\zeta) \geq \begin{pmatrix} \zeta \\ \bar{\zeta} \end{pmatrix} & \text{for } \zeta \in \mathcal{Z}_r \\
& & x, y(\zeta) \geq 0 & \text{for } \zeta \in \mathcal{Z}_r \\
& & x \in \mathbb{R}^{n_1 - p_1} \times \mathbb{N}^{p_1} & \\
& & y(\zeta) \in \mathbb{R}^{n_2} & \text{for } \zeta \in \mathcal{Z}_r,
\end{array}$$

with $\bar{\zeta} \in \mathbb{Q}^{m-r}_{\geq 0}$ fixed and optimal value $z'_{\mathrm{arc}}(\mathcal{Z}_r) \in \mathbb{R}_{\geq 0}$. The proof of the subsequent lemma goes along the same lines as in [BG11].

Lemma A.1. *For $P'_{arc}(\mathcal{Z}_r)$, it holds true that*

$$z'_{arc}(\mathcal{Z}_0) \leq \left(4\sqrt{r} + 1\right) \cdot z'_{arc}(\mathcal{Z}_r),$$

where $\mathcal{Z}_0$ is a prevailing set corresponding to $\mathcal{Z}_r$, i.e., with β^j for $j \in \{1, \dots, r\}$ and β as in (A.1)*, we have*

$$\mathcal{Z}_0 := \left\{2\beta, \lceil 2\sqrt{r} \rceil \cdot \beta^1, \dots, \lceil 2\sqrt{r} \rceil \cdot \beta^r \right\}.$$

Proof. Let $(x^\star, y^\star)$ be optimal for $\mathrm{P}'_{\mathrm{arc}}(\mathcal{Z}_r)$. Let $v^j := \lceil 2\sqrt{r} \rceil \cdot \beta^j$ for $j \in \{1, \dots, r\}$ and $v^{r+1} := 2\beta$, where $\beta = \sum_{l=1}^{K} u^l$ and $u^l \in \mathcal{Z}_r$ is the scenario found in Step 4 and iteration $l \in \{1, \dots, K\}$ for $K \in \mathbb{N}_{>0}$ of Algorithm 4.1. We define a tentative solution (x, y) to $\mathrm{P}'_{\mathrm{arc}}(\mathcal{Z}_0)$ as follows:

$$\begin{aligned}
x &:= \lceil 4\sqrt{r} \rceil \cdot x^\star, \\
y(v^j) &:= \lceil 4\sqrt{r} \rceil \cdot y^\star(\beta^j) \text{ for } j \in \{1, \dots, r\}, \\
y(v^{r+1}) &:= \frac{\lceil 4\sqrt{r} \rceil}{K} \sum_{l=1}^{K} y^\star(u^l).
\end{aligned}$$

Note that we have $K \leq 2\sqrt{r}$ by [BG11]. Now, we show that (x, y) is feasible for $\mathrm{P}'_{\mathrm{arc}}(\mathcal{Z}_0)$. First, let $j \leq r$. Then, we have

$$Ax + By(v^j) = \lceil 4\sqrt{r} \rceil \cdot \left(Ax^\star + By^\star(\beta^j)\right) \geq \lceil 4\sqrt{r} \rceil \cdot \begin{pmatrix} \beta^j \\ \bar{\zeta} \end{pmatrix} \geq \begin{pmatrix} v^j \\ \bar{\zeta} \end{pmatrix}$$

as $r \geq 1$. For $j = r+1$ we compute

$$\begin{aligned} Ax + By(v^{r+1}) &= \frac{\lceil 4\sqrt{r}\rceil}{K} \sum_{l=1}^{K} \big(Ax^\star + By^\star(u^l)\big) \geq \frac{\lceil 4\sqrt{r}\rceil}{K} \sum_{l=1}^{K} \begin{pmatrix} u^l \\ \bar{\zeta} \end{pmatrix} \\ &\geq \frac{2\sqrt{r}}{K} \begin{pmatrix} 2\beta \\ 2K \cdot \bar{\zeta} \end{pmatrix} \geq \frac{2\sqrt{r}}{K} \begin{pmatrix} v^{r+1} \\ \bar{\zeta} \end{pmatrix} \geq \begin{pmatrix} v^{r+1} \\ \bar{\zeta} \end{pmatrix} \end{aligned}$$

since $1 \leq K \leq 2\sqrt{r}$. For the solution value, we obtain for $j \leq r$:

$$c^T x + d^T y(v^j) = \lceil 4\sqrt{r}\rceil \cdot \big(c^T x^\star + d^T y^\star(\beta^j)\big) \leq \big(4\sqrt{r}+1\big) \cdot z'_{\mathrm{arc}}(\mathcal{Z}_r)$$

as $c, d \geq 0$. Finally, let $j = r+1$. Here, we infer

$$\begin{aligned} c^T x + d^T y(v^{r+1}) &= \frac{\lceil 4\sqrt{r}\rceil}{K} \sum_{l=1}^{K} \big(c^T x^\star + d^T y^\star(u^l)\big) \\ &\leq \frac{\lceil 4\sqrt{r}\rceil}{K} \sum_{l=1}^{K} z'_{\mathrm{arc}}(\mathcal{Z}_r) \\ &\leq \big(4\sqrt{r}+1\big) \cdot z'_{\mathrm{arc}}(\mathcal{Z}_r) \end{aligned}$$

and the claim follows. □

B. APX-hardness of Min $\{k\}$-Domination

In this chapter, we focus on the $\{k\}$-Domination problem and extend the NP-hardness proof of [Gai+03] to an APX-hardness proof. Thus, call the optimization version of the problem Min $\{k\}$-Domination. We provide an L-reduction from Min Set Cover(3), which was also used to show APX-hardness of Min q-MSMC for fixed $q \geq 3$ in Theorem 3.14. In the following, only slight changes to the proof of [Gai+03] are made.

Theorem B.1. *For any fixed $k \in \mathbb{N}_{>0}$, Min $\{k\}$-Domination is* APX*-hard.*

Proof. Let $\mathcal{I}$ be an instance of Min Set Cover(3). This is a finite set $S = \{s_1, \ldots, s_n\}$ and a collection $\mathcal{C} = \{C_1, \ldots, C_m\}$ of subsets of S with $|C_l| \leq 3$ for $l \in \{1, \ldots, m\}$. We construct the graph G of the instance $\mathcal{I}'$ of Min $\{k\}$-Domination as follows: For each element $s_i \in S$, we build k paths of three vertices each. Label the vertices a_{ij}, b_{ij}, and c_{ij} for $j \in \{1, \ldots, k\}$. Then, the c-vertices are connected to a common vertex d_i. We refer to this element gadget as G_i with vertex set $V(G_i)$. Furthermore, for each set C_l, we have a vertex u_l. All u-vertices are connected to a common vertex v and v has another adjacent vertex w. Finally, consider an element $s_i \in S$ and a set $C_l \in \mathcal{C}$. If $s_i \in C_l$, we connect all a-vertices of gadget G_i with the vertex u_l.

Given an optimal solution to $\mathcal{I}$, i.e., a cover $\mathcal{C}' \subseteq \mathcal{C}$, we define a solution to $\mathcal{I}'$ as follows: Set $x(a_{ij}) \coloneqq 0$, $x(b_{ij}) \coloneqq k - 1$, $x(c_{ij}) \coloneqq 1$, and $x(d_i) \coloneqq 0$ for all i, j. Furthermore, set $x(v) \coloneqq k$, $x(w) \coloneqq 0$, and $x(u_l) \coloneqq 1$ if $C_l \in \mathcal{C}'$ and zero otherwise. It is easy to see that x is a solution to $\mathcal{I}'$ with value $k^2|S| + k + |\mathcal{C}'|$. Thus, we get

$$\begin{aligned} \mathrm{OPT}(\mathcal{I}') &\leq k^2|S| + k + \mathrm{OPT}(\mathcal{I}) \\ &\leq \left(3k^2 + k + 1\right) \cdot \mathrm{OPT}(\mathcal{I}) \end{aligned} \tag{B.1}$$

using that $|S| \leq 3 \cdot \mathrm{OPT}(\mathcal{I})$ as $\mathcal{C}'$ covers S and $|C_l| \leq 3$ for $l \in \{1, \ldots, m\}$.

On the other hand, let x be a solution to $\mathcal{I}'$. First of all, we need to adjust x slightly. As $x(w) + x(v) \geq k$, we do not lose feasibility if we set $x(w) := 0$ and $x(v) := k$. Moreover, as $x(a_{ij}) + x(b_{ij}) + x(c_{ij}) \geq k$ for every i, j, we get that $x(G_i) \geq k^2$ for every $i \in \{1, \dots, n\}$. If $x(G_i) = k^2$, we must have $x(a_{ij}) + x(b_{ij}) + x(c_{ij}) = k$ for every $j \in \{1, \dots, k\}$ so that $x(d_i) = 0$. Therefore, we have that $x(b_{ij}) + x(c_{ij}) = k$ and $x(a_{ij}) = 0$. If $x(G_i) > k^2$, we can reassign $x(a_{ij}) := 0$, $x(b_{ij}) := k-1$, and $x(c_{ij}) := 1$ for all $j \in \{1, \dots, k\}$ and $x(d_i) := 0$. Furthermore, we choose one subset C_l with $s_i \in C_l$ and set $x(u_l) := 1$. Our adjusted solution x is still feasible and its solution value did not increase. Thus, we can assume that $\mathrm{SOL}(\mathcal{I}', x) \geq k^2|S| + k + r$ for some $r \in \mathbb{N}$. Consider the set $\mathcal{C}' := \{C_l \in \mathcal{C} \colon x(u_l) = 1\}$ with $|\mathcal{C}'| = r$. We claim that $\mathcal{C}'$ is a cover for S. Let $s_i \in S$. Then, there is a vertex $c_{ij} \in V(G_i)$ with $x(c_{ij}) > 0$ as otherwise the vertex d_i is not dominated. Thus, by our construction of x we have $x(b_{ij}) < k$. As $x(a_{ij}) = 0$ there must be some vertex u_l adjacent to a_{ij} with $x(u_l) = 1$. In total, we obtain

$$\begin{aligned}
\mathrm{SOL}(\mathcal{I}, \mathcal{C}') - \mathrm{OPT}(\mathcal{I}) &= r - \mathrm{OPT}(\mathcal{I}) \\
&= k^2|S| + k + r - \big(\mathrm{OPT}(\mathcal{I}) + k^2|S| + k\big) \\
&\overset{(\mathrm{B.1})}{\leq} \mathrm{SOL}(\mathcal{I}', x) - \mathrm{OPT}(\mathcal{I}').
\end{aligned}$$

As k is fixed, this constitutes an L-reduction from Min Set Cover(3) to Min $\{k\}$-Domination. As Min Set Cover(3) is APX-complete, the claim follows. ☐

Corollary B.2. *For any fixed $k \in \mathbb{N}_{>0}$, $\{k\}$-Domination is strongly* NP-*complete.*

Proof. From the proof of Theorem B.1, we see that a solution to $\mathcal{I}$ with value at most $r \in \mathbb{N}$ leads to a solution to $\mathcal{I}'$ with value at most $k^2|S| + k + r$ in polynomial time and vice versa. As $\{k\}$-Domination is also contained in NP, we obtain the desired result. ☐

C. Problem Index

In this chapter, we give a full, alphabetically sorted list of the decision problems encountered in this thesis. Note that the corresponding optimization problems can be derived from the list with little effort and are therefore not mentioned separately.

b-Edge Cover
Instance: A graph $G = (V, E)$, a vector $b \in \mathbb{N}_{>0}^{|V|}$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there a b-edge cover $x \in \mathbb{N}^{|E|}$ of size at most B, i.e., $x(E) \leq B$ and $\sum_{e\in\delta(v)} \mathrm{m}(v, e) \cdot x_e \geq b_v$ for $v \in V$?

b-Matching
Instance: A graph $G = (V, E)$, a vector $b \in \mathbb{N}_{>0}^{|V|}$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there a b-matching $x \in \mathbb{N}^{|E|}$ of size at least B, i.e., $x(E) \geq B$ and $\sum_{e\in\delta(v)} \mathrm{m}(v, e) \cdot x_e \leq b_v$ for $v \in V$?

Clique
Instance: A simple graph $G = (V, E)$ and a positive integer $B \leq |V|$.
Question: Does G contain a clique of size at least B, i.e., a subset $V' \subseteq V$ with $|V'| \geq B$ such that every two distinct vertices in V' are adjacent in G?

Covering Integer Program (CIP$_\infty$)
Instance: A matrix $A \in \mathbb{Q}_{\geq 0}^{m\times n}$, vectors $b \in \mathbb{Q}_{>0}^{m}$, $c \in \mathbb{N}_{>0}^{n}$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^n$ with $c^T x \leq B$ and $Ax \geq b$?

Covering Integer Program with multiplicity constraints (CIP)
Instance: A matrix $A \in \mathbb{Q}_{\geq 0}^{m \times n}$, vectors $b \in \mathbb{Q}_{>0}^{m}$, $c, d \in \mathbb{N}_{>0}^{n}$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^n$ with $c^T x \leq B$, $x \leq d$, and $Ax \geq b$?

Cut
Instance: A network $G = (V, R, c)$, two distinct vertices $s, t \in V$, and a rational $B \in \mathbb{Q}_{>0}$.
Question: Does G contain an s-t-cut with cut capacity at most B, i.e., a partition of V into two sets S, T with $s \in S$ and $t \in T$ such that $c(S, T) \leq B$?

Dominating Set
Instance: A simple graph $G = (V, E)$ and an integer $B \in \mathbb{N}_{>0}$.
Question: Does G contain a dominating set of size at most B, i.e., a subset $V' \subseteq V$ with $|V'| \leq B$ such that, for all $u \in V \setminus V'$, there is $v \in V'$ dominating u?

Dominating Set(K)
Instance: A simple graph $G = (V, E)$ with $\Delta \leq K$ and an integer $B \in \mathbb{N}_{>0}$.
Question: Does G contain a dominating set of size at most B, i.e., a subset $V' \subseteq V$ with $|V'| \leq B$ such that, for all $u \in V \setminus V'$, there is $v \in V'$ dominating u?

Edge Cover
Instance: A simple graph $G = (V, E)$ and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there an edge cover for G of size at most B, i.e., a subset $E' \subseteq E$ with $|E'| \leq B$ such that every vertex $v \in V$ is incident to some edge in E'?

Exact Cover by 3-Sets
Instance: A set S with $|S| = 3r$ for $r \in \mathbb{N}_{>0}$ and a collection $\mathcal{C}$ of 3-element subsets of S.
Question: Does $\mathcal{C}$ contain an exact cover for S, i.e., a subcollection $\mathcal{C}' \subseteq \mathcal{C}$ such that every element of S appears in exactly one set of $\mathcal{C}'$?

Feasibility for Robust Min q-MSMC (Feasibility)
Instance: Finite sets I, J with $I \cap J = \varnothing$, a vector $\bar{x} \in \mathbb{N}^{|I|}$, an uncertainty set $\mathcal{U} \subseteq \mathbb{N}^{|J|}$, and a bipartite graph $G = (I \cup J, E)$.
Question: Does $q \cdot \bar{x}(N(S)) \geq \max_{\xi \in \mathcal{U}} \xi(S)$ hold for every subset $S \subseteq J$?

Flow
Instance: A network $G = (V, R, c)$, two distinct vertices $s, t \in V$, and a rational $B \in \mathbb{Q}_{>0}$.
Question: Is there an s-t-flow f with flow value at least B, i.e., a mapping $f\colon R \to \mathbb{Q}_{\geq 0}$ with $\mathrm{val}(f) \geq B$ such that, for $v \in V \setminus \{s, t\}$, it holds true that $f(\delta^+(v)) = f(\delta^-(v))$, and, for $r \in R$, we have $f(r) \leq c(r)$?

Independent Set
Instance: A simple graph $G = (V, E)$ and a positive integer $B \leq |V|$.
Question: Does G contain an independent set of size at least B, i.e., a subset $V' \subseteq V$ with $|V'| \geq B$ and no two vertices in V' are neighbors?

Independent Set(K)
Instance: A simple graph $G = (V, E)$ with $\Delta \leq K$ and a positive integer $B \leq |V|$.
Question: Does G contain an independent set of size at least B, i.e., a subset $V' \subseteq V$ with $|V'| \geq B$ and no two vertices in V' are neighbors?

$\{k\}$-Domination
Instance: A simple graph $G = (V, E)$ and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|V|}$ such that $x(V) \leq B$ and $x(N[v]) \geq k$ for $v \in V$?

Knapsack
Instance: A finite set U, a size $s_u \in \mathbb{N}_{>0}$ and a profit $p_u \in \mathbb{N}_{>0}$ for each $u \in U$, and two integers $B, K \in \mathbb{N}_{>0}$.
Question: Is there a subset $U' \subseteq U$ such that $s(U') \leq B$ as well as $p(U') \geq K$?

Matching
Instance: A simple graph $G = (V, E)$.
Question: Does G contain a perfect matching, i.e., a subset $E' \subseteq E$ with $|E'| = {}^1\!/_2|V|$ such that any two edges of E' do not share a common end vertex?

Multiset Multicover
Instance: A matrix $A \in \mathbb{N}^{m \times n}$, vectors $b \in \mathbb{N}^m_{>0}$, $c \in \mathbb{N}^n_{>0}$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^n$ with $c^T x \leq B$ and $Ax \geq b$?

Multiset Multicover(K)
Instance: A matrix $A \in \mathbb{N}^{m \times n}$ with $\sum_{i=1}^m a_{ij} \leq K$ for every $j \in \{1, \ldots, n\}$, vectors $b \in \mathbb{N}^m_{>0}$, $c \in \mathbb{N}^n_{>0}$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^n$ with $c^T x \leq B$ and $Ax \geq b$?

Order-Domination
Instance: A simple graph $G = (V, E)$ with $n \in \mathbb{N}_{>0}$ vertices and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|V|}$ such that $x(V) \leq B$ and $x(N[v]) \geq 2\sqrt{n}$ for $v \in V$?

Poly-Domination
Instance: A simple graph $G = (V, E)$ with $n \in \mathbb{N}_{>0}$ vertices, a positive integer k that is $\mathcal{O}(n^2)$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|V|}$ such that $x(V) \leq B$ and $x(N[v]) \geq k$ for $v \in V$?

q-Adapting Clients (q-AC)
Instance: Finite sets I, J with $I \cap J = \varnothing$, a demand value $d_j \in \mathbb{N}_{>0}$ for each $j \in J$, a bipartite graph $G = (I \cup J, E)$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for every $j \in J$, there is $i \in N(j)$ with $x_i \geq 1$ and, for all $y \in \mathbb{N}^{|I| \times |J|}$ with $\sum_{i \in N(j)\colon x_i \geq 1} y_{ij} = d_j$ for $j \in J$, it holds that

$$\sum_{j \in N(i)} y_{ij} \leq q \cdot x_i \text{ for } i \in I \text{ with } x_i \geq 1?$$

q-Free Clients (q-FC)
Instance: Finite sets I, J with $I \cap J = \varnothing$, a demand value $d_j \in \mathbb{N}_{>0}$ for each $j \in J$, a bipartite graph $G = (I \cup J, E)$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for all $y \in \mathbb{N}^{|I| \times |J|}$ with $\sum_{i \in N(j)} y_{ij} = d_j$ for $j \in J$, it holds that

$$\sum_{j \in N(i)} y_{ij} \leq q \cdot x_i \text{ for } i \in I?$$

q-Multiset Multicover (q-MSMC)
Instance: Finite sets I, J with $I \cap J = \varnothing$, a demand value $d_j \in \mathbb{N}_{>0}$ for each $j \in J$, a bipartite graph $G = (I \cup J, E)$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that there exists $y \in \mathbb{N}^{|I| \times |J|}$ satisfying

$$\sum_{i \in N(j)} y_{ij} \geq d_j \text{ for } j \in J \text{ and } \sum_{j \in N(i)} y_{ij} \leq q \cdot x_i \text{ for } i \in I?$$

q-Ordered Clients (q-OC)
Instance: Finite sets I, J with $I \cap J = \varnothing$, a demand value $d_j \in \mathbb{N}_{>0}$ for each $j \in J$, a bipartite graph $G = (I \cup J, E)$, a preference order σ_j over $N(j)$ for each $j \in J$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for every $j \in J$, there is $i \in N(j)$ with $x_i \geq 1$ and, for $y \in \mathbb{B}^{|I| \times |J|}$ defined by

$$y_{ij} \coloneqq \begin{cases} 1, & \text{if } i \in \arg\min \left\{ \sigma_j^{-1}(i') \colon i' \in N(j) \wedge x_{i'} \geq 1 \right\}, \\ 0, & \text{otherwise}, \end{cases}$$

for $i \in I$, $j \in J$, it holds true that $\sum_{j \in N(i)} d_j \cdot y_{ij} \leq q \cdot x_i$ for $i \in I$?

Quadratic Congruences
Instance: Integers $a, b, c \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}_{>0}$ with $x < c$ and $x^2 \equiv a \mod b$?

Robust q-Adapting Clients (Robust q-AC)
Instance: Finite sets I, J with $I \cap J = \varnothing$, an uncertainty set $\mathcal{U} \subseteq \mathbb{N}^{|J|}$, a bipartite graph $G = (I \cup J, E)$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for every $j \in J$, there is $i \in N(j)$ with $x_i \geq 1$ and, for all $\xi \in \mathcal{U}$ and for all $y \in \mathbb{N}^{|I| \times |J|}$ with $\sum_{i \in N(j) \colon x_i \geq 1} y_{ij} = \xi_j$ for $j \in J$, it holds that

$$\sum_{j \in N(i)} y_{ij} \leq q \cdot x_i \text{ for } i \in I \text{ with } x_i \geq 1?$$

Robust q-Free Clients (Robust q-FC)
Instance: Finite sets I, J with $I \cap J = \varnothing$, an uncertainty set $\mathcal{U} \subseteq \mathbb{N}^{|J|}$, a bipartite graph $G = (I \cup J, E)$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for all $\xi \in \mathcal{U}$ and for all $y \in \mathbb{N}^{|I| \times |J|}$ with $\sum_{i \in N(j)} y_{ij} = \xi_j$ for $j \in J$, it holds that

$$\sum_{j \in N(i)} y_{ij} \leq q \cdot x_i \text{ for } i \in I?$$

Robust q-Multiset Multicover (Robust q-MSMC)
Instance: Finite sets I, J with $I \cap J = \varnothing$, an uncertainty set $\mathcal{U} \subseteq \mathbb{N}^{|J|}$, a bipartite graph $G = (I \cup J, E)$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for every subset $S \subseteq J$ and every scenario $\xi \in \mathcal{U}$, we have $q \cdot x(N(S)) \geq \xi(S)$?

Robust q-MSMC with affine policy
Instance: Finite sets I, J with $I \cap J = \varnothing$, an uncertainty set $\mathcal{U} \subseteq \mathbb{N}^{|J|}$, a bipartite graph $G = (I \cup J, E)$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Are there $x \in \mathbb{N}^{|I|}$, $W \in \mathbb{R}^{|I||J| \times |J|}$, and $w \in \mathbb{R}^{|J|}$ with $x(I) \leq B$ such that, for all $\xi \in \mathcal{U}$, $y(\xi) := W\xi + w \geq 0$ and

$$\sum_{i \in N(j)} y(\xi)_{ij} \geq \xi_j \text{ for } j \in J \text{ and } \sum_{j \in N(i)} y(\xi)_{ij} \leq q \cdot x_i \text{ for } i \in I?$$

Robust q-MSMC with dominating uncertainty
Instance: Finite sets I, J with $I \cap J = \varnothing$, an uncertainty set $\mathcal{U} \subseteq \mathbb{N}^{|J|}$, generating vectors $\beta, \beta^1, \ldots, \beta^{|J|}$ corresponding to $\mathcal{U}$, a bipartite graph $G = (I \cup J, E)$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for every subset $S \subseteq J$ and every scenario ξ in the prevailing set generated by $\beta, \beta^1, \ldots, \beta^{|J|}$, we have $q \cdot x(N(S)) \geq \xi(S)$?

Robust q-Ordered Clients (Robust q-OC)
Instance: Finite sets I, J with $I \cap J = \varnothing$, an uncertainty set $\mathcal{U} \subseteq \mathbb{N}^{|J|}$, a bipartite graph $G = (I \cup J, E)$, a preference order σ_j over $N(j)$ for each $j \in J$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|I|}$ with $x(I) \leq B$ such that, for every $j \in J$, there is $i \in I$ with $x_i \geq 1$ and the assignment matrix $y := y(I_x) \in \mathbb{B}^{|I| \times |J|}$ satisfies $\sum_{j \in N(i)} \xi_j \cdot y_{ij} \leq q \cdot x_i$ for $i \in I, \xi \in \mathcal{U}$?

Robust Sum
Instance: An uncertainty set $\mathcal{U} \subseteq \mathbb{N}^n$, a subset $S \subseteq \{1, \ldots, n\}$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there a scenario $\xi \in \mathcal{U}$ with $\xi(S) \geq B$?

Separation for Robust Min q-MSMC (Separation)
Instance: Finite sets I, J with $I \cap J = \varnothing$, a vector $\bar{x} \in \mathbb{N}^{|I|}$, an uncertainty set $\mathcal{U} \subseteq \mathbb{N}^{|J|}$, and a bipartite graph $G = (I \cup J, E)$.
Question: Is there a subset $S \subseteq J$ such that $q \cdot \bar{x}(N(S)) < \max_{\xi \in \mathcal{U}} \xi(S)$?

Set Cover
Instance: A collection $\mathcal{C}$ of subsets of a finite set S and an integer $B \in \mathbb{N}_{>0}$.
Question: Does $\mathcal{C}$ contain a (set) cover for S of size at most B, i.e., a subcollection $\mathcal{C}' \subseteq \mathcal{C}$ with $|\mathcal{C}'| \leq B$ such that every element in S is contained in at least one set of $\mathcal{C}'$?

Set Cover(K)
Instance: A collection $\mathcal{C}$ of subsets of a finite set S with $|C| \leq K$ for every $C \in \mathcal{C}$ and an integer $B \in \mathbb{N}_{>0}$.
Question: Does $\mathcal{C}$ contain a (set) cover for S of size at most B, i.e., a subcollection $\mathcal{C}' \subseteq \mathcal{C}$ with $|\mathcal{C}'| \leq B$ such that every element in S is contained in at least one set of $\mathcal{C}'$?

Set Multicover
Instance: A collection $\mathcal{C}$ of subsets of a finite set S, a demand value $b_i \in \mathbb{N}_{>0}$ for each $s_i \in S$, a weight c_j for each $C_j \in \mathcal{C}$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|\mathcal{C}|}$ with $c^T x \leq B$ and, for each element $s_i \in S$, we have $\sum_{C_j \in \mathcal{C}} \mathrm{m}(s_i, C_j) \cdot x_j \geq b_i$?

Set Multicover(K)
Instance: A collection $\mathcal{C}$ of subsets of a finite set S with $|C_j| \leq K$ for every $C_j \in \mathcal{C}$, a demand value $b_i \in \mathbb{N}_{>0}$ for each $s_i \in S$, a weight c_j for each $C_j \in \mathcal{C}$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there $x \in \mathbb{N}^{|\mathcal{C}|}$ with $c^T x \leq B$ and, for each element $s_i \in S$, we have $\sum_{C_j \in \mathcal{C}} \mathrm{m}(s_i, C_j) \cdot x_j \geq b_i$?

3-Dimensional Cover
Instance: Disjoint sets W, X, Y, a subset $\mathcal{C} \subseteq W \times X \times Y$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Does $\mathcal{C}$ contain a cover for $W \cup X \cup Y$ of size at most B, i.e., a subset $\mathcal{C}' \subseteq \mathcal{C}$ with $|\mathcal{C}'| \leq B$ such that every element $z \in W \cup X \cup Y$ appears in at least one element (w, x, y) of $\mathcal{C}'$?

3-Dimensional Matching
Instance: Disjoint sets W, X, Y with $r \in \mathbb{N}_{>0}$ elements each and a subset $M \subseteq W \times X \times Y$.
Question: Does M contain a perfect matching, i.e., a subset $M' \subseteq M$ with $|M'| = r$ such that no two elements of M' agree in any coordinate?

3-SAT
Instance: A set U of variables, a collection C of disjunctive clauses of at most three literals.
Question: Is there a truth assignment $f\colon U \to \{\textsc{true}, \textsc{false}\}$ fulfilling all clauses?

3-SAT(K)
Instance: A set U of variables, a collection C of disjunctive clauses of at most three literals, where each variable appears at most K times, and an integer $B \in \mathbb{N}_{>0}$.
Question: Is there a truth assignment $f\colon U \to \{\textsc{true}, \textsc{false}\}$ fulfilling all clauses?

2-Dimensional Cover
Instance: Disjoint sets W and X, a subset $\mathcal{C} \subseteq W \times X$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Does $\mathcal{C}$ contain a cover for $W \cup X$ of size at most B, i.e., a subset $\mathcal{C}' \subseteq \mathcal{C}$ with $|\mathcal{C}'| \leq B$ such that every element $z \in W \cup X$ appears in at least one element (w, x) of $\mathcal{C}'$?

Weighted Set Cover
Instance: A collection $\mathcal{C}$ of subsets of a finite set S, a weight c_j for each $C_j \in \mathcal{C}$, and an integer $B \in \mathbb{N}_{>0}$.
Question: Does $\mathcal{C}$ contain a (set) cover for S of weight at most B, i.e., a subcollection $\mathcal{C}' \subseteq \mathcal{C}$ with $c(\mathcal{C}') \leq B$ such that every element in S is contained in at least one set of $\mathcal{C}'$?

Bibliography

[AB17] Sanjeev Arora and Boaz Barak. *Computational complexity: A modern approach*. Cambridge University Press, 2017. ISBN: 0521424267.

[AK00] Paola Alimonti and Viggo Kann. "Some APX-completeness results for cubic graphs." In: *Theoretical Computer Science* 237.1-2 (Apr. 2000), pp. 123–134. DOI: 10.1016/s0304-3975(98)00158-3.

[Alo+09] Noga Alon et al. "The online set cover problem." In: *SIAM Journal on Computing* 39.2 (Jan. 2009), pp. 361–370. DOI: 10.1137/060661946.

[AMO93] Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. *Network flows: Theory, algorithms, and applications*. Prentice Hall, 1993. ISBN: 9780136175490.

[AP05] Elodie Adida and Georgia Perakis. "A robust optimization approach to dynamic pricing and inventory control with no backorders." In: *Mathematical Programming* 107.1-2 (Dec. 2005), pp. 97–129. DOI: 10.1007/s10107-005-0681-5.

[Ata06] Alper Atamtürk. "Strong formulations of robust mixed 0–1 programming." In: *Mathematical Programming* 108.2-3 (Apr. 2006), pp. 235–250. DOI: 10.1007/s10107-006-0709-5.

[Aus+02] Giorgio Ausiello et al. *Complexity and approximation*. Springer Berlin Heidelberg, Dec. 2002. ISBN: 3540654313.

[BB14] Dimitris Bertsimas and Hoda Bidkhori. "On the performance of affine policies for two-stage adaptive optimization: A geometric perspective." In: *Mathematical Programming* 153.2 (Sept. 2014), pp. 577–594. DOI: 10.1007/s10107-014-0818-5.

[BC10] Dimitris Bertsimas and Constantine Caramanis. "Finite adaptability in multistage linear optimization." In: *IEEE Transactions on Automatic Control* 55.12 (Dec. 2010), pp. 2751–2766. DOI: 10.1109/tac.2010.2049764.

[BE85] Reuven Bar-Yehuda and Shimon Even. "A local-ratio theorem for approximating the weighted vertex cover problem." In: *Analysis and Design of Algorithms for Combinatorial Problems*. Vol. 109. Elsevier BV, 1985, pp. 27–45. DOI: 10.1016/s0304-0208(08)73101-3.

[Ben+04] Aharon Ben-Tal et al. "Adjustable robust solutions of uncertain linear programs." In: *Mathematical Programming* 99.2 (Mar. 2004), pp. 351–376. DOI: 10.1007/s10107-003-0454-y.

[Ben+05] Aharon Ben-Tal et al. "Retailer-supplier flexible commitments contracts: A robust optimization approach." In: *Manufacturing & Service Operations Management* 7.3 (July 2005), pp. 248–271. DOI: 10.1287/msom.1050.0081.

[Ben+11] Aharon Ben-Tal et al. "Robust optimization for emergency logistics planning: Risk mitigation in humanitarian relief supply chains." In: *Transportation Research Part B: Methodological* 45.8 (Sept. 2011), pp. 1177–1189. DOI: 10.1016/j.trb.2010.09.002.

[Ben62] Jacques F. Benders. "Partitioning procedures for solving mixed-variables programming problems." In: *Numerische Mathematik* 4.1 (Dec. 1962), pp. 238–252. DOI: 10.1007/bf01386316.

[Beu94] Albrecht Beutelspacher. *Lineare algebra*. Vieweg+Teubner Verlag, 1994. DOI: 10.1007/978-3-322-89448-9.

[BF06] Alexandre Belloni and Robert M. Freund. "On the symmetry function of a convex set." In: *Mathematical Programming* 111.1-2 (Dec. 2006), pp. 57–93. DOI: 10.1007/s10107-006-0074-4.

[BF95] Piotr Berman and Toshihiro Fujito. "On approximation properties of the independent set problem for degree 3 graphs." In: *Algorithms and Data Structures*. Vol. 955. Lecture Notes in Computer Science. Springer Berlin Heidelberg, Aug. 1995, pp. 449–460. DOI: 10.1007/3-540-60220-8_84.

[BG10] Dimitris Bertsimas and Vineet Goyal. "On the power of robust solutions in two-stage stochastic and adaptive optimization problems." In: *Mathematics of Operations Research* 35.2 (May 2010), pp. 284–305. DOI: 10.1287/moor.1090.0440.

[BG11] Dimitris Bertsimas and Vineet Goyal. "On the power and limitations of affine policies in two-stage adaptive optimization." In: *Mathematical Programming* 134.2 (Feb. 2011), pp. 491–531. DOI: 10.1007/s10107-011-0444-4.

[BGE19] Aharon Ben-Tal, Vineet Goyal, and Omar El Housni. "A tractable approach for designing piecewise affine policies in two-stage adjustable robust optimization." In: *Mathematical Programming* 182.1-2 (Mar. 2019), pp. 57–102. DOI: 10.1007/s10107-019-01385-0.

[BGN09] Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi Nemirovski. *Robust optimization.* Princeton University Press, Aug. 2009. DOI: 10.1515/9781400831050.

[BGS11] Dimitris Bertsimas, Vineet Goyal, and Xu Andy Sun. "A geometric characterization of the power of finite adaptability in multistage stochastic and adaptive optimization." In: *Mathematics of Operations Research* 36.1 (Feb. 2011), pp. 24–54. DOI: 10.1287/moor.1110.0482.

[BN00] Aharon Ben-Tal and Arkadi Nemirovski. "Robust solutions of linear programming problems contaminated with uncertain data." In: *Mathematical Programming* 88.3 (Sept. 2000), pp. 411–424. DOI: 10.1007/pl00011380.

[BN05] Niv Buchbinder and Joseph Naor. "Online primal-dual algorithms for covering and packing problems." In: *Algorithms – ESA 2005.* Vol. 3669. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2005, pp. 689–701. DOI: 10.1007/11561071_61.

[BN98] Aharon Ben-Tal and Arkadi Nemirovski. "Robust convex optimization." In: *Mathematics of Operations Research* 23.4 (Nov. 1998), pp. 769–805. DOI: 10.1287/moor.23.4.769.

[BN99] Aharon Ben-Tal and Arkadi Nemirovski. "Robust solutions of uncertain linear programs." In: *Operations Research Letters* 25.1 (Aug. 1999), pp. 1–13. DOI: 10.1016/s0167-6377(99)00016-4.

[BPP19] Marin Bougeret, Artur Alves Pessoa, and Michael Poss. "Robust scheduling with budgeted uncertainty." In: *Discrete Applied Mathematics* 261 (May 2019), pp. 93–107. DOI: `10.1016/j.dam.2018.07.001`.

[BR02] Patrizia Beraldi and Andrzej Ruszczyński. "The probabilistic set-covering problem." In: *Operations Research* 50.6 (Dec. 2002), pp. 956–967. DOI: `10.1287/opre.50.6.956.345`.

[BS03] Dimitris Bertsimas and Melvyn Sim. "Robust discrete optimization and network flows." In: *Mathematical Programming* 98.1-3 (Sept. 2003), pp. 49–71. DOI: `10.1007/s10107-003-0396-4`.

[BS04] Dimitris Bertsimas and Melvyn Sim. "The price of robustness." In: *Operations Research* 52.1 (Feb. 2004), pp. 35–53. DOI: `10.1287/opre.1030.0065`.

[Büs+21] Christina Büsing et al. "Robust strategic planning for mobile medical units with steerable and unsteerable demands." In: *European Journal of Operational Research* 295.1 (Nov. 2021), pp. 34–50. DOI: `10.1016/j.ejor.2021.02.037`.

[BZ08] Eli Berger and Ran Ziv. "A note on the edge cover number and independence number in hypergraphs." In: *Discrete Mathematics* 308.12 (June 2008), pp. 2649–2654. DOI: `10.1016/j.disc.2007.05.006`.

[Cán+07] Lázaro Cánovas et al. "A strengthened formulation for the simple plant location problem with order." In: *Operations Research Letters* 35.2 (Mar. 2007), pp. 141–150. DOI: `10.1016/j.orl.2006.01.012`.

[Cha+18] André Chassein et al. "On recoverable and two-stage robust selection problems with budgeted uncertainty." In: *European Journal of Operational Research* 265.2 (Mar. 2018), pp. 423–436. DOI: `10.1016/j.ejor.2017.08.013`.

[CHS16] Antares Chen, David G. Harris, and Aravind Srinivasan. "Partial resampling to approximate covering integer programs." In: *Proceedings of the 27th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '16)*. SIAM, Dec. 2016, pp. 1984–2003. DOI: `10.1137/1.9781611974331.ch139`.

[Chv79] Vašek Chvátal. "A greedy heuristic for the set-covering problem." In: *Mathematics of Operations Research* 4.3 (Aug. 1979), pp. 233–235. DOI: 10.1287/moor.4.3.233.

[CNS98] Sebastián Ceria, Paolo Nobili, and Antonio Sassano. "A Lagrangian-based heuristic for large-scale set covering problems." In: *Mathematical Programming* 81.2 (Apr. 1998), pp. 215–228. DOI: 10.1007/bf01581106.

[Com21] Martin Comis. "Robust primary care systems." PhD thesis. Rheinisch-Westfälische Technische Hochschule Aachen, 2021.

[Cor+09] Thomas H. Cormen et al. *Introduction to algorithms.* Chapter 9. The MIT Press, July 2009. ISBN: 0262033844.

[CQ19] Chandra Chekuri and Kent Quanrud. "On approximating (sparse) covering integer programs." In: *Proceedings of the 30th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '19).* SIAM, Jan. 2019, pp. 1596–1615. DOI: 10.1137/1.9781611975482.97.

[Cre97] Pierluigi Crescenzi. "A short guide to approximation preserving reductions." In: *Proceedings of the 12th Annual IEEE Conference on Computational Complexity (CCC '97).* IEEE Computer Society, June 1997, pp. 262–273. DOI: 10.1109/ccc.1997.612321.

[CTF00] Alberto Caprara, Paolo Toth, and Matteo Fischetti. "Algorithms for the set covering problem." In: *Annals of Operations Research* 98 (2000), pp. 353–371. DOI: 10.1023/a:1019225027893.

[CZ09] Xin Chen and Yuhan Zhang. "Uncertain linear programs: Extended affinely adjustable robust counterparts." In: *Operations Research* 57.6 (Dec. 2009), pp. 1469–1482. DOI: 10.1287/opre.1080.0605.

[DDS05] Guy Desaulniers, Jacques Desrosiers, and Marius M. Solomon, eds. *Column generation.* Springer US, 2005. DOI: 10.1007/b135457.

[Den+10] Brian T. Denton et al. "Optimal allocation of surgery blocks to operating rooms under uncertainty." In: *Operations Research* 58.4-part-1 (Aug. 2010), pp. 802–816. DOI: 10.1287/opre.1090.0791.

[Dha+05] Kedar Dhamdhere et al. “How to pay, come what may: Approximation algorithms for demand-robust covering problems.” In: *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS ’05)*. IEEE Computer Society, 2005, pp. 367–376. DOI: `10.1109/sfcs.2005.42`.

[DI15] Erick Delage and Dan A. Iancu. “Robust multistage decision making.” In: *The Operations Research Revolution*. INFORMS, Sept. 2015, pp. 20–46. DOI: `10.1287/educ.2015.0139`.

[Die17] Reinhard Diestel. *Graph theory*. Springer Berlin Heidelberg, 2017. DOI: `10.1007/978-3-662-53622-3`.

[Dob82] Gregory Dobson. “Worst-case analysis of greedy heuristics for integer programming with nonnegative data.” In: *Mathematics of Operations Research* 7.4 (Nov. 1982), pp. 515–531. DOI: `10.1287/moor.7.4.515`.

[DS14] Irit Dinur and David Steurer. “Analytical approach to parallel repetition.” In: *Proceedings of the 46th Annual ACM Symposium on Theory of Computing (STOC ’14)*. ACM, May 2014, pp. 624–633. DOI: `10.1145/2591796.2591884`.

[EL97] Laurent El Ghaoui and Hervé Lebret. “Robust solutions to least-squares problems with uncertain data.” In: *SIAM Journal on Matrix Analysis and Applications* 18.4 (Oct. 1997), pp. 1035–1064. DOI: `10.1137/s0895479896298130`.

[EOL98] Laurent El Ghaoui, Francois Oustry, and Hervé Lebret. “Robust solutions to uncertain semidefinite programs.” In: *SIAM Journal on Optimization* 9.1 (Jan. 1998), pp. 33–52. DOI: `10.1137/s1052623496305717`.

[Fei+07] Uriel Feige et al. “Robust combinatorial optimization with exponential scenarios.” In: *Integer Programming and Combinatorial Optimization*. Springer Berlin Heidelberg, 2007, pp. 439–453. DOI: `10.1007/978-3-540-72792-7_33`.

[FM12] Matteo Fischetti and Michele Monaci. “Cutting plane versus compact formulations for uncertain (integer) linear programs.” In: *Mathematical Programming Computation* 4.3 (Apr. 2012), pp. 239–273. DOI: `10.1007/s12532-012-0039-y`.

[FW82] Marshall L. Fisher and Laurence A. Wolsey. "On the greedy heuristic for continuous covering and packing problems." In: *SIAM Journal on Algebraic Discrete Methods* 3.4 (Dec. 1982), pp. 584–591. DOI: 10.1137/0603059.

[Gab+14] Virginie Gabrel et al. "Robust location transportation problems under uncertain demands." In: *Discrete Applied Mathematics* 164 (Feb. 2014), pp. 100–111. DOI: 10.1016/j.dam.2011.09.015.

[Gai+03] Martin Gairing et al. "Self-stabilizing algorithms for $\{k\}$-domination." In: *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2003, pp. 49–60. DOI: 10.1007/3-540-45032-7_4.

[GHY93] Olivier Goldschmidt, Dorit S. Hochbaum, and Gang Yu. "A modified greedy heuristic for the set covering problem with improved worst case bound." In: *Information Processing Letters* 48.6 (Dec. 1993), pp. 305–310. DOI: 10.1016/0020-0190(93)90173-7.

[GJ79] Michael R. Garey and David S. Johnson. *Computers and intractability: A guide to the theory of NP-completeness*. W. H. Freeman and Company, Apr. 1979. ISBN: 9780716710455.

[GLS88] Martin Grötschel, Lászlo Lovász, and Alexander Schrijver. *Geometric algorithms and combinatorial optimization*. Springer Berlin Heidelberg, 1988. DOI: 10.1007/978-3-642-78240-4.

[GLS93] Martin Grötschel, Lászlo Lovász, and Alexander Schrijver. *Geometric algorithms and combinatorial optimization*. Springer Berlin Heidelberg, 1993. DOI: 10.1007/978-3-642-78240-4.

[GMT14] Virginie Gabrel, Cécile Murat, and Aurélie Thiele. "Recent advances in robust optimization: An overview." In: *European Journal of Operational Research* 235.3 (June 2014), pp. 471–483. DOI: 10.1016/j.ejor.2013.09.036.

[GNR14] Anupam Gupta, Viswanath Nagarajan, and R. Ravi. "Thresholded covering algorithms for robust and max–min optimization." In: *Mathematical Programming* 146.1-2 (Aug. 2014), pp. 583–615. DOI: 10.1007/s10107-013-0705-5.

[GS16] Marc Goerigk and Anita Schöbel. "Algorithm engineering in robust optimization." In: *Algorithm Engineering*. Lecture Notes in Computer Science. Springer, Cham, Nov. 2016, pp. 245–279. DOI: 10.1007/978-3-319-49487-6_8.

[HCL04] Ming-Jiu Hwang, Chin-I Chiang, and Yi-Hsin Liu. "Solving a fuzzy set-covering problem." In: *Mathematical and Computer Modelling* 40.7-8 (Oct. 2004), pp. 861–865. DOI: 10.1016/j.mcm.2004.10.015.

[Hem+91] Lane A. Hemachandra et al. "On sets polynomially enumerable by iteration." In: *Theoretical Computer Science* 80.2 (1991), pp. 203–225. DOI: 10.1016/0304-3975(91)90388-i.

[Heu08] Harro Heuser. *Lehrbuch der Analysis*. Vol. 2. Vieweg+Teubner Verlag, 2008. ISBN: 3835102087.

[Heu09] Harro Heuser. *Lehrbuch der Analysis*. Vol. 1. Vieweg+Teubner Verlag, 2009. ISBN: 383480777X.

[HG18] Omar El Housni and Vineet Goyal. *On the optimality of affine policies for budgeted uncertainty sets*. 2018. arXiv: 1807.00163 [math.OC].

[HH86] Nicholas G. Hall and Dorit S. Hochbaum. "A fast approximation algorithm for the multicovering problem." In: *Discrete Applied Mathematics* 15.1 (Sept. 1986), pp. 35–40. DOI: 10.1016/0166-218x(86)90016-8.

[HH92] Nicholas G. Hall and Dorit S. Hochbaum. "The multicovering problem." In: *European Journal of Operational Research* 62.3 (Nov. 1992), pp. 323–339. DOI: 10.1016/0377-2217(92)90122-p.

[HJ12] Roger A. Horn and Charles R. Johnson. *Matrix analysis*. Cambridge University Press, Dec. 2012. DOI: 10.1017/cbo9781139020411.

[HLS09] Pierre Hansen, Martine Labbé, and David Schindl. "Set covering and packing formulations of graph coloring: Algorithms and first polyhedral results." In: *Discrete Optimization* 6.2 (May 2009), pp. 135–147. DOI: 10.1016/j.disopt.2008.10.004.

[Hoc82] Dorit S. Hochbaum. "Approximation algorithms for the set covering and vertex cover problems." In: *SIAM Journal on Computing* 11.3 (Aug. 1982), pp. 555–556. DOI: 10.1137/0211045.

[HP87] Pierre Hanjoul and Dominique Peeters. "A facility location problem with clients' preference orderings." In: *Regional Science and Urban Economics* 17.3 (Aug. 1987), pp. 451–473. DOI: 10.1016/0166-0462(87)90011-1.

[Hua+09] Qiang-Sheng Hua et al. "Exact algorithms for set multicover and multiset multicover problems." In: *Algorithms and Computation*. Springer Berlin Heidelberg, 2009, pp. 34–44. DOI: 10.1007/978-3-642-10631-6_6.

[Hua+10] Qiang-Sheng Hua et al. "Dynamic programming based algorithms for set multicover and multiset multicover problems." In: *Theoretical Computer Science* 411.26-28 (June 2010), pp. 2467–2474. DOI: 10.1016/j.tcs.2010.02.016.

[Joh74] David S. Johnson. "Approximation algorithms for combinatorial problems." In: *Journal of Computer and System Sciences* 9.3 (Dec. 1974), pp. 256–278. DOI: 10.1016/s0022-0000(74)80044-9.

[Kan91] Viggo Kann. "Maximum bounded 3-dimensional matching is MAX SNP-complete." In: *Information Processing Letters* 37.1 (Jan. 1991), pp. 27–35. DOI: 10.1016/0020-0190(91)90246-e.

[Kar72] Richard M. Karp. "Reducibility among combinatorial problems." In: *Complexity of Computer Computations*. Vol. 40. Springer, Boston, MA, Jan. 1972, pp. 85–103. DOI: 10.1007/978-1-4684-2001-2_9.

[Kas08] Adam Kasperski. *Discrete optimization with interval data*. Studies in Fuzziness and Soft Computing. Springer Berlin Heidelberg, 2008. DOI: 10.1007/978-3-540-78484-5.

[Kla79] Gabriel Klambauer. *Problems and propositions in analysis*. New York: Marcel Dekker, 1979. ISBN: 0824768876.

[KN12] Sven O. Krumke and Hartmut Noltemeier. *Graphentheoretische Konzepte und Algorithmen*. Vieweg+Teubner Verlag, 2012. DOI: 10.1007/978-3-8348-2264-2.

[Koc11] Yury Kochetov. "Facility location: Discrete models and local search methods." In: *Combinatorial Optimization*. Vol. 31. NATO Science for Peace and Security Series. IOS Press, Feb. 2011, pp. 97–134. DOI: 10.3233/978-1-60750-718-5-97.

[KP16] Arif Khan and Alex Pothen. "A new 3/2-approximation algorithm for the b-edge cover problem." In: *Proceedings of the 7th SIAM Workshop on Combinatorial Scientific Computing (CSC '16)*. SIAM, Jan. 2016, pp. 52–61. DOI: 10.1137/1.9781611974690.ch6.

[KR08] Subhash Khot and Oded Regev. "Vertex cover might be hard to approximate to within $2 - \varepsilon$." In: *Journal of Computer and System Sciences* 74.3 (May 2008), pp. 335–349. DOI: 10.1016/j.jcss.2007.06.019.

[KSS19] Sven O. Krumke, Eva Schmidt, and Manuel Streicher. "Robust multicovers with budgeted uncertainty." In: *European Journal of Operational Research* 274.3 (May 2019), pp. 845–857. DOI: 10.1016/j.ejor.2018.11.049.

[KY05] Stavros G. Kolliopoulos and Neal E. Young. "Approximation algorithms for covering/packing integer programs." In: *Journal of Computer and System Sciences* 71.4 (Nov. 2005), pp. 495–505. DOI: 10.1016/j.jcss.2005.05.002.

[KY97] Panos Kouvelis and Gang Yu. *Robust discrete optimization and its applications*. Springer US, 1997. DOI: 10.1007/978-1-4757-2620-6.

[KZ16] Adam Kasperski and Paweł Zieliński. "Robust discrete optimization under discrete and interval uncertainty: A survey." In: *Robustness Analysis in Decision Aiding, Optimization, and Analytics*. Vol. 241. International Series in Operations Research & Management Science. Springer, Cham, 2016, pp. 113–143. DOI: 10.1007/978-3-319-33121-8_6.

[Law76] Eugene Lawler. *Combinatorial optimization: Networks and matroids*. Holt, Rinehart and Winston, Nov. 1976. ISBN: 0030848660.

[Leh82] Jeno Lehel. "Covers in hypergraphs." In: *Combinatorica* 2.3 (Sept. 1982), pp. 305–309. DOI: 10.1007/bf02579237.

[Len83] Hendrik W. Lenstra. "Integer programming with a fixed number of variables." In: *Mathematics of Operations Research* 8.4 (Nov. 1983), pp. 538–548. DOI: 10.1287/moor.8.4.538.

[Lov75] Lászlo Lovász. "On the ratio of optimal integral and fractional covers." In: *Discrete Mathematics* 13.4 (1975), pp. 383–390. DOI: 10.1016/0012-365x(75)90058-8.

[Lu16] Brian Yin Lu. "Essays on approximation algorithms for robust linear optimization problems." PhD thesis. Columbia University, 2016. DOI: 10.7916/D8ZG6SGM.

[Lüb10] Marco E. Lübbecke. "Column generation." In: *Wiley Encyclopedia of Operations Research and Management Science*. John Wiley & Sons, 2010. DOI: 10.1002/9780470400531.eorms0158.

[Lut+17] Pascal Lutter et al. "Improved handling of uncertainty and robustness in set covering problems." In: *European Journal of Operational Research* 263.1 (Nov. 2017), pp. 35–49. DOI: 10.1016/j.ejor.2017.04.044.

[LY94] Carsten Lund and Mihalis Yannakakis. "On the hardness of approximating minimization problems." In: *Journal of the ACM* 41.5 (Sept. 1994), pp. 960–981. DOI: 10.1145/185675.306789.

[Mey74] Robert R. Meyer. "On the existence of optimal solutions to integer and mixed-integer programming problems." In: *Mathematical Programming* 7.1 (Dec. 1974), pp. 223–235. DOI: 10.1007/bf01585518.

[Mit02] John E. Mitchell. "Branch-and-cut algorithms for combinatorial optimization problems." In: *Handbook of Applied Optimization*. Oxford University Press, Jan. 2002, pp. 65–77.

[Mos15] Dana Moshkovitz. "The projection games conjecture and the NP-hardness of $\ln n$-approximating set-cover." In: *Theory of Computing* 11.1 (2015), pp. 221–235. DOI: 10.4086/toc.2015.v011a007.

[MS18] Andrea Maier and Eva Schmidt. Personal communication. 2018.

[NO13] Ebrahim Nasrabadi and James B. Orlin. *Robust optimization with incremental recourse*. Dec. 2013. arXiv: 1312.4075 [cs.CC].

[NW88] George L. Nemhauser and Laurence A. Wolsey. *Integer and combinatorial optimization*. John Wiley & Sons, June 1988. DOI: 10.1002/9781118627372.

[PA13] Jordi Pereira and Igor Averbakh. "The robust set covering problem with interval data." In: *Annals of Operations Research* 207.1 (Aug. 2013), pp. 217–235. DOI: 10.1007/s10479-011-0876-5.

[PC10] David Pritchard and Deeparnab Chakrabarty. "Approximability of sparse integer programs." In: *Algorithmica* 61.1 (July 2010), pp. 75–93. DOI: 10.1007/s00453-010-9431-z.

[PR91] Manfred Padberg and Giovanni Rinaldi. "A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems." In: *SIAM Review* 33.1 (Mar. 1991), pp. 60–100. DOI: 10.1137/1033004.

[Pré95] András Prékopa. *Stochastic programming*. Mathematics and Its Applications. Springer Netherlands, 1995. DOI: 10.1007/978-94-017-3087-7.

[PVV12] Alessandra Parisio, Carmen Del Vecchio, and Alfredo Vaccaro. "A robust optimization approach to energy hub management." In: *International Journal of Electrical Power & Energy Systems* 42.1 (Nov. 2012), pp. 98–104. DOI: 10.1016/j.ijepes.2012.03.015.

[PY91] Christos H. Papadimitriou and Mihalis Yannakakis. "Optimization, approximation, and complexity classes." In: *Journal of Computer and System Sciences* 43.3 (Dec. 1991), pp. 425–440. DOI: 10.1016/0022-0000(91)90023-x.

[Roc97] Ralph Tyrell Rockafellar. *Convex analysis*. Princeton University Press, 1997. ISBN: 0691015864.

[Rud76] Walter Rudin. *Principles of mathematical analysis*. McGraw-Hill, 1976. ISBN: 9781259064784.

[Sch02] Alexander Schrijver. *Combinatorial optimization, polyhedra and efficiency*. Springer Berlin Heidelberg, 2002. ISBN: 3540443894.

[Sch98] Alexander Schrijver. *Theory of linear and integer programming*. Wiley Series in Discrete Mathematics & Optimization. John Wiley & Sons, 1998. ISBN: 0471982326.

[SGL10] Anureet Saxena, Vineet Goyal, and Miguel Lejeune. "MIP reformulations of the probabilistic set covering problem." In: *Mathematical Programming* 121.1 (June 2010), pp. 1–31. DOI: 10.1007/s10107-008-0224-y.

[Sny06] Lawrence V. Snyder. "Facility location under uncertainty: A review." In: *IIE Transactions* 38.7 (June 2006), pp. 547–564. DOI: 10.1080/07408170500216480.

[Soy73] Allen L. Soyster. "Technical note — convex programming with set-inclusive constraints and applications to inexact linear programming." In: *Operations Research* 21.5 (Oct. 1973), pp. 1154–1157. DOI: 10.1287/opre.21.5.1154.

[Sri01] Aravind Srinivasan. "New approaches to covering and packing problems." In: *Proceedings of the 12th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '01)*. SIAM, Jan. 2001, pp. 567–576.

[Sri06] Aravind Srinivasan. "An extension of the Lovász local lemma, and its applications to integer programming." In: *SIAM Journal on Computing* 36.3 (Jan. 2006), pp. 609–634. DOI: 10.1137/s0097539703434620.

[Sri95] R. Sridharan. "The capacitated plant location problem." In: *European Journal of Operational Research* 87.2 (Dec. 1995), pp. 203–213. DOI: 10.1016/0377-2217(95)00042-o.

[SS12] Rishi Saket and Maxim Sviridenko. "New and improved bounds for the minimum set cover problem." In: *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*. Springer Berlin Heidelberg, 2012, pp. 288–300. DOI: 10.1007/978-3-642-32512-0_25.

[Str21] Manuel Streicher. "Uncertainty in discrete optimization: Connectivity and covering." PhD thesis. Technische Universität Kaiserslautern, 2021.

[SWW19] David Simchi-Levi, He Wang, and Yehua Wei. "Constraint generation for two-stage robust network flow problems." In: *INFORMS Journal on Optimization* 1.1 (Jan. 2019), pp. 49–70. DOI: 10.1287/ijoo.2018.0003.

[Var75] James M. Varah. "A lower bound for the smallest singular value of a matrix." In: *Linear Algebra and its Applications* 11.1 (1975), pp. 3–5. DOI: 10.1016/0024-3795(75)90112-3.

[Vaz03] Vijay V. Vazirani. *Approximation algorithms*. Springer Berlin Heidelberg, 2003. DOI: 10.1007/978-3-662-04565-7.

[Vem98] Rao R. Vemuganti. "Applications of set covering, set packing and set partitioning models: A survey." In: *Handbook of Combinatorial Optimization*. Springer, Boston, MA, 1998, pp. 573–746. DOI: 10.1007/978-1-4613-0303-9_9.

[VK10] Igor L. Vasilyev and Kseniya B. Klimentova. "The branch and cut method for the facility location problem with client's preferences." In: *Journal of Applied and Industrial Mathematics* 4.3 (July 2010), pp. 441–454. DOI: 10.1134/s1990478910030178.

[VKK09] Igor L. Vasil'ev, Kseniya B. Klimentova, and Yury A. Kochetov. "New lower bounds for the facility location problem with clients' preferences." In: *Computational Mathematics and Mathematical Physics* 49.6 (June 2009), pp. 1010–1020. DOI: 10.1134/s0965542509060098.

[Wil98] David P. Williamson. *Lecture notes on approximation algorithms*. Accessed: 2021-01-20. 1998. URL: http://yildirim.bilkent.edu.tr/handouts/williamson_cornell.pdf.

[WS09] David P. Williamson and David B. Shmoys. *The design of approximation algorithms*. Cambridge University Press, 2009. DOI: 10.1017/cbo9780511921735.

[XB17] Guanglin Xu and Samuel Burer. "A copositive approach for two-stage adjustable robust optimization with uncertain right-hand sides." In: *Computational Optimization and Applications* 70.1 (Dec. 2017), pp. 33–59. DOI: 10.1007/s10589-017-9974-x.

[YGH19] İhsan Yanıkoğlu, Bram L. Gorissen, and Dick den Hertog. "A survey of adjustable robust optimization." In: *European Journal of Operational Research* 277.3 (Sept. 2019), pp. 799–813. DOI: 10.1016/j.ejor.2018.08.031.

Curriculum Vitae

Eva Maria Schmidt

since 10/2017	**Doctoral Studies in Mathematics**, TUK
06/2017	**Master of Science in Mathematics**, TUK
03/2017 – 06/2017	**Research Assistant**, TUK
07/2016 – 09/2016	**Internship**, Robert Bosch GmbH, Homburg (Saar)
04/2015 – 08/2015	**Teaching Assistant**, TUK
04/2015 – 06/2017	**Master Studies in Mathematics**, TUK
03/2015	**Bachelor of Science in Mathematics**, TUK
08/2014 – 12/2014	**Semester Abroad**, Lunds Universitet, Lund, Sweden
04/2014 – 07/2014	**Internship**, Fraunhofer ITWM, Kaiserslautern
10/2013 – 09/2014	**Teaching Assistant**, TUK
10/2012 – 03/2013	**Teaching Assistant**, TUK
10/2011 – 03/2015	**Bachelor Studies in Mathematics**, TUK
03/2011	**Abitur**, Helmholtz-Gymnasium, Zweibrücken

Wissenschaftlicher Werdegang

Eva Maria Schmidt

seit 10.2017	**Promotionsstudium in Mathematik**, TUK
06.2017	**Master of Science in Mathematik**, TUK
03.2017 – 06.2017	**Forschungsassistentin**, TUK
07.2016 – 09.2016	**Praktikum**, Robert Bosch GmbH, Homburg (Saar)
04.2015 – 08.2015	**Lehrassistentin**, TUK
04.2015 – 06.2017	**Masterstudium in Mathematik**, TUK
03.2015	**Bachelor of Science in Mathematik**, TUK
08.2014 – 12.2014	**Auslandssemester**, Lunds Universitet, Lund, Schweden
04.2014 – 07.2014	**Fachpraktikum**, Fraunhofer ITWM, Kaiserslautern
10.2013 – 09.2014	**Lehrassistentin**, TUK
10.2012 – 03.2013	**Lehrassistentin**, TUK
10.2011 – 03.2015	**Bachelorstudium in Mathematik**, TUK
03.2011	**Abitur**, Helmholtz-Gymnasium, Zweibrücken