LEARNING ON GRAPHS WITH HIGH-ORDER RELATIONS: SPECTRAL METHODS, OPTIMIZATION AND APPLICATIONS

BY

PAN LI

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2019

Urbana, Illinois

Doctoral Committee:

Professor Olgica Milenkovic, Chair
Professor Bruce Hajek
Professor Jiawei Han
Associate Professor David Gleich, Purdue University
Assistant Professor Niao He

ABSTRACT

Learning on graphs is an important problem in machine learning, computer vision and data mining. Traditional algorithms for learning on graphs primarily take into account only low-order connectivity patterns described at the level of individual vertices and edges. However, in many applications, high-order relations among vertices are necessary to properly model a real-life problem. In contrast to the low-order cases, in-depth algorithmic and analytic studies supporting high-order relations among vertices are still lacking. To address this problem, we introduce a new mathematical model family, termed *inhomogeneous hypergraphs*, which captures the high-order relations among vertices in a very extensive and flexible way. Specifically, as opposed to classic hypergraphs that treat vertices within a high-order structure in a uniform manner, inhomogeneous hypergraphs allow one to model the fact that different subsets of vertices within a high-order relation may have different structural importance. We propose a series of algorithms and relevant analytic results for this new model.

First, after we introduce the formal definitions and some preliminaries, we propose clustering algorithms over inhomogeneous hypergraphs. The first clustering method is based on a projection method, where we use graphs with pairwise relations to approximate high-order relations and then directly use spectral clustering methods over obtained graphs. For this type of method, we provide provable performance guarantee, which works for a sub-class of inhomogeneous hypergraphs that additionally impose constraints on the internal structures of high-order relations. Such constraints are related to submodular functions, so we term such a sub-class of inhomogeneous hypergraphs as submodular hypergraphs. Later, we study the Laplacian operators for these hypergraphs and generalize many important results in spectral theory for this setting including Cheeger's inequalities and discrete nodal domain theorems. Based on these new results, we further develop new clustering al-

gorithms with tighter approximating properties than projection methods.

Second, we propose some optimization algorithms for inhomogeneous hypergraphs. We first find that min-cut problems over submodular hypergraphs are closely related to an extensively studied optimization problem termed decomposable submodular hypergraph minimization (DSFM). Our contribution is how to leverage hypergraph structures to accelerate canonical solvers for DSFM problems. Later, we connect PageRank approaches to submodular hypergraphs and propose a new optimization problem termed quadratic decomposable submodular hypergraph minimization (QDSFM). For this new problem, we propose algorithms with first provable linear convergence guarantee and identify new relevant applications.

To my parents and grandparents, for their love and support.

ACKNOWLEDGMENTS

I would like to thank many people for their support over the past four years.

First, I owe my deepest gratitude to my advisor Professor Olgica Milenkovic for her continuous support. She guided me to find and dive into many interesting research problems. Without her guidance, I would not have completed this thesis. She helped me a lot with improving my writing and presentation skills, which I will treasure throughout my academic life. She also worked hard on applying for grants and kept generously providing financial support of all my research projects.

Next, I would like to particularly thank my doctoral committee member Professor David Gleich. As an expert in high-order data processing, he generously offered insightful advice on my works. More than technique-level guidance, David also kindly assisted me in deciding my career path. Without his constant support, I would have given up the academic career. I am more than thrilled to become his colleague at Purdue University in the future.

I also would like to acknowledge other members of my doctoral committee including Professors Jiawei Han, Niao He and Bruce Hajek for their valuable feedback and discussions on my works. Special thanks to Niao: We have closely worked together. She has always been ready for discussions and willing to tackle problems together with me.

I also would like to thank many friends at UIUC, Weihao Gao, Zhenzhe Zheng, Yuheng Bu, Kaiqing Zhang, Shiyu Liang, Ruochen Lu, Xiaolong Zhang, Bangqi Wang, Siyu Lai, Yanyun Wang and many others, for their companionship and encouragement. Also, many thanks to my colleagues in our lab, Eli Chien, Jianhao Peng, Minji Kim, Sri Pattabiraman and others, for their help in many ways. I feel particularly fortunate to work with Eli on some research projects. He is a very brilliant young researcher.

Finally, my deepest gratitude to my family! I thank my parents and grandparents for their endless love and constant support in every aspect of my life.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

A graph is a mathematical tool for systematical modeling of relations between objects. It has been used in a wide range of real-life settings, such as social networks [1], biological networks [2], communication/computer networks [3], or modeling topological adjacency relations between data points [4].

Traditional graphs only model pairwise dependencies between objects. However, in many real-world problems, it is necessary to capture joint, higher-order relations between subsets of objects. From a mathematical point of view, these higher-order relations may be described via hypergraphs, where objects correspond to vertices and higher-order relations among objects correspond to hyperedges [5]. Recent work on hypergraph analysis has focused on a variety of problems, including finding min-cuts [6], clustering [7], random walk studies [8] and others [9].

In the traditional definition of hypergraphs, each hyperedge is typically equipped with a scalar weight to represent the strength of the high-order connection. We term such hypergraphs as homogeneous hypergraphs. The definition is simple but it limits the power of representation for hypergraphs to model high-order relations. Clearly, a scalar weight itself prohibits the use of information regarding how different vertices or subsets of vertices belonging to a hyperedge contribute to the higher-order relation. A more appropriate formulation entails assigning different weights to different subsets of the hyperedges, thereby endowing hyperedges with vector weights capturing the high-order relation. To illustrate the point, consider the example of metabolic networks [10]. In these networks, vertices describe metabolites while edges describe transformative, catalytic or binding relations. Metabolic reactions are usually described via equations that involve more than two metabolites, such as $M_1 + M_2 \rightarrow M_3$. Here, both metabolites M_1 and M_2 need to be present in order to complete the reaction that leads to the creation of the product M_3. The three metabolites play different roles: M_1, M_2 are reac-

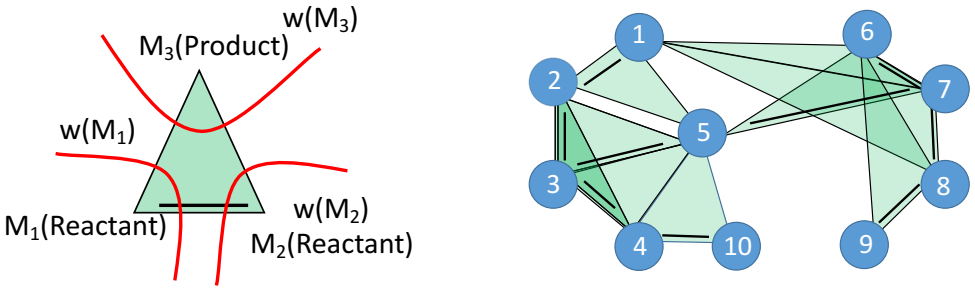


Figure 1.1: An example of inhomogeneous hypergraph. Left: Each reaction is represented by a hyperedge. The hyperedge is associated with three weights $w(M_1)$, $w(M_2)$, $w(M_3)$ that indicate how much each vertex contributes to this high-order relation. Right: a metabolic network.

tants, while M_3 is the product metabolite. This high-order relation can be modeled as a hyperedge. A synthetic metabolic network involving reactions with three reagents as described above is depicted in Figure 1.1. A proper way to represent this high-order relation is to allow to assign three weights $w(M_1)$, $w(M_2)$ and $w(M_3)$ that describe the contribution of each metabolite to the whole relation. According to their roles, we might set $w(M_1) = w(M_2)$ while allowing $w(M_3)$ to be different from the other two, which captures the semantic meaning underlying this hypergraph. Generally, for each hyperedge e and a subset $S \subset e$, we let $w_e(S)$ denote the weight of the subset S with respect to the hyperedge e (formal definitions are postponed to Chapter 2). We term such a type of hypergraph as an inhomogeneous hypergraph. The scope of this thesis is to solve a collection of problems associated with inhomogeneous hypergraphs.

First, we examine the inhomogeneous hypergraph clustering (partitioning) problem. In Chapter 3, we propose an efficient algorithm for inhomogeneous hypergraph partitioning, which is based on the clique-expansion method [11]. The algorithmic method first transforms hypergraphs into graphs and subsequently performs spectral clustering based on the normalized Laplacian of the derived graph. As opposed to the uniform clique expansion method for homogeneous hypergraph clustering, the inhomogeneous clustering algorithm allows for non-uniform expansion of one hyperedge. We also analyze the conditions under which clique-expansion-based algorithm have theoretical performance guarantees. We prove that when the weights associated with the inhomogeneous hyperedges are *symmetric* and *submodular*, the inhomoge-

neous clustering procedure has the same quadratic approximation properties as spectral graph clustering [12] (with the presence of an additional constant factor that is determined by the size of the hyperedges). We term this subclass of inhomogeneous hypergraphs as submodular hypergraphs. At the end of the chapter, we also provide multiple applications including hierarchical biological network decompositions, structure learning of rankings and subspace clustering, all to demonstrate the effectiveness of inhomogeneous hypergraph models and the proposed clique-expansion-based clustering algorithm.

Although clique expansion works well when the size of the hyperedges is a small constant, it may introduce a large distortion when expanding large hyperedges. This motivates the work in the Chapter 4. We avoid resorting to the spectral clustering of standard graphs and try to directly perform spectral clustering based on the Laplacians of inhomogeneous hypergraphs. We focus on the subclass of submodular hypergraphs and define p-Laplacians of submodular hypergraphs and analyze their spectra. In particular, we generalize two fundamental results in spectral graph theory to submodular hypergraphs: discrete nodal domain theorems and Cheeger inequalities. We also provide algorithms to learn the spectra of these Laplacian operators, which can be shown to provide better clustering results than clique expansion.

Inhomogeneous hypergraphs, especially submodular hypergraphs, also have arisen in many applications other than data clustering. Actually, many semi-supervised learning problems [13, 14] where practitioners know a few labels while intending to infer other missing labels naturally appear in the settings of inhomogeneous hypergraphs. For example, in images, pixels can be viewed as vertices and pairs of adjacent pixels in the same columns and rows naturally correspond to edges, which overall model images as graphs. If the relations between pixels are more than one-hop relations, e.g., a super pixel — a local region of the image that covers multiple pixels — then images correpond to hypergraphs instead of graphs. Moreover, such high-order relations may be associated with set functions to characterize partitioning costs, which typically satisfies submodularity, and then the whole image naturally becomes a submodular hypergraph. Many recent works on image segmentation leveraged these modeling strategies and used the min-cut solutions between user-specified pixels of an object and of the background as the boundary for segementation [15, 16, 17]. Because of this important application, it is always an important problem to consider how to efficiently solve the

min-cut problem over these submodular hypergraphs. Mathematically, such a min-cut problem is essentially a submodular function minimization problem, while some additional structures may be leveraged here to accelerate the generic approachers for submodular function minimization: Basically, the whole submodular function is naturally decomposed into the submodular functions defined over hyperedges, and thus this type of submodular functions can be termed termed as *decomposable submodular functions* in relevant literature [18]. The overall minimization problem is termed decomposable submodular function minimization (DSFM) problems. Chapter 5 in this thesis proposes a way to properly leverage such decomposable structures to accelerate DSFM. In Chapter 5, we analyze in-depth how to leverage the incident relations between vertices and hyperedges to improve the previously proposed algorithms for DSFM including alternative projection (AP) [19] and random coordinate descent (RCD) [20]. We find AP can be significantly accelerated by leveraging incident relations while RCD cannot, for which we show a negative example. However, incident relations may be used to accelerate RCD if multiple cooredinates are allowed to descend in parallel. Moreover, we show that properly setting the weights of different vertices based on their incident relations when we formulate DSFM problems in continuous domain is also helpful to decrease the complexity of AP and RCD.

The min-cut type of solutions are effective in tasks like image segmentation where the output consists of discrete labels that correspond to either an object or the background, while many other applications want to have soft scores rather than discrete labels, e.g., ranking problems in information retrieval and recommender systems: When using web search, one will not expect the response of the search engine to only tell binary information about which documents are relevant to the query and which are not. One may expect to have a ranking of these documents based on some relevant scores. For these applications, we need to use another graph-based algorithm termed PageRank that builds the initial algorithmic foundation for the world-wide search engine — Google Search [21]. Actually, the PageRank vector can be obtained via solving an optimization problem [22]. In the graph case, this optimization problem has a form similar to that of the optimization problem to compute the min-cut solution shown in Chapter 5, but the only difference appears in the powers of the regularizer defined for smoothing values according to graph topology. Specifically, the power order of the regularizer

to have PageRanks as solutions is two while that to have min-cut solutions is one. Based on this obsevation, we may revise the power of the regularizer appearing in DSFM to two and obtain a new optimization problem termed quadratic decomposable submodular function minimization (QDSFM) problems. QDSFM can be used to compute PageRanks over submodular hyergraphs that are determined by the 2-Laplacian operators defined in Chapter 4. Actually, relevant formulation for standard hypergraphs has been considered in some previous works [9, 23] and our new formulation generalizes it. Chapter 6 gives the formal formulation of QDSFM and discusses how to solve it. Some idea of DSFM solvers can be leveraged but many new techniques have to be developed. We may use AP and RCD to solve QDSFM while we need new techniques to prove their linearly convergent properties. Moreover, each iteration of both AP and RCD requires computing projection to cones generated from the base polytopes of submodular functions. To compute such projection for general submodular functions, we generalize the Frank-Wolfe algorithm [24] and the min-norm-point algorithm [25] for the conic cases. We also prove that the obtained PageRank over hypergraphs can be used to find partitions of hypergraphs that approximate hypergraph conductance.

CHAPTER 2

PRELIMINARIES AND NOTATIONS

2.1 Graph partitioning and the Cheeger inequality

A weighted graph $G = (V, E, \boldsymbol{\mu}, \mathbf{w})$ is an ordered pair of two sets, the vertex set $V = [N] = \{1, 2, \ldots, N\}$, equipped with a positive weight function $\boldsymbol{\mu} = \{\mu_u\}_{u \in V}$, and the edge set $E \subseteq V \times V$, equipped with a positive weight function $\mathbf{w} = \{w_e\}_{e \in E}$. Given a subset of vertices $S \subseteq V$, the volume of S is defined as

$$\mathrm{vol}(S) = \sum_{u \in S} \mu_u. \tag{2.1}$$

For graph, the degree of a vertex is defined as $d_u = \sum_{v:(u,v)\in E} w_{uv}$ for $u \in V$. A common choice of the vertex weight is to set $\mu_u = d_u$. However, here, μ_u is allowed for a different value for more general results. Let U and D be the diagonal matrices such that $U_{vv} = \mu_v$ and $D_{vv} = d_v$ for $v \in V$ respectively. Also, define $\tau = \max_{u \in V} \frac{d_u}{\mu_u}$.

A cut $C = (S, \bar{S})$ is a bipartition of the set V, while the cut-set (boundary) of the cut C is defined as the set of edges that have one endpoint in S and one in the complement of S, $\bar{S}$, i.e., $\partial S = \{(u, v) \in E \mid u \in S, v \in \bar{S}\}$. The volume of the cut induced by S equals $\mathrm{vol}(\partial S) = \sum_{u \in S,\, v \in \bar{S}} w_{uv}$. Based on this definition, the conductance of the cut is defined as

$$c(S) = \frac{\mathrm{vol}(\partial S)}{\min\{\mathrm{vol}(S), \mathrm{vol}(\bar{S})\}}.$$

The smallest conductance of any bipartition of a graph G is denoted by h_2 and referred to as the Cheeger constant of the graph. A generalization of the Cheeger constant is the *$k-$way Cheeger constant* of a graph G. Let P_k denote the set of all partitions of V into k-disjoint nonempty subsets, i.e.,

$P_k = \{(S_1, S_2, ..., S_k) | S_i \subset V, S_i \neq \emptyset, S_i \cap S_j = \emptyset, \forall i, j \in [k], i \neq j\}$. The $k-$way Cheeger constant is defined as

$$h_k = \min_{(S_1,S_2,...,S_k)\in P_k} \max_{i\in[k]} c(S_i).$$

It is known to be NP-complete to find the partition that achieves the Cheeger constant for graphs [26]. However, an efficient algorithm based on spectral techniques (Algorithm 2.1 below) can produce a solution $\hat{S}$ such that

$$c(\hat{S}) \leq \sqrt{2\tau\lambda}, \tag{2.2}$$

where λ is the second smallest eigenvalue of the U-normalized graph Laplacian $\mathcal{L}$. The well known Cheeger inequality further asserts the following relationship between h_2 and λ:

$$h_2 \leq \sqrt{2\tau\lambda} \leq 2\sqrt{\tau h_2}. \tag{2.3}$$

Therefore, spectral clustering for graphs provides a quadratically optimal graph partition. Note that both (2.2) and (2.3) can be obtained by accomodating the proof of [12] with the vertex weights $\boldsymbol{\mu}$.

Algorithm 2.1: Spectral graph partitions for graphs

Input: $G = (V, E, \boldsymbol{\mu}, \mathbf{w})$

1: Construct the adjacency matrix A: $A_{uv} = w_{uv}$ if $(u, v) \in E$ or 0 otherwise.

2: Construct the diagonal degree matrix D.

3: Construct the diagonal vertex weight matrix U.

4: Construct the U-normalized Laplacian matrix $\mathcal{L} = U^{-1/2}(D - A)U^{-1/2}$.

5: Compute the eigenvector $\mathbf{x} = (x_1, x_2, ..., x_n)^T$ corresponding to the second smallest eigenvalue of $\mathcal{L}$.

6: Let u_i be the index of the i-th smallest entry of $U^{-1/2}\mathbf{x}$.

7: Compute $S = \arg\min_{S_i, 1\leq i\leq N-1} c(S_i)$ over all sets $S_i = \{u_1, u_2, \ldots, u_i\}$.

Output: Output S if $\mathrm{vol}(S) < \mathrm{vol}(\bar{S})$, and $\bar{S}$ otherwise.

2.2 Definition of inhomogeneous hypergraphs and its submodular subclass

A weighted hypergraph $G = (V, E, \boldsymbol{\mu}, \mathbf{w})$ is an ordered pair of two sets, the vertex set $V = [N]$ and the hyperedge set $E \subseteq 2^V$, both equipped with a weight function $\boldsymbol{\mu} : V \to \mathbb{R}^+$ and $\mathbf{w} : E \to \mathbb{R}^+$ respectively. The relevant notions of cuts, boundaries and volumes for hypergraphs can be defined in a similar manner as for graphs. If each cut of a hyperedge e has the same weight w_e, we refer to the cut as a *homogeneous cut* and the corresponding hypergraph as a *homogeneous hypergraph*.

A weighted hypergraph $G = (V, E, \boldsymbol{\mu}, \mathbf{w})$ is termed a *inhomogeneous hypergraph* with vertex set V, hyperedge set E and positive vertex weight vector $\boldsymbol{\mu} \triangleq \{\mu_v\}_{v\in V}$, if each hyperedge $e \in E$ is associated with a weight function $w_e(\cdot) : 2^e \to [0, 1]$ that satisfies

- Normalized, so that $w_e(\emptyset) = 0$, and all cut weights corresponding to a hyperedge e are normalized by $\vartheta_e = \max_{S\subseteq e} w_e(S)$. In this case, $w_e(\cdot) \in [0, 1]$;
- Symmetric, so that $w_e(S) = w_e(e\backslash S)$ for any $S \subseteq e$;

The inhomogeneous hyperedge weight functions are summarized in the vector $\mathbf{w} \triangleq \{(w_e, \vartheta_e)\}_{e\in E}$. Let $\zeta(E) \triangleq \max_e |e|$.

For a ground set Ω, a set function $f : 2^\Omega \to \mathbb{R}$ is termed submodular if for all $S, T \subseteq \Omega$, one has $f(S) + f(T) \geq f(S \cup T) + f(S \cap T)$. In addition, if the weight function $w_e(\cdot)$ also satisfy submodularity, we term this subclass of inhomogeneous hypergraphs as *submodular hypergraphs*.

If $w_e(S) = 1$ for all $S \in 2^e\backslash\{\emptyset, e\}$, inhomogeneous hypergraphs reduce to homogeneous hypergraphs. We omit the designation homogeneous whenever there is no context ambiguity.

We define a vertex v is *incident* to a hyperedge e if for some $S \subseteq e\backslash\{v\}$, $w_e(S \cup \{v\}) \neq w_e(S)$. Clearly, for submodular hypergraphs, a vertex v is in e if and only if $w_e(\{v\}) > 0$: as $|w_e(S \cup \{v\}) - w_e(S)| \leq w_e(\{v\})$ if $w_e(\cdot)$ is symmetric and submodular. Moreover, we define the degree of a vertex v as $d_v = \sum_{e\in E:\, v\in e} \vartheta_e$, i.e., as the sum of the max weights of edges incident to the vertex v. The volume of a subset of vertices $S \subseteq V$ equals $\mathrm{vol}(S) = \sum_{v\in S} \mu_v$.

For any $S \subseteq V$, we generalize the notions of the boundary of S and the volume of the boundary of S according to $\partial S = \{e \in E | e \cap S \neq \emptyset, e \cap \bar{S} \neq \emptyset\}$,

and

$$\mathrm{vol}(\partial S) = \sum_{e \in \partial S} \vartheta_e w_e(S) = \sum_{e \in E} \vartheta_e w_e(S), \tag{2.4}$$

respectively. Then, the conductance of the cut induced by S, the Cheeger constant and the k-way Cheeger constant for hypergraphs are defined in an analogous manner as for graphs.

2.3 Relevant background on submodular functions

Given an arbitrary set function $F : 2^V \to \mathbb{R}$, the *Lovász extension* [27] $f : \mathbb{R}^N \to \mathbb{R}$ of F is defined as follows: For any vector $x \in \mathbb{R}^N$, we order its entries in nonincreasing order $x_{i_1} \geq x_{i_2} \geq \cdots \geq x_{i_n}$ while breaking the ties arbitrarily, and set

$$f(x) = \sum_{j=1}^{N-1} F(S_j)(x_{i_j} - x_{i_{j+1}}) + F(V), \tag{2.5}$$

with $S_j = \{i_1, i_2, ..., i_j\}$. For submodular F, the Lovász extension is a convex function [27].

Let $\mathbf{1}_S \in \mathbb{R}^N$ be the indicator vector of the set S. Hence, for any $S \subseteq V$, one has $F(S) = f(\mathbf{1}_S)$. For a submodular F, we define a convex set termed the *base polytope*

$$\mathcal{B} \triangleq \{y \in \mathbb{R}^N | y(S) \leq F(S), \text{ for all } S \subseteq V, \text{and such that } y(V) = F(V) = 0\}.$$

According to the defining property of submodular functions [27], we may write $f(x) = \max_{y \in \mathcal{B}} \langle y, x \rangle$.

The subdifferential $\nabla f(x)$ of f is defined as

$$\{y \in \mathbb{R}^N \,|\, f(x') - f(x) \geq \langle y, x' - x \rangle, \, \forall x' \in \mathbb{R}^N\}.$$

An important result from [28] characterizes the subdifferentials $\nabla f(x)$: If $f(x)$ is the Lovász extension of a submodular function F with base polytope

$\mathcal{B}$, then

$$\nabla f(x) = \arg\max_{y \in \mathcal{B}} \langle y, x \rangle. \tag{2.6}$$

Observe that $\nabla f(x)$ is a set and that the right-hand side of the definition represents a set of maximizers of the objective function. If $f(x)$ is the Lovász extension of a submodular function, then $\langle q, x \rangle = f(x)$ for all $q \in \nabla f(x)$.

For each hyperedge $e \in E$ of a submodular hypergraph, following the above notations, we let $\mathcal{B}_e$, $\mathcal{E}(\mathcal{B}_e)$, f_e denote the base polytope, the set of extreme points of the base polytope, and the Lovász extension of the submodular hyperedge weight function w_e, respectively. Note that for any $S \subseteq V$, $w_e(S) = w_e(S \cap e)$. Consequently, for any $y \in \mathcal{B}_e$, $y_v = 0$ for $v \notin e$. Since $\nabla f_e \subseteq \mathcal{B}_e$, it also holds that $(\nabla f_e)_v = 0$ for $v \notin e$. When using formula (2.5) to explicitly describe the Lovász extension f_e, we can either use a vector x of dimension N or only those of its components that lie in e. Furthermore, in the later case, $|\mathcal{E}(\mathcal{B}_e)| = |e|!$.

CHAPTER 3

CLIQUE-EXPANSION ALGORITHMS FOR INHOMOGENEOUS HYPERGRAPH CLUSTERING

Graph partitioning or clustering is a ubiquitous learning task that has found many applications in statistics, data mining, social science and signal processing [29, 30]. In most settings, clustering is formally cast as an optimization problem that involves entities with different pairwise similarities and aims to maximize the total "similarity" of elements within clusters [31, 32, 33], or simultaneously maximize the total similarity within clusters and dissimilarity between clusters [34, 35, 36]. Graph partitioning may be performed in an agnostic setting, where part of the optimization problem is to automatically learn the number of clusters [34, 35].

In this chapter and the next, our partitioning/clustering problems choose to use the 2-way partition $(S, \bar{S})$ that may approximate the Cheeger constant h_2 as the objective. Concretely, we essentially try to solve the following optimization problem:

$$\min_{S \subset V} \quad c(S). \tag{3.1}$$

In this chapter, we introduce the first spectral clustering approach for inhomogeneous hypergraphs which is inspired by the clique expansion method for homogeneous hypergraphs [11, 7]. We show that the clique expansion method does not offer good performance guarantees for general inhomogeneous hypergraph, but provably works for submodular hypergraphs with small hyperedge sizes. We also introduce several applications to evaluate our algorithms, including learning ranking models, hierarchical network clustering and subspace segmentation. In all these cases, inhomogeneous hypergraphs prove to be outstanding modeling and data mining tools.

3.1 The clique expansion algorithm and related theoretical analysis

We first introduce the clique expansion methods. The approach first transforms hypergraphs into graphs and then leverages the classic spectral clustering (Algorithm 3.1). Specifically, it includes three steps: 1) Projecting each hyperedge onto a subgraph; 2) Merging the subgraphs into a graph; 3) Performing spectral clustering (Algorithm 3.1) based on the obtained graph. In contrast to the homogeneous case, the novelty of the clique expansion approach of inhomogeneous hypergraphs is in introducing the additional constraints in the projection step according to the projection, and stating an optimization problem that provides the provably best weight splitting for projections. For each inhomogeneous hyperedge (e, w_e), we aim to find a complete subgraph $G_e = (V^{(e)}, E^{(e)}, w^{(e)})$ that "best" represents this hyperedge; here, $V^{(e)} = e$, $E^{(e)} = \{\{v, \tilde{v}\} | v, \tilde{v} \in e, v \neq \tilde{v}\}$, and $w^{(e)} : E^{(e)} \to R$ denotes the hyperedge weight vector. The goal is to find the graph edge weights that provide the best approximation to the split hyperedge weight according to:

$$\min_{w^{(e)}, \beta^{(e)}} \beta^{(e)} \text{ s.t. } w_e(S) \leq \sum_{v \in S, \tilde{v} \in e/S} w_{v\tilde{v}}^{(e)} \leq \beta^{(e)} w_e(S), \quad \text{for all } S \in 2^e, \, w_e(S) \text{ is defined.} \tag{3.2}$$

Upon solving for the weights $w^{(e)}$, we construct a graph $\mathcal{G} = (V, E_o, w)$, where V are the vertices of the hypergraph, E_o is the complete set of edges, and where the weights $w_{v\tilde{v}}$, are computed via

$$w_{v\tilde{v}} \triangleq \sum_{e \in E} \vartheta_e w_{v\tilde{v}}^{(e)}, \quad \forall \{v, \tilde{v}\} \in E_o. \tag{3.3}$$

This step represents the projection weight merging procedure, which simply reduces to the sum of weights of all hyperedge projections on a pair of vertices. Due to the linearity of (2.1) and (2.4) of sets S of vertices, for any $S \subset V$, we have

$$\text{Vol}(\partial S) \leq \text{Vol}_{\mathcal{G}}(\partial S) \leq \beta^* \text{Vol}(\partial S), \tag{3.4}$$

where $\mathrm{Vol}_{\mathcal{G}}(\cdot)$ denotes the volumes of corresponding sets over the obtained graph $\mathcal{G}$ and $\beta^* = \max_{e \in E} \beta^{(e)}$. Applying spectral clustering on $\mathcal{G} = (V, E_o, w)$ produces the desired partition $(S^*, \bar{S^*})$. The next result is a consequence of combining the bounds of (3.4) with the approximation guarantees of spectral graph clustering (2.2) and (2.3).

Theorem 3.1.1. *If the optimization problem* (3.2) *is feasible for all hyperedges and the weights* $w_{v\tilde{v}}$ *obtained from* (3.3) *are nonnegative for all* $\{v, \tilde{v}\} \in E_o$*, then*

$$c(S^*) \leq 2\beta^* \sqrt{\tau h_2} \tag{3.5}$$

where $\beta^* = \max_{e \in E} \beta^{(e)}$.

There are no guarantees that the $w_{v\tilde{v}}$ will be nonnegative: The optimization problem (3.2) may result in solutions $w^{(e)}$ that are negative. The performance of spectral methods in the presence of negative edge weights is not well understood [37, 38]; hence, it would be desirable to have the weights $w_{v\tilde{v}}$ generated from (3.3) be nonnegative. Unfortunately, imposing nonngativity constraints in the optimization problem may render it infeasible. In practice, one may use $(w_{v\tilde{v}})_+ = \max\{w_{v\tilde{v}}, 0\}$ to remove negative weights (other choices, such as $(w_{v\tilde{v}})_+ = \sum_e (w_{v\tilde{v}}^{(e)})_+$ do not appear to perform well). This change invalidates the theoretical result of Theorem 3.1.1, but provides solutions with very good empirical performance. The issues discussed are illustrated by the next example.

Example 3.1.1. *Let* $e = \{1, 2, 3\}$, $(w_e(\{1\}), w_e(\{2\}), w_e(\{3\})) = (0, 0, 1)$. *The solution to the weight optimization problem is* $(\beta^{(e)}, w_{12}^{(e)}, w_{13}^{(e)}, w_{23}^{(e)}) = (1, -1/2, 1/2, 1/2)$. *If all components* $w^{(e)}$ *are constrained to be nonnegative, the optimization problem is infeasible. Nevertheless, the above choice of weights is very unlikely to be encountered in practice, as* $w_e(\{1\}), w_e(\{2\}) = 0$ *indicates that vertices* 1 *and* 2 *have no relevant connections within the given hyperedge* e*, while* $w_e(\{3\}) = 1$ *indicates that vertex* 3 *is strongly connected to* 1 *and* 2*, which is a contradiction. Let us assume next that the negative weight is set to zero. Then, we adjust the weights* $((w_{12}^{(e)})_+, w_{13}^{(e)}, w_{23}^{(e)}) = (0, 1/2, 1/2)$, *which produce clusterings ((1,3)(2)) or ((2,3)(1)); both have zero costs based on* w_e.

Another problem is that arbitrary choices for w_e may cause the optimization problem to be infeasible (3.2) even if negative weights of $w^{(e)}$ are allowed, as illustrated by the following example.

Example 3.1.2. *Let $e = \{1, 2, 3, 4\}$, with $w_e(\{1, 4\}) = w_e(\{2, 3\}) = 1$ and $w_e(S) = 0$ for all other choices of sets S. To force the weights to zero, we require $w^{(e)}_{v\tilde{v}} = 0$ for all pairs $v\tilde{v}$, which fails to work for $w_e(\{1, 4\}), w_e(\{2, 3\})$. For a hyperedge e, the degrees of freedom for w_e are $2^{|e|-1} - 1$, as two values of w_e are fixed, while the other values are paired up by symmetry. When $|e| > 3$, we have $\binom{|e|}{2} < 2^{|e|-1} - 1$, which indicates that the problem is overdetermined/infeasible.*

In what follows, we provide sufficient conditions for the optimization problem to have a feasible solution with nonnegative values of the weights $w^{(e)}$. Also, we provide conditions for the weights w_e that result in a small constant β^* and hence allow for quadratic approximations of the optimum solution. Our results depend on the availability of information about the weights w_e: In practice, the weights have to be inferred from observable data, which may not suffice to determine more than the weight of singletons or pairs of elements.

Only the values of $w_e(\{v\})$ are known. In this setting, we are only given information about how much each node contributes to a higher-order relation, i.e., we are only given the values of $w_e(\{v\})$, $v \in V$. Hence, we have $|e|$ costs (equations) and $|e| \geq 3$ variables, which makes the problem underdetermined and easy to solve. The optimal $\beta^e = 1$ is attained by setting for all edges $\{v, \tilde{v}\}$

$$w^{(e)}_{v\tilde{v}} = \frac{1}{|e| - 2} \left[w_e(\{v\}) + w_e(\{\tilde{v}\}) \right] - \frac{1}{(|e| - 1)(|e| - 2)} \sum_{v' \in e} w_e(\{v'\}). \quad (3.6)$$

The components of $w_e(\cdot)$ with positive coefficients in (3.1) are precisely those associated with the endpoints of edges $v\tilde{v}$. Using simple algebraic manipulations, one can derive the conditions under which the values $w^{(e)}_{v\tilde{v}}$ are nonnegative.

The solution to (3.6) produces a perfect projection with $\beta^{(e)} = 1$. Unfortunately, one cannot guarantee that the solution is nonnegative. Hence, the question of interest is to determine for what types of cuts one can deviate from a perfect projection but ensure that the weights are nonnegative. The

proposed approach is to set the unspecified values of $w_e(\cdot)$ so that the inhomogeneous hypergraph becomes a submodular hypergraph, which guarantees nonnegative weights $w_{v\tilde{v}}^{(e)}$ that can constantly approximate $w_e(\cdot)$, although with a larger approximation constant β. In submodular hypergraphs, the constraint on hyperedge weights w_e performs as a sufficient condition for the optimization problem to have a feasible solution with nonnegative values of the weights $w^{(e)}$, which we will prove later on. Also, we provide conditions for the weights w_e that result in a small constant β^* and hence allow for quadratic approximations of the optimum solution.

Theorem 3.1.2. *If w_e is normalized, submodular and symmetric, then*

$$w_{v\tilde{v}}^{*(e)} = \sum_{S\in 2^e/\{\emptyset,e\}} \Bigg[\frac{w_e(S)}{2|S|(|e|-|S|)}1_{|\{v,\tilde{v}\}\cap S|=1} \tag{3.7}$$
$$- \frac{w_e(S)}{2(|S|+1)(|e|-|S|-1)}1_{|\{v,\tilde{v}\}\cap S|=0} - \frac{w_e(S)}{2(|S|-1)(|e|-|S|+1)}1_{|\{v,\tilde{v}\}\cap S|=2}\Bigg]$$

is nonnegative. For $2 \leq |e| \leq 7$, the function above is a feasible solution for the optimization problem (3.2) *with parameters $\beta^{(e)}$ listed in Table 3.1.*

Table 3.1: Feasible values of $\beta^{(e)}$ for $\delta^{(e)}$.

$\lvert e\rvert$	2	3	4	5	6	7
β	1	1	3/2	2	4	6

Theorem 3.1.2 also holds when some weights in the set w_e are not specified, but may be completed to satisfy submodularity constraints (see Example 3.1.3).

Example 3.1.3. *Let $e = \{1,2,3,4\}$, $(w_e(\{1\}), w_e(\{2\}), w_e(\{3\}), w_e(\{4\})) = (1/3, 1/3, 1, 1)$. Solving* (3.6) *yields $w_{12}^{(e)} = -1/9$ and $\beta^{(e)} = 1$. Completing the missing components in w_e as $(w_e(\{1,2\}), w_e(\{1,3\}), w_e(\{1,4\})) = (2/3, 1, 1)$ leads to submodular weights (observe that completions are not necessarily unique). Then, the solution of* (3.7) *gives $w_{12}^{(e)} = 0$ and $\beta^{(e)} \in (1, 2/3]$, which is clearly larger than one.*

Remark 3.1.1. *It is worth pointing out that $\beta = 1$ when $|e| = 3$, which asserts that homogeneous triangle clustering may be performed via spectral methods on graphs without any weight projection distortion [39]. The above*

results extend this finding to a much more general case, i.e., submodular hypergraphs. In addition, triangle clustering based on random walks [40] may be extended to submodular hypergraphs similarly.

Also, (3.7) leads to an optimal approximation ratio $\beta^{(e)}$ if we restrict $w^{(e)}$ to be a linear mapping of w_e, which is formally stated next.

Theorem 3.1.3. *Suppose that for all pairs of $\{v, \tilde{v}\} \in E_o$, $w^{(e)}_{v\tilde{v}}$ is a linear function of w_e, denoted by $w^{(e)}_{v\tilde{v}} = \mathcal{T}_{v\tilde{v}}(w_e)$, where $\{\mathcal{T}_{v\tilde{v}}\}_{\{v\tilde{v}\in E^{(e)}\}}$ depends on $|e|$ but not on w_e. Then, when $|e| \leq 7$, the optimal values of β for the following optimization problem depend only on $|e|$, and are equal to those listed in Table 3.1.*

$$\begin{aligned} \min_{\{\mathcal{T}_{v\tilde{v}}\}_{\{v,\tilde{v}\}\in E_o},\beta} \quad & \max_{w_e} \quad \beta \\ s.t. \quad & w_e(S) \leq \sum_{v\in S,\tilde{v}\in e/S} \mathcal{T}_{v\tilde{v}}(w_e) \leq \beta w_e(S), \quad \text{for all } S \in 2^e \\ & w_e \text{ is normalized, symmetric and submodular.} \end{aligned} \tag{3.8}$$

Remark 3.1.2. *Although we were able to prove optimality of linear solutions (Theorem 3.1.3) only for small values of $|e|$, we conjecture the results to be true for all $|e|$.*

The following corollary shows that if the weights w_e of hyperedges in a hypergraph are generated from graph cuts of a latent weighted graph, then the projected weights of hyperedges are proportional to the corresponding weights in the latent graph.

Corollary 3.1.4. *Suppose that $G_e = (V^{(e)}, E^{(e)}, w^{(e)})$ is a latent graph that generates hyperedge weights w_e according to the following procedure: for any $S \subseteq e$, $w_e(S) = \sum_{v\in S,\tilde{v}\in e/S} w^{(e)}_{v\tilde{v}}$. Then, equation (3.7) establishes that $w^{*(e)}_{v\tilde{v}} = \beta^{(e)} w^{(e)}_{v\tilde{v}}$, for all $v\tilde{v} \in E^{(e)}$, with $\beta^{(e)} = \frac{2^{|e|}-2}{|e|(|e|-1)}$.*

Corollary 3.1.4 establishes consistency of the linear map (3.7), and also shows that the min-max optimal approximation ratio for linear functions equals $\Omega(2^{|e|}/|e|^2)$. An independent line of work [41], based on Gomory-Hu trees (non-linear), established that submodular functions represent nonnegative solutions of the optimization problem (3.2) with $\beta^{(e)} = |e| - 1$. Therefore, an unrestricted solution of the optimization problem (3.2) ensures that $\beta^{(e)} \leq |e| - 1$.

For those practical applications that involve hypergraphs with small, constant $|e|$, the Gomory-Hu tree approach in this case is suboptimal in approximation ratio compared to (3.7). The expression (3.7) can be rewritten as $w^{*(e)} = M\, w_e$, where M is a matrix that only depends on $|e|$. Hence, the projected weights can be computed in a very efficient and simple manner, as opposed to constructing the Gomory-Hu tree or solving (3.2) directly. In the rare case that one has to deal with hyperedges for which $|e|$ is large, the Gomory-Hu tree approach and a solution of (3.2) may be preferred. However, in this case, we suggest to use the approach based on $p-$Laplacian of submodular hypergraphs which is proposed in the next chapter.

3.2 Applications

In this section, we will introduce a collection of applications to evaluate the proposed clique-expansion-based inhomogeneous hypergraph clustering methods, including network motif clustering (3.2.1), structural learning over ranking data (3.2.2), subspace clustering (3.2.3). In the following, for brevity, we term the inhomogeneous method as InH-partition, while the homogeneous method as H-partition.

3.2.1 Network motif clustering

Real-world networks exhibit rich higher-order connectivity patterns frequently referred to as network motifs [42]. Motifs are special subgraphs of the graph and may be viewed as hyperedges of a hypergraph over the same set of vertices. Recent work has shown that hypergraph clustering based on motifs may be used to learn hidden high-order organization patterns in networks [39, 36, 40]. However, this approach treats all vertices and edges within the motifs in the same manner, and hence ignores the fact that each structural unit within the motif may have a different relevance or different role. As a result, the vertices of the motifs are partitioned with a uniform cost. However, this assumption is hardly realistic as in many real networks, only some vertices of higher-order structures may need to be clustered together. Hence, inhomogeneous hyperedges are expected to elucidate more subtle high-order organizations of network. We illustrate the utility of InH-

partition on the Florida Bay foodweb [43] and compare our findings to those of [39].

The Florida Bay foodweb comprises 128 vertices corresponding to different species or organisms that live in the Bay, and 2106 directed edges indicating carbon exchange between two species. The Foodweb essentially represents a layered flow network, as carbon flows from so-called producers organisms to high-level predators. Each layer of the network consists of "similar" species that play the same role in the food chain. Clustering of the species may be performed by leveraging the layered structure of the interactions. As a network motif, we use a subgraph of four species, and correspondingly, four vertices denoted by v_i, for $i = 1, 2, 3, 4$. The motif captures, among others, relations between two producers and two consumers: The producers v_1 and v_2 both transmit carbons to v_3 and v_4, and all types of carbon flow between v_1 and v_2, v_3 and v_4 are allowed (see Figure 3.1 Left). Such a motif is the smallest structural unit that captures the fact that carbon exchange occurs in one direction between layers, while being allowed freely within layers. The inhomogeneous hyperedge costs are assigned according to the following heuristics: First, as v_1 and v_2 share two common carbon recipients (predators) while v_3 and v_4 share two common carbon sources (prey), we set $w_e(\{v_i\}) = 1$ for $i = 1, 2, 3, 4$, and $w_e(\{v_1, v_2\}) = 0$, $w_e(\{v_1, v_3\}) = 2$, and $w_e(\{v_1, v_4\}) = 2$. Based on the solution of the optimization problem (3.2), one can construct a weighted subgraph whose costs of cuts match the inhomogeneous costs, with $\beta^{(e)} = 1$. The graph is depicted in Figure 3.1 (left).

Our approach is to perform hierarchical clustering via iterative application of the InH-partition method. In each iteration, we construct a hypergraph by replacing the chosen motif subnetwork by an hyperedge. The result is shown in Figure 3.1. At the first level, we partitioned the species into three clusters corresponding to producers, primary consumers and secondary consumers. The producer cluster is homogeneous in so far that it contains only producers, a total of nine of them. At the second level, we partitioned the obtained primary-consumer cluster into two clusters, one of which almost exclusively comprises invertebrates (28 out of 35), while the other almost exclusively comprises forage fishes. The secondary-consumer cluster is partitioned into two clusters, one of which comprises top-level predators, while the other cluster mostly consists of predatory fishes and birds. Overall, we recovered five clusters that fit five layers ranging from producers to top-level consumers.

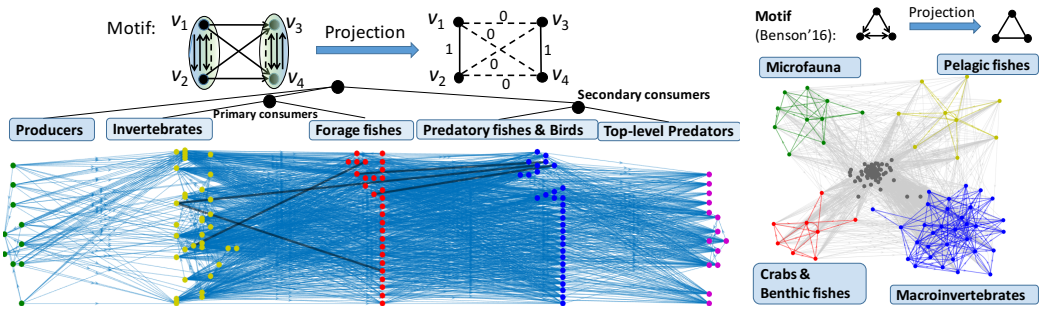


Figure 3.1: Motif clustering in the Florida Bay foodweb. Left: inhomogeneous case. Left-top: Hyperedge (network motif) & the weighted induced subgraph; Left-bottom: Hierarchical clustering structure and five clusters via InH-partition. The vertices belonging to different clusters are distinguished by the colors of vertices. Edges with a uni-direction (right to left) are colored black while other edges are kept blue. Right: Homogeneous partitioning [39] with four clusters. Grey vertices are not connected by motifs and thus unclassified.

It is easy to check that the producer, invertebrate and top-level predator clusters exhibit high functional similarity of species ($> 80\%$). An exact functional classification of forage and predatory fishes is not known, but our layered network appears to capture an overwhelmingly large number of prey-predator relations among these species. Among the 1714 edges, obtained after removing isolated vertices and detritus species vertices, only five edges point in the opposite direction from a higher to a lower-level cluster, two of which go from predatory fishes to forage fishes. Detailed information about the species and clusters is provided in the Table A.1.

In comparison, the related work of Benson et al. [39] which used homogeneous hypergraph clustering and triangular motifs reported a very different clustering structure. The corresponding clusters covered less than half of the species (62 out of 128) as many vertices were not connected by the triangle motif; in contrast, 127 out of 128 vertices were covered by our choice of motif. We attribute the difference between our results and the results of [39] to the choices of the network motif. A triangle motif, used in [39], leaves a large number of vertices unclustered and fails to enforce a hierarchical network structure. On the other hand, our fan motif with homogeneous weights produces a giant cluster as it ties all the vertices together, and the hierarchical decomposition is only revealed when the fan motif is used with inhomogeneous weights. In order to identify hierarchical network structures, instead of hypergraph clustering, one may use topological sorting to rank species

based on their carbon flows [44]. Unfortunately, topological sorting cannot use biological side information and hence fails to automatically determine the boundaries of the clusters.

3.2.2 Learning the riffled independence structure of ranking data

Learning probabilistic models for ranking data has attracted significant interest in social and political sciences as well as in machine learning [45, 46]. Recently, a probabilistic model, termed the riffled-independence model, was shown to accurately describe many benchmark ranked datasets [47]. In the riffled independence model, one first generates two rankings over two disjoint sets of elements independently, and then riffle shuffles the rankings to arrive at an interleaved order. The structure learning problem in this setting reduces to distinguishing the two categories of elements based on limited ranking data. More precisely, let Q be the set of candidates to be ranked, with $|Q| = n$. A full ranking is a bijection $\sigma : Q \to [n]$, and for an $a \in Q$, $\sigma(a)$ denotes the position of candidate a in the ranking σ. We use $\sigma(a) < (>)\sigma(b)$ to indicate that a is ranked higher (lower) than b in σ. If $S \subseteq Q$, we use $\sigma_S : S \to [|S|]$ to denote the ranking σ projected onto the set S. We also use $S(\sigma) \triangleq \{\sigma(a)|a \in S\}$ to denote the subset of positions of elements in S. Let $\mathbb{P}(E)$ denote the probability of the event E. Riffled independence asserts that there exists a riffled-independent set $S \subset Q$, such that for a fixed ranking σ' over $[n]$,

$$\mathbb{P}(\sigma = \sigma') = \mathbb{P}(\sigma_S = \sigma'_S)\mathbb{P}(\sigma_{Q/S} = \sigma'_{Q/S})\mathbb{P}(S(\sigma) = S(\sigma')).$$

Suppose that we are given a set of rankings $\Sigma = \{\sigma^{(1)}, \sigma^{(2)}, ..., \sigma^{(m)}\}$ drawn independently according to some probability distribution $\mathbb{P}$. If $\mathbb{P}$ has a riffled-independent set S^*, the structure learning problem is to find S^*. In [47], the described problem was cast as an optimization problem over all possible subsets of Q, with the objective of minimizing the Kullback-Leibler divergence between the ranking distribution with riffled independence and the empirical distribution of Σ [47]. A simplified version of the optimization problem reads

as

$$\arg\min_{S\subset Q} \mathcal{F}(S) \triangleq \sum_{(i,j,k)\in\Omega^{cross}_{S,\bar{S}}} I_{i;j,k} + \sum_{(i,j,k)\in\Omega^{cross}_{\bar{S},S}} I_{i;j,k}, \tag{3.9}$$

where $\Omega^{cross}_{A,B} \triangleq \{(i,j,k)|i \in A, j,k \in B\}$, and where $I_{i;j,k}$ denotes the estimated mutual information between the position of the candidate i and two "comparison candidates" j,k. If $1_{\sigma(j)<\sigma(k)}$ denotes the indicator function of the underlying event, we may write

$$\begin{aligned} I_{i;j,k} \triangleq & \hat{I}(\sigma(i); 1_{\sigma(j)<\sigma(k)}) \\ = & \sum_{\sigma(i)} \sum_{1_{\sigma(j)<\sigma(k)}} \hat{\mathbb{P}}(\sigma(i), 1_{\sigma(j)<\sigma(k)}) \log \frac{\hat{\mathbb{P}}(\sigma(i), 1_{\sigma(j)<\sigma(k)})}{\hat{\mathbb{P}}(\sigma(i))\hat{\mathbb{P}}(1_{\sigma(j)<\sigma(k)})}, \end{aligned} \tag{3.10}$$

where $\hat{\mathbb{P}}$ denotes an estimate of the underlying probability. If i and j,k are in different riffled-independent sets, the estimated mutual information $\hat{I}(\sigma(i); 1_{\sigma(j)<\sigma(k)})$ converges to zero as the number of samples increases. When the number of samples is small, one may use mutual information estimators described in [48, 49, 50].

One may recast the above problem as an InH-partition problem over a hypergraph where each candidate represents a vertex in the hypergraph, and $I_{i;j,k}$ represents the inhomogeneous cost $w_e(\{i\})$ for the hyperedge $e = \{i,j,k\}$. Note that as mutual information $\hat{I}(\sigma(i); 1_{\sigma(j)<\sigma(k)})$ is in general asymmetric, one would not have been able to use H-partitions. The optimization problem reduces to $\min_S \mathrm{vol}(\partial S)$. The two optimization tasks are different, and we illustrate next that the InH-partition outperforms the original optimization approach AnchorsPartition (Apar) [47] both on synthetic data and real data.

Synthetic data. We first compare the InH-partition (InH-Par) method with the AnchorsPartition (APar) technique proposed in [47] on synthetic data. Note that APar is assumed to know the correct size of the riffled-independent sets while InH-partition automatically determines the sizes of the parts. We set the number of elements to $n = 16$, and partition them into a pair $(S^*, \bar{S}^*)$, where $|S^*| = q$, $1 \le q \le n$. For a sample set size m, we first independently choose scores $s_i, \bar{s}_i \sim$Uniform([0,1]) for $i \in V$, and then generate m rankings via the following procedure: We first use the Plackett-

Luce model [51] with parameters $s_i, i \in S^*$ and $\bar{s}_i, i \in \bar{S}^*$, to generate σ_{S^*} and $\sigma_{\bar{S}^*}$. Then, we interleave σ_{S^*} and $\sigma_{\bar{S}^*}$, which were sampled uniformly at random without replacement, to form σ. The performance of the method is characterized via the success rate of full recovery of $(S^*, \bar{S}^*)$. The results of various algorithms based on 100 independently generated sample sets are listed in Figure 3.2 a) and b). For almost all m, InH-partition outperforms APar. Only when $q = 4$ and the sample size m is large, InH-partition may offer worse performance than Apar. The explanation for this finding is that InH-partition performs a normalized cut that tends to balance the sizes of different classes. With regard to the computational complexity of the methods, both require one to evaluate the mutual information of all triples of elements at the cost of $O(mn^3)$ operations. To reduce the time complexity of this step, one may sample each triple independently with probability r. Results pertaining to triple-sampling with $m = 10^4$ are summarized in Figure 3.2 c) and d). The InH-partition can achieve high success rate 80% even when only a small fraction of triples ($r < 0.2$) is available. On the other hand, APar only works when almost all triples are sampled ($r > 0.7$).

To further test the performance of InH-partition, instead of using the previously described s_i values as the parameters for Plakett-Luce model, we use the values s_i^3 instead. This choice of parameters further restricts the positions of the candidates within S^* and $\bar{S}^*$. Hence, the mutual information of interest is closer to zero and hence harder to estimate. The results for this setting are shown in part e) and f) of Figure 3.2. As may be seen, in this setting, the performance of APar is poor while that of InH-partition changes little.

Real data Here, we first analyzed the Irish House of Parliament election dataset (2002) [52]. The dataset consists of 2490 ballots fully ranking 14 candidates. Those candidates came from different parties, listed in Table 3.2. Fianna Fáil (F.F.) and Fine Gael (F.G.) are the two largest (and rival) Irish political parties. Using InH-partition (InH-Par), one can split the candidates iteratively into two sets (see Figure 3.3) which yield to meaningful clusters that correspond to large parties: $\{1, 4, 13\}$ (F.F.), $\{2, 5, 6\}$ (F.G.), $\{7, 8, 9\}$ (Ind.). We ran InH-partition to split the 14 election candidates to obtain a hierarchical clustering structure as the one shown in Figure 3.4. We compared InH-partition with Apar based on their performance in detecting these three clusters using a small training set: We independently sampled

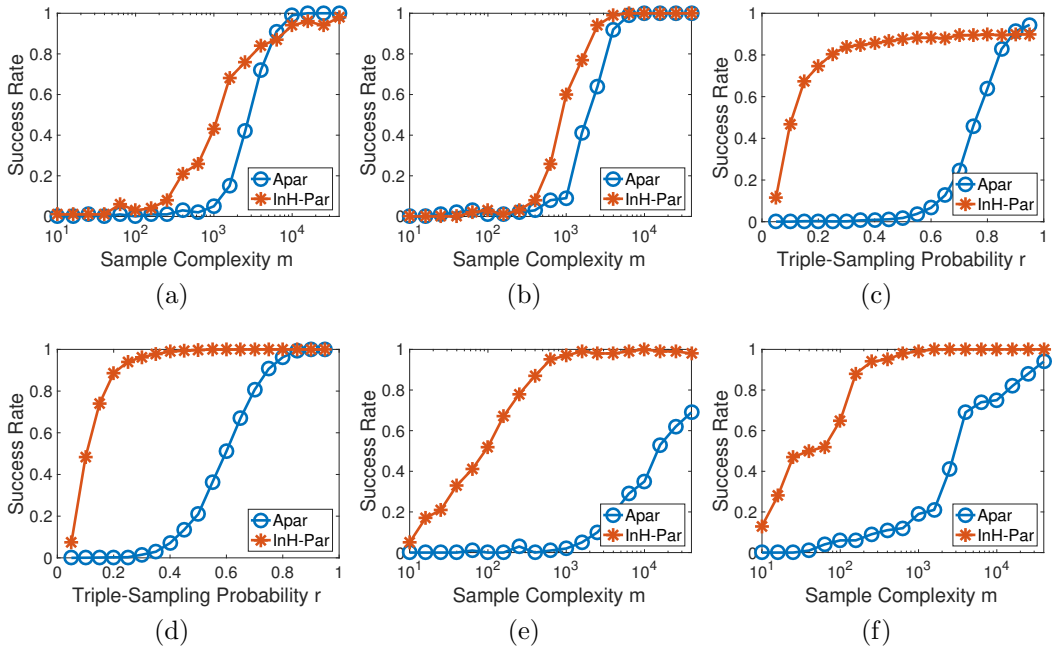
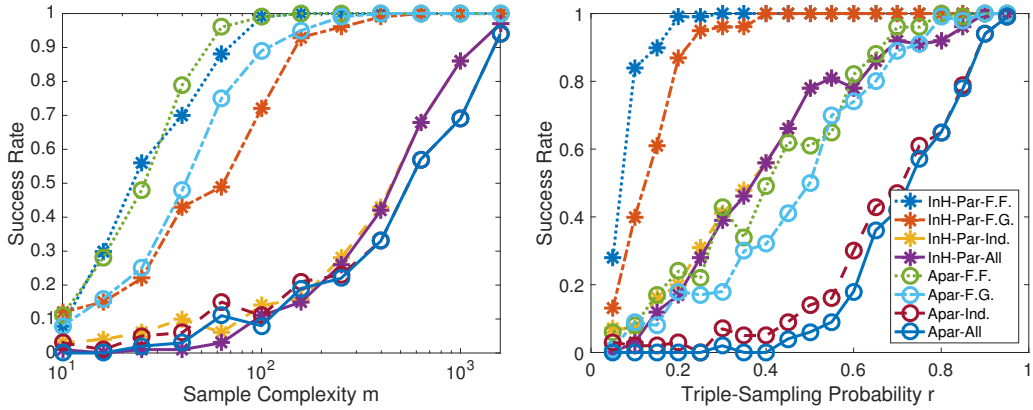


Figure 3.2: Success rate vs Sample Complexity & Triple-sampling Rate. a),c): $q = 4$ with scores s_i; b),d): $q = 8$ with scores s_i; e) $q = 4$ with scores s_i^3; f): $q = 8$ with scores s_i^3.

m rankings 100 times and executed both algorithms to partition the set of candidates iteratively. During the partitioning procedure, "party success" was declared if one exactly detected one of the three party clusters ("F.F.", "F.G." & "Ind."). "All" was used to designate that all three party clusters were detected completely correctly. InH-partition outperforms Apar in recovering the cluster Ind. and achieved comparable performance for cluster F.F., although it performs a little worse than Apar for cluster F.G.; InH-partition also offers superior overall performance compared to Apar. We also compared InH-partition with APar in the large sample regime ($m = 2490$), using only a subset of triple comparisons (hyperedges) sampled independently with probability r (This strategy significantly reduces the complexity of both algorithms). The averaged results based on 100 independent tests are depicted in Figure 3.3 a).

Table 3.2: List of candidates from the Meath Constituency Election in 2002 (reproduced from [47, 52]).

	Candidate	Party
1	Brady, J.	Fianna Fáil
2	Bruton, J.	Fine Gael
3	Colwell, J.	Independent
4	Dempsey, N.	Fianna Fáil
5	English, D.	Fine Gael
6	Farrelly, J.	Fine Gael
7	Fitzgerald, B.	Independent
8	Kelly, T.	Independent
9	O'Brien, P.	Independent
10	O'Byrne, F.	Green Party
11	Redmond, M.	Christian Solidarity
12	Reilly, J.	Sinn Féin
13	Wallace, M.	Fianna Fáil
14	Ward, P.	Labour



(a) (b)

Figure 3.3: Clusters detected in the Irish election dataset: a) Success rate vs Sample Complexity; b) Success rate vs Triple-Sampling Probability.

In addition, we performed the same structure learning task on the sushi preference ranking dataset [53]. This dataset consists of 5000 full rankings of ten types of sushi. The different types of sushi evaluated are listed in

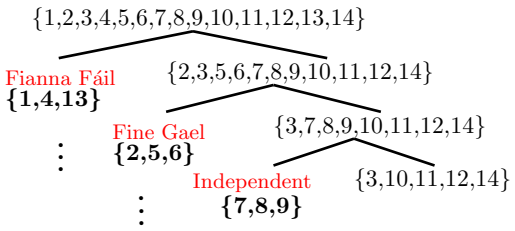


Figure 3.4: Hierarchical partitioning structure of Irish election detected by InH-Par.

Table 3.3: List of 10 sushi from the sushi preference dataset (reproduced from [53]).

	Sushi	Type		Candidate	Party
1	ebi	shrimp	6	sake	salmon roe
2	anago	sea eel	7	tamago	egg
3	maguro	tuna	8	toro	fatty tuna
4	ika	squid	9	tekka-maki	tuna roll
5	uni	sea urchin	10	kappa-maki	cucumber roll

Table 3.3. We ran InH-partition to split the ten sushi types to obtain a hierarchical clustering structure as the one shown in Figure 3.5. The figure reveals two meaningful clusters, $\{5, 6\}$ (uni,sake) and $\{3, 8, 9\}$ (tuna-related sushi): The sushi types labeled by 5 and 6 have the commonality of being expensive and branded as "daring, luxury sushi," while sushi types labeled by $3, 8, 9$ all contain tuna. InH-partition cannot detect the so-called "vegeterian-choice sushi" cluster $\{7, 10\}$, which was recovered by Apar [47]. This may be a consequence of the ambiguity and overlap of clusters, as the cluster $\{4, 7\}$ may also be categorized as "rich in lecithin". The detailed comparisons between InH-partition and APar are performed based on their ability to detect the two previously described standard clusters, $\{5, 6\}$ and $\{3, 8, 9\}$, using small training sets. The averaged results based on 100 independent tests are depicted in Figure 3.6 a). As may be seen, InH-partition outperforms APar in recovering both the clusters (uni,sake) and (tuna sushi), and hence is superior to APar when learning both classes simultaneously. We also compared InH-partition and APar in the large sample regime ($m = 5000$) while using only a subset of triples. The averaged results over 100 sets of independent samples are shown Figure 3.6 b), again indicating the robustness of InH-partition to missing triple information.

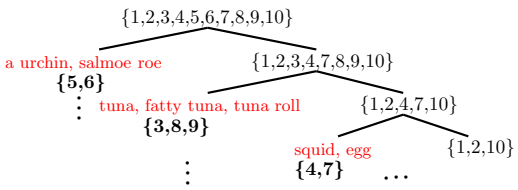
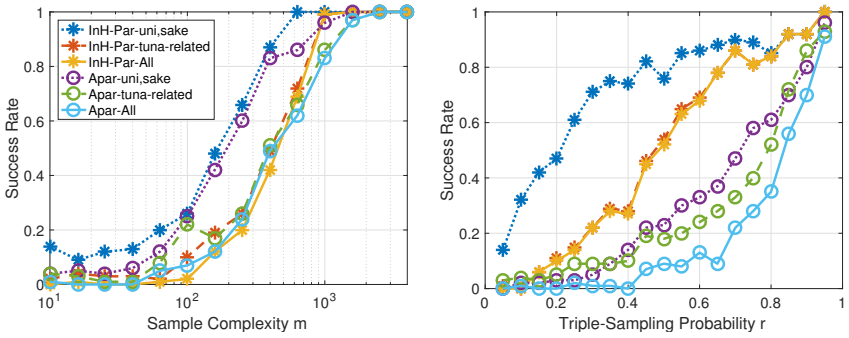


Figure 3.5: Hierarchical partitioning structure of sushi preference detected by InH-Par.



(a) (b)

Figure 3.6: Clusters detected in the sushi preference dataset: a) Success rate vs Sample Complexity; b) Success rate vs Triple-Sampling Probability.

3.2.3 Subspace segmentation

Subspace segmentation is an extension of traditional data segmentation problems that has the goal to partition data according to their intrinsically embedded subspaces. Among subspace segmentation methods, those based on hypergraph clustering exhibit superior performance compared to others [54]. They also exhibit other distinguishing features, such as loose dependence on the choice of parameters [11], robustness to outliers [31, 33], and clustering robustness and accuracy [55].

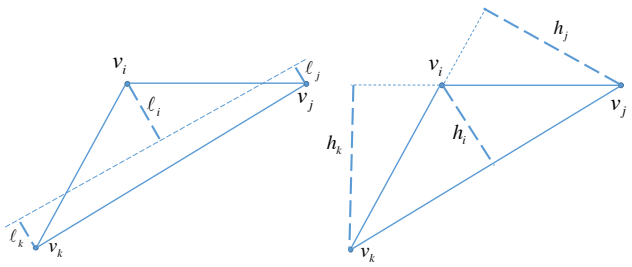
Hypergraph clustering algorithms are exclusively homogeneous: If the intrinsic affine space is p-dimensional (p-D), the algorithms use ψ-uniform ($\psi > p+1$, typically set to $p+2$) hypergraphs $\mathcal{H} = (V, E)$, where the vertices in V correspond to observed data vectors and the hyperedges in E are chosen ψ-tuples of vertices. To each hyperedge e in the hypergraph $\mathcal{H}$ one assigns a weight w_e, typically of the form $w_e = \exp(-d_e^2/\theta^2)$, where d_e describes the deviation needed to fit the corresponding ψ-tuple of vectors into a p-D affine subspace, and θ represents a tunable parameter obtained

by cross validation [11] or computed empirically [55]. A small value of d_e corresponds to a large value of w_e, and indicates that ψ-tuples of vectors in e tend to be clustered together. As a good fit of the subspace yields a large weight for the corresponding hyperedge, hypergraph clustering tends to avoid cutting hyperedges of large weight and thus mostly groups vectors within one subspace together. The performance of the methods varies due to different techniques used for computing the deviation d_e and for sampling the hyperedges. Some widely used deviations include $d_e^{\mathrm{H}-1}$, defined as the mean Euclidean distance to the optimal fitted affine subspace [11, 31, 33, 32], and the *polar curvature* (PC) [55], both of which lead to a homogeneous partition. Instead, we propose to use an inhomogeneous deviation defined as

$$d_e^{\mathrm{InH}}(\{v\}) = \text{Euclidean distance between } v \text{ and the affine subspace generated by } e/\{v\}, \text{ for all } v \in e.$$

This deviation measures the "distance" needed to fit v into the subspace supported by e/v and will be used to construct inhomogeneous cost functions $w_e(\cdot)$ via $w_e(\{v\}) = \exp[-d_e^{\mathrm{InH}}(\{v\})^2/\theta^2]$, as described in what follows. Note that the choice of a "good" deviation is still an open problem, which may depend on specific datasets. Hence, to make a comprehensive comparison, besides $d_e^{\mathrm{H}-1}$ and PC, we also made use of another homogeneous deviation, $d_e^{\mathrm{H}-2} = \sum_{v\in e} d_e^{\mathrm{InH}}(\{v\})/|e|$ which is the average of all the defined inhomogeneous deviations. Comparing the results obtained from d_e^{InH} with $d_e^{\mathrm{H}-2}$ will highlight the improvements obtained from InH-partition, rather than from the choice of the deviation. The inhomogeneous form of deviation $d_e^{\mathrm{InH}}(\cdot)$ has a geometric interpretation based on the polytopes ($p = 1$) shown in Figure 3.7 (with $d_e^{\mathrm{H}-1}$ and $d_e^{\mathrm{H}-2}$). There, $d_e^{\mathrm{InH}}(\{v\})$ is the distance of $\{v\}$ from the hyperplane spanned by $e/\{v\}$. The induced inhomogeneous weight $w_e(\{v\}) = \exp(-d_e^{\mathrm{InH}}(\{v\})^2/\theta^2)$ may be interpreted as the cost of separating $\{v\}$ away from the other points (vertices) in e.

All hypergraph-partitioning based subspace segmentation algorithms essentially use the NCut procedure described in the main text, but their performances vary due to different approaches for constructing the hypergraphs. Three steps in the clustering procedure are key to the performance quality: The first is to quantify the deviation to fit a collection of vectors into an affine subspace; the second is to choose the parameter θ; the third is to sam-



$$d_e^{\mathrm{H}-1} = (\ell_i + \ell_j + \ell_k)/3$$
$$d_e^{\mathrm{H}-2} = (h_i + h_j + h_k)/3$$
$$d_e^{\mathrm{InH}}(\{v_i\}) = h_i$$

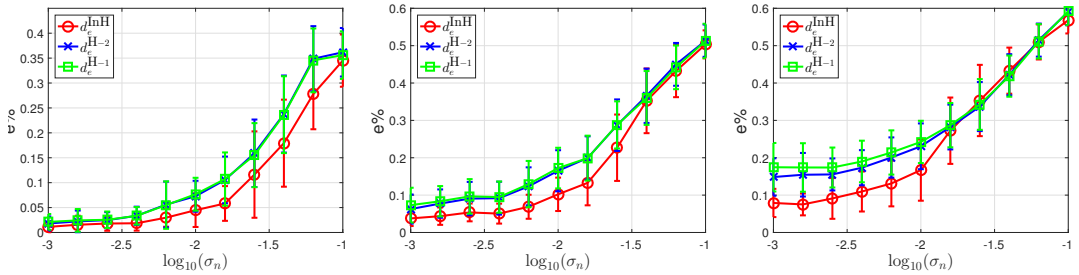
Figure 3.7: Illustration of the deviation ($p = 1$) used for subspace segmentation.

ple ψ-tuples of vectors, i.e., choose the hyperedges of the hypergraph. For fairness of comparison, in all our experiments we computed an inhomogeneous deviation d_e for the hyperedge e instead of a homogeneous one in the first step, and kept the other two key steps the same as used in the standard literature. In particular, we performed hyperedge sampling uniformly at random for the experiments pertaining to k-line segmentation; we used the same hyperedge sampling procedure as that of SCC [55] for the experiments pertaining to motion segmentation. The reason for these two different types of settings are to assess the contribution of InH methods, rather than the sampling procedure.

Synthetic data: Our first experiment pertains to segmenting k-lines in a 3D Euclidean space ($D = 3, p = 1, k = 2, 3, 4$). The k-lines all pass through the origin, and their directions, listed in Table 3.4, are such that the minimal angles between two lines are restricted to 30 degree; 40 points are sampled uniformly from the segment of each line lying in the unit ball so there are $40k$ points in total. Each point is independently corrupted by 3D mean-zero Gaussian noise with covariance matrix $\theta_n^2\mathbf{I}$. We determined the parameter θ through cross validation and uniformly at random picked $100 \times k^2$ many triples. We computed the percentage of misclassified points based on 50 independent tests; the misclassification rate is denoted by $e\%$ and the results are shown in Figure 3.8. The InH-partition only has 50% of the misclassification errors of H-partition, provided that the noise is small ($\theta_n < 0.01$). To see why this may be the case, let us consider a triple of datapoints $\{v_i, v_j, v_k\}$ where v_i and v_j belong to the same cluster, while v_k may belong to a different cluster. The line that goes through v_i and v_j is close to the true affine subspace when the noise is small and thus the distance from the third point v_k to this line can serve as a precise indicator whether

Table 3.4: The directions of the k-lines.

k =2	k =3	k =4
(0.97,0.26,0.00) (0.97,-0.26,0.00)	(0.95,0.30,0.00) (0.95,-0.15,0.26) (0.95,-0.15,-0.26)	(0.93,0.37,0.00) (0.93,0.00,0.37) (0.93,-0.37,0.00) (0.93,0.00,-0.37)



(a) (b) (c)

Figure 3.8: Misclassification rate (mean and standard deviation) vs noise level: a) $k = 2$; b) $k = 3$; c) $k = 4$.

v_k lies within the same true affine subspace. When the noise is high, the InH-partition also performs better when the number of classes is $k = 2$, but starts to deteriorate in performance as k increases. The reason behind this phenomena is as follows: inhomogeneous costs of a hyperedge provide more accurate information about the subspaces than the homogeneous costs when at least two points of the hyperedge belong to the same line cluster. This is due to the definition of the deviation d_e^{inH}; but hyperedges of this type become less likely as k increases.

Real data: The second problem we investigated in the context of subspace clustering is motion segmentation. Motion segmentation, a widely used application in computer vision, is the task of clustering point trajectories extracted from a video of a scene according to different rigid-body motions. The problem can be reduced to a subspace clustering problem as all the trajectories associated with one motion lie in one specified 3D affine subspace ($p = 3$) [56]. We evaluate the performance of the InH-partition method over the well-known motion segmentation dataset, Hopkins155 [57]. This dataset consists of 155 sequences of two and three motions from three categories of scenes: Checkerboard, traffic and articulated sequences. Our experiments show the InH-partition algorithm outperforms the benchmark

algorithms based on the use of H-partitions over this dataset including spectral curvature clustering technique (SCC [55]). To make the comparison fair, we simply replaced the homogeneous distance *polar curvature* in SCC with the inhomogeneous distance d_e^{InH}, the homogeneous distances $d_e^{\mathrm{H}-1}$ and $d_e^{\mathrm{H}-2}$, and keep all other steps the same. We also evaluated the performance of some other methods, including Generalized PCA (GPCA) [58], Local Subspace Affinity [59], Agglomerative Lossy Compression (ALC) [60], and Sparse Subspace Clustering (SSC) [61]. The results based on the average over 50 runs for each video are shown in Table 3.5.

As may be seen, InH-partition outperforms all methods except for SSC (not based on hypergraph clustering), which shows the superiority of replacing H-hyperedges with inhomogeneous ones. Although InH-partition fails to outperform SSC, it has significantly lower complexity and is much easier to use and implement in practice. In addition, some recent algorithms based on H-partitions may leverage the complex hyperedge-sampling steps for this application [62], and we believe that the InH-partition method can be further improved by changing the sampling procedure, and made more appropriate for inhomogeneous hypergraph clustering as opposed to SSC.

Table 3.5: Misclassification rates $e\%$ for the Hopkins 155 dataset. (MN: mean; MD: median)

	Two Motions								Three Motions							
Method	Chck.(78)		Trfc.(31)		Artc.(11)		All(120)		Chck.(26)		Trfc.(7)		Artc.(2)		All(115)	
	MN	MD	MN	MD	MN	MD	MN	MD	MN	MD	MN	MD	MN	MD	MN	MD
GPCA [58]	6.09	1.03	1.41	0.00	2.88	0.00	4.59	0.38	31.95	32.93	19.83	19.55	16.85	16.85	28.66	28.26
LSA [59]	2.57	0.27	5.43	1.48	4.10	1.22	3.45	0.59	5.80	1.77	25.07	23.79	7.25	7.25	9.73	2.33
ALC [60]	1.49	0.27	1.75	1.51	10.70	0.95	2.40	0.43	5.00	0.66	8.86	0.51	21.08	21.08	6.69	0.67
SSC [61]	1.12	0.00	0.02	0.00	0.62	0.00	0.82	0.00	2.97	0.27	0.58	0.00	1.42	1.42	2.45	0.20
SCC [55]	1.77	0.00	0.63	0.14	4.02	2.13	1.68	0.07	6.23	1.70	1.11	1.40	5.41	5.41	5.16	1.58
H+d_e^{H-1}	12.27	5.06	14.91	9.94	12.85	3.66	12.92	6.01	22.13	23.98	21.99	18.12	19.79	19.79	21.97	20.45
H+d_e^{H-2}	4.20	0.43	0.33	0.00	1.53	0.10	2.93	0.06	7.05	2.22	7.02	3.98	6.47	6.47	7.01	2.12
InH-par	1.69	0.00	0.61	0.22	1.22	0.62	1.40	0.04	4.82	0.69	2.46	0.60	4.23	4.23	4.06	0.65

CHAPTER 4

SUBMODULAR HYPERGRAPHS: SPECTRAL GRAPH THEORY AND SPECTRAL CLUSTERING

In this chapter, we further consider other algorithms to solve the partitioning/clustering problem (3.1). Although clique-expansion-based methods perform extremely well on all applications described in Chapter 3, clique expansion methods in general have two drawbacks. First, the spectral clustering algorithm for graphs used in the second step is merely quadratically optimal (see inequality (2.3)). Second, for large hyperedges, clique expansion may cause large distortions: As shown in Theorem 3.1.1, there will be an additional constant approximation factor β^*. Therefore, the distortion may be as large as $\Omega(2^{\zeta}(E)/\zeta(E)^2)$ for linear clique expansion and $\Omega(\zeta(E))$ for non-linear clique expansion (e.g. via Gomory-Hu trees).

There have been some proposed strategies to avoid the above two problems for graph and homogeneous hypergraph clustering. To address the quadratic optimality issue in graph clustering, Amghibech [63] introduced the notion of p-Laplacians of graphs and derived Cheeger-type inequalities for the second smallest eigenvalue of a p-Laplacian, with $p > 1$, of a graph. These results motivated Bühler and Hein's work [64] on spectral clustering based on p-Laplacians that provided tighter approximations of the Cheeger constant. Szlam and Bresson [65] showed that the 1-Laplacian allows one to exactly compute the Cheeger constant, but at the cost of computational hardness [66]. Very little is known about the use of p-Laplacians for hypergraph clustering and their spectral properties. To address the clique expansion problem for homogeneous hypergraphs, Hein et al. [9] introduced a clustering method that avoids expansions and works directly with the total variation of homogeneous hypergraphs, without investigating the spectral properties of the operator. The only other line of work trying to mitigate the projection problem is due to Louis [8], who used a natural extension of 2-Laplacians for homogeneous hypergraphs, derived quadratically-optimal Cheeger-type inequalities and proposed a semidefinite programing (SDP) based algorithm

whose complexity scales with the size of the largest hyperedge in the hypergraph.

In this chapter, we will generalize the above described strategies to apply to submodular hypergraphs. We first define p-Laplacians for submodular hypergraphs and generalize the corresponding discrete nodal domain theorems [67, 68] and higher-order Cheeger inequalities. An analytical obstacle in the development of such a theory is the fact that p-Laplacians of hypergraphs are operators that act on vectors and produce *sets of values*. Consequently, operators and eigenvalues have to be defined in a set-theoretic manner. Then, based on the newly established spectral hypergraph theory, we propose two spectral clustering methods that learn the second smallest eigenvalues of 2- and 1-Laplacians. The algorithm for 2-Laplacian eigenvalue computation is based on an SDP framework and can provably achieve quadratic optimality with an $O(\sqrt{\zeta(E)})$ approximation constant. The algorithm for 1-Laplacian eigenvalue computation is based on the inverse power method (IPM) [69] that only has convergence guarantees. The key novelty of the IPM-based method is that the critical inner-loop optimization problem of the IPM is efficiently solved by algorithms recently developed for decomposable submodular minimization [19, 20, 70]. Although without performance guarantees, given that the 1-Laplacian provides the tightest approximation guarantees, the IPM-based algorithm – as opposed to the clique expansion method [71] – performs very well empirically even when the size of the hyperedges is large. This fact is illustrated on several UC Irvine machine learning datasets available from [72].

4.1 p-Laplacians for submodular hypergraphs and the spectra

We start our discussion by defining the notion of a p-Laplacian operator for submodular hypergraphs. We find the following definitions useful for our subsequent exposition.

Let $\mathrm{sgn}(\cdot)$ be the sign function defined as $\mathrm{sgn}(a) = 1$, for $a > 0$, $\mathrm{sgn}(a) = -1$, for $a < 0$, and $\mathrm{sgn}(a) = [-1, 1]$, for $a = 0$. For all $v \in V$, define the entries of a vector φ_p over $\mathbb{R}^N$ according to $(\varphi_p(x))_v = |x_v|^{p-1}\mathrm{sgn}(x_v)$. Let $\|x\|_{\ell_p,\mu} = (\sum_{v\in V} \mu_v |x_v|^p)^{1/p}$ and $\mathcal{S}_{p,\mu} \triangleq \{x \in \mathbb{R}^N | \|x\|_{\ell_p,\mu} = 1\}$. For a function

Φ over $\mathbb{R}^N$, let $\Phi|_{\mathcal{S}_{p,\mu}}$ stand for Φ restricted to $\mathcal{S}_{p,\mu}$.

Definition 4.1.1. *The p-Laplacian operator of a submodular hypergraph, denoted by $\triangle_p$ ($p \geq 1$), is defined for all $x \in \mathbb{R}^N$ according to*

$$\langle x, \triangle_p(x) \rangle \triangleq Q_p(x) = \sum_{e \in E} \vartheta_e f_e(x)^p. \tag{4.1}$$

Hence, $\triangle_p(x)$ may also be specified directly as an operator over $\mathbb{R}^N$ that reads as

$$\triangle_p(x) = \begin{cases} \sum_{e \in E} \vartheta_e f_e(x)^{p-1} \nabla f_e(x) & p > 1, \\ \sum_{e \in E} \vartheta_e \nabla f_e(x) & p = 1. \end{cases}$$

Definition 4.1.2. *A pair $(\lambda, x) \in \mathbb{R} \times \mathbb{R}^N / \{\mathbf{0}\}$ is called an eigenpair of the p-Laplacian $\triangle_p$ if $\triangle_p(x) \cap \lambda U \, \varphi_p(x) \neq \emptyset$.*

As $f_e(\mathbf{1}) = 0$, we have $\triangle_p(\mathbf{1}) = 0$, so that $(0, \mathbf{1})$ is an eigenpair of the operator $\triangle_p$. A p-Laplacian operates on vectors and produces sets. In addition, since for any $t > 0$, $\triangle_p(tx) = t^{p-1} \triangle_p(x)$ and $\varphi_p(tx) = t^{p-1} \varphi_p(x)$, (tx, λ) is an eigenpair if and only if (x, λ) is an eigenpair. Hence, one only needs to consider normalized eigenpairs: In our setting, we choose eigenpairs that lie in $\mathcal{S}_{p,\mu}$ for a suitable choice for the dimension of the space.

For linear operators, the Rayleigh-Ritz method [73] allows for determining approximate solutions to eigenproblems and provides a variational characterization of eigenpairs based on the critical points of functionals. To generalize the method, we introduce two even functions,

$$\tilde{Q}_p(x) \triangleq Q_p(x)|_{\mathcal{S}_{p,\mu}}, \quad R_p(x) \triangleq \frac{Q_p(x)}{\|x\|^p_{\ell_p,\mu}}.$$

Definition 4.1.3. *A point $x \in \mathcal{S}_{p,\mu}$ is termed a* critical point *of $R_p(x)$ if $0 \in \nabla R_p(x)$. Correspondingly, $R_p(x)$ is termed a* critical value *of $R_p(x)$. Similarly, x is termed a* critical point *of $\tilde{Q}_p$ if there exists a $\sigma \in \nabla Q_p(x)$ such that $P(x)\sigma = 0$, where $P(x)\sigma$ stands for the projection of σ onto the tangent space of $\mathcal{S}_{p,\mu}$ at the point x. Correspondingly, $\tilde{Q}_p(x)$ is termed a* critical value *of $\tilde{Q}_p$.*

The relationships between the critical points of $\tilde{Q}_p(x)$ and $R_p(x)$ and the

eigenpairs of $\triangle_p$ relevant to our subsequent derivations are listed in Theorem 4.1.4.

Theorem 4.1.4. *A pair (λ, x) ($x \in \mathcal{S}_{p,\mu}$) is an eigenpair of the operator $\triangle_p$ 1) if and only if x is a critical point of $\tilde{Q}_p$ with critical value λ, and provided that $p \geq 1$.*
2) if and only if x is a critical point of R_p with critical value λ, and provided that $p > 1$.
3) if x is a critical point of R_p with critical value λ, and provided that $p = 1$.

The critical points of $\tilde{Q}_p$ bijectively characterize eigenpairs for all choices of $p \geq 1$. However, R_p has the same property only if $p > 1$. This is a consequence of the nonsmoothness of the set $\mathcal{S}_{1,\mu}$, which has been observed for graphs as well (See the examples in Section 2.2 in [66]).

Once Theorem 4.1.4 has been established, a standard way to analyze the spectrum of $\triangle_p$ is to study the critical points of $\tilde{Q}_p = Q_p(x)|_{\mathcal{S}_{p,\mu}}$. A crucial component within this framework is the Lusternik-Schnirelman theory that allows one to characterize a series of these critical points. As Q_p and $\mathcal{S}_{p,\mu}$ are symmetric, one needs to use the notion of a Krasnoselski genus, defined below. This type of approach has also been used to study the spectrum of p-Laplacians of graphs, and the readers interested in the mathematical theory behind the derivations are referred to [66, 67] and references therein for more details.

Definition 4.1.5. *Let $A \subset \mathbb{R}^N/\{0\}$ be a closed and symmetric set. The Krasnoselski genus of A is defined as*

$$\gamma(A) = \begin{cases} 0, & \text{if } A = \emptyset, \\ \inf\{k \in \mathbb{Z}^+ | \exists \text{ odd continuous } h : \ A \to \mathbb{R}^k \backslash \{0\}\} \\ \infty & \text{if for any finite } k \in \mathbb{Z}^+, \text{ no such } h \text{ exists.} \end{cases} \tag{4.2}$$

We now focus on a particular subset of $\mathcal{S}_{p,\mu}$, defined as

$$\mathcal{F}_k(\mathcal{S}_{p,\mu}) \triangleq \{A \subseteq \mathcal{S}_{p,\mu} | A = -A, \text{closed}, \gamma(A) \geq k\}.$$

As Q_p may not be differentiable, we apply Chang's generalization of the Lusternik-Schnirelman theorem for *locally Lipschitz continuous* functionals defined on smooth Banach-Finsler manifolds (corresponding to the case $p >$

1) and those defined on piecewise linear manifolds (corresponding to the case $p = 1$).

Definition 4.1.6. *We say $g : \mathcal{S}_{p,\mu} \to \mathbb{R}$ is* locally Lipschitz*: if for each $x \in \mathcal{S}_{p,\mu}$, there exists a neighborhood $\mathcal{N}_x$ of x and a constant C depending on $\mathcal{N}_x$ such that $|g(x') - g(x)| \leq C\|x' - x\|_{\ell_2}$ for any $x' \in \mathcal{S}_{p,\mu} \cap \mathcal{N}_x$.*

Theorem 4.1.7 (Theorem 3.2 [74] and Theorem 4.9 [66])**.** *Suppose function $g : \mathcal{S}_{p,\mu} \to \mathbb{R}$ is* locally Lipschitz*, even, bounded below, then*

$$\min_{A:\mathcal{F}_k(\mathcal{S}_{p,\mu})} \max_{x \in A} g(x) \qquad k = 1, 2, ..., N$$

characterize the critical values of g.

It is easy to check if $\tilde{Q}_p$ is locally Lipschitz, even and bounded below. By invoking the Lusternik-Schnirelman theorem, we claim that there are at least n critical values of $\tilde{Q}_p$ equaling

$$\lambda_k^{(p)} = \min_{A:\mathcal{F}_k(\mathcal{S}_{p,\mu})} \max_{x \in A} \tilde{Q}_p, \qquad k = 1, 2, ..., N. \tag{4.3}$$

Note that as $\mathcal{F}_{k+1}(\mathcal{S}_{p,\mu}) \subseteq \mathcal{F}_k(\mathcal{S}_{p,\mu})$, $\lambda_{k+1}^{(p)} \geq \lambda_k^{(p)}$. Combining (4.3) and Theorem 4.1.4, $\{\lambda_k^{(p)}\}_{k \in [N]}$ are a collection of eigenvalues of p-Laplacian operators $\triangle_p$.

4.2 Discrete nodal domain theorems for $p-$Laplacians

Nodal domain theorems are essential for understanding the structure of eigenvectors of operators and they have been the subject of intense study in geometry and graph theory alike [75]. The eigenfunctions of a Laplacian operator may take positive and negative values. The signs of the values induce a partition of the vertices in V into maximal connected components on which the sign of the eigenfunction does not change: These components represent the nodal domains of the eigenfunction and approximate the clusters of the graphs.

Davies et al. [76] derived the first discrete nodal domain theorem for the $\triangle_2^{(g)}$ operator. Chang et al. [68] and Tudisco et al. [67] generalized these

theorem for $\triangle_1^{(g)}$ and $\triangle_p^{(g)}$ $(p > 1)$ of graphs. In what follows, we prove that the discrete nodal domain theorem applies to $\triangle_p$ of submodular hypergraphs.

As every nodal domain theorem depends on some underlying notion of connectivity, we first define the relevant notion of connectivity for submodular hypergraphs. In a graph or a homogeneous hypergraph, vertices on the same edge or hyperedge are considered to be connected. However, this property does not generalize to submodular hypergraphs, as one can merge two nonoverlapping hyperedges into one without changing the connectivity of the hyperedges. To see why this is the case, consider two hyperedges e_1 and e_2 that are nonintersecting. One may transform the submodular hypergraph so that it includes a hyperedge $e = e_1 \cup e_2$ with weight $w_e = w_{e_1} + w_{e_2}$. This transformation essentially does not change the submodular hypergraph, but in the newly obtained hypergraph, according to the standard definition of connectivity, the vertices in e_1 and e_2 are connected. This problem may be avoided by defining connectivity based on the volume of the boundary set.

Definition 4.2.1. *Two distinct vertices $u, v \in V$ are said to be* connected *if for any S such that $u \in S$ and $v \notin S$, $vol(\partial S) > 0$. A submodular hypergraph is* connected *if for any non-empty $S \subset V$, one has $vol(\partial S) > 0$.*

According to the following lemma, it is always possible to transform the weight functions of submodular hypergraph in such a way as to preserve connectivity.

Lemma 4.2.2. *Any submodular hypergraph $G = (V, E, \mathbf{w}, \boldsymbol{\mu})$ can be reduced to another submodular hypergraph $G' = (V, E', \mathbf{w}', \boldsymbol{\mu})$ without changing $vol(\partial S)$ for any $S \subseteq V$ and ensuring that for any $e \in E'$, and $u, v \in e$, u and v are connected.*

Definition 4.2.3. *Let $x \in \mathbb{R}^N$. A* positive (respectively, negative) strong nodal domain *is the set of vertices of a maximally connected induced subgraph of G such that $\{v \in V | x_v > 0\}$ (respectively, $\{v \in V | x_v < 0\}$). A* positive (respectively, negative) weak nodal domain *is defined in the same manner, except for changing the strict inequalities as $\{v \in V | x_v \geq 0\}$ (*respectively*, $\{v \in V | x_v \leq 0\}$).*

The following lemma establishes that for a connected submodular hypergraph G, all nonconstant eigenvectors of the operator $\triangle_p$ correspond to nonzero eigenvalues.

Lemma 4.2.4. *If G is connected, then all eigenvectors associated with the zero eigenvalue have constant entries.*

We next state new nodal domain theorems for submodular hypergraph $p-$Laplacians. The results imply the bounds for the numbers of nodal domains induced from eigenvectors of p-Laplacian do not essentially change compared to those for graphs [67].

Theorem 4.2.5. *Assume that G is a connected submodular hypergraph. Furthermore, let the eigenvalues of $\triangle_p$ obtained by (4.3) be ordered as $0 = \lambda_1^{(p)} < \lambda_2^{(p)} \leq \cdots \leq \lambda_{k-1}^{(p)} < \lambda_k^{(p)} = \cdots = \lambda_{k+r-1}^{(p)} < \lambda_{k+r}^{(p)} \leq \cdots \leq \lambda_n^{(p)}$, with $\lambda_k^{(p)}$ having topological multiplicity r. Let x be an arbitrary eigenvector associated with $\lambda_k^{(p)}$. Then, when $p > 1$, x induces at most $k + r - 1$ strong and at most k weak nodal domains. When $p = 1$, the number of corresponding weak nodal domains can be greater than k while not greater than the number of corresponding strong nodal domains.*

The next lemma derives a general lower bound on the number of nodal domains of connected submodular hypergraphs.

Lemma 4.2.6. *Let G be a connected submodular hypergraph. For $p > 1$, any nonconstant eigenvector has at least two weak (strong) nodal domains. Hence, the eigenvectors associated with the second smallest eigenvalue $\lambda_2^{(p)}$ have exactly two weak (strong) nodal domains. For $p = 1$, the eigenvectors associated with the second smallest eigenvalue $\lambda_2^{(1)}$ may have only one single weak (strong) nodal domain.*

We define next the following three functions:

$$\mu_p^+(x) \triangleq \sum_{v\in V: x_v>0} \mu_v |x_v|^{p-1},\ \mu^0(x) \triangleq \sum_{v\in V: x_v=0} \mu_v,\ \mu_p^-(x) \triangleq \sum_{v\in V: x_v<0} \mu_v |x_v|^{p-1}.$$

The following lemma characterizes eigenvectors from another perspective that might be useful latter.

Lemma 4.2.7. *Let G be a connected submodular hypergraph. Then, for any nonconstant eigenvector x of $\triangle_p$, one has $\mu_p^+(x) - \mu_p^-(x) = 0$ for $p > 1$, and $|\mu_1^+(x) - \mu_1^-(x)| \leq \mu^0(x)$ for $p = 1$. Consequently, $0 \in \arg\min_{c\in\mathbb{R}} \|x - c\mathbf{1}\|_{\ell_p,\mu}^p$ for any $p \geq 1$.*

The nodal domain theorem characterizes the structure of the eigenvectors of the operator, and the number of nodal domains determines the approximation guarantees in Cheeger-type inequalities relating the spectra of graphs and hypergraphs and the Cheeger constant. These observations are rigorously formalized in the next section.

4.3 Higher-order Cheeger inequalities

In what follows, we analytically characterize the relationship between the Cheeger constants and the eigenvalues $\lambda_k^{(p)}$ of $\triangle_p$ for submodular hypergraphs.

Theorem 4.3.1. *Suppose that $p \geq 1$ and x is an eigenvector of $\triangle_p$ corresponding to the eigenvalue $\lambda_k^{(p)}$, with m_k denoting the number of strong nodal domains of x_k. Then,*

$$\left(\frac{1}{\tau}\right)^{p-1}\left(\frac{h_{m_k}}{p}\right)^{p} \leq \lambda_k^{(p)} \leq (\min\{\zeta(E), k\})^{p-1}\, h_k.$$

For homogeneous hypergraphs, a tighter bound holds that reads as

$$\left(\frac{2}{\tau}\right)^{p-1}\left(\frac{h_{m_k}}{p}\right)^{p} \leq \lambda_k^{(p)} \leq 2^{p-1}\, h_k.$$

It is straightforward to see that setting $p = 1$ produces the tightest bounds on the eigenvalues, while the case $p = 2$ reduces to the classical Cheeger inequality. This motivates an in-depth study of algorithms for evaluating the spectrum of $p = 1, 2$-Laplacians, described next.

4.4 Spectral clustering algorithms based on p-Laplacians

The Cheeger constant is frequently used as an objective function for (balanced) graph and hypergraph partitioning [7, 64, 65, 69, 9, 71]. Theorem 4.3.1 implies that $\lambda_k^{(p)}$ is a good approximation for the k-way Cheeger constant of submodular graphs. Hence, to perform accurate hypergraph clustering, one

has to be able to efficiently learn $\lambda_k^{(p)}$ [77, 78]. We outline next how to do so for $k = 2$.

In Theorem 4.4.1, we describe an objective function that allows us to characterize $\lambda_2^{(p)}$ in a computationally tractable manner; the choice of the objective function is related to the objective developed for graphs in [64, 65]. Minimizing the proposed objective function produces a real-valued output vector $x \in \mathbb{R}^N$. Theorem 4.4.3 describes how to round the vector x and obtain a partition which provably upper bounds $c(S)$. Based on the theorems, we propose two algorithms for evaluating $\lambda_2^{(2)}$ and $\lambda_2^{(1)}$. Since $\lambda_2^{(1)} = h_2$, the corresponding partition corresponds to the tightest approximation of the 2-way Cheeger constant. The eigenvalue $\lambda_2^{(2)}$ can be evaluated in polynomial time with provable performance guarantees. The problem of devising good approximations for values $\lambda_k^{(p)}$, $k \neq 2$, is still open.

Let $Z_{p,\mu}(x, c) \triangleq \|x - c\mathbf{1}\|_{\ell_p,\mu}^p$ and $Z_{p,\mu}(x) \triangleq \min_{c\in\mathbb{R}} Z_{p,\mu}(x, c)$, and define

$$\mathcal{R}_p(x) \triangleq \frac{Q_p(x)}{Z_{p,\mu}(x)}. \tag{4.4}$$

Theorem 4.4.1. *For $p > 1$, $\lambda_2^{(p)} = \inf_{x\in\mathbb{R}^N} \mathcal{R}_p(x)$. Moreover, $\lambda_2^{(1)} = \inf_{x\in\mathbb{R}^N} \mathcal{R}_1(x) = h_2$.*

Definition 4.4.2. *Given a nonconstant vector $x \in \mathbb{R}^N$, and a threshold θ, set $\Theta(x, \theta) = \{v : x_v > \theta\}$. The optimal conductance obtained from thresholding vector x equals*

$$c(x) = \inf_{\theta\in[x_{\min},x_{\max})} \frac{vol(\partial\Theta(x,\theta))}{\min\{vol(\Theta(x,\theta)), vol(V/\Theta(x,\theta))\}}.$$

Theorem 4.4.3. *For any $x \in \mathbb{R}^N$ that satisfies $0 \in \arg\min_c Z_{p,\mu}(x, c)$, i.e., such that $Z_{p,\mu}(x, 0) = Z_{p,\mu}(x)$, one has $c(x) \leq p\,\tau^{(p-1)/p}\,\mathcal{R}_p(x)^{1/p}$, where $\tau = \max_{v\in V} d_v/\mu_v$.*

In what follows, we present two algorithms. The first algorithm describes how to minimize $\mathcal{R}_2(x)$, and hence provides a polynomial-time solution for submodular hypergraph partitioning with provable approximation guarantees, given that the size of the largest hyperedge is a constant. The result is concluded in Theorem 4.4.5. The algorithm is based on an SDP, and may be computationally too intensive for practical applications involving large hypergrpahs of even moderately large hyperedges. The second algorithm is

based on IPM [69] and aims to minimize $\mathcal{R}_1(x)$. Although this algorithm does not come with performance guarantees, it provably converges (see Theorem 4.4.6) and has good heuristic performance. Moreover, the inner loop of the IPM involves solving a version of the proximal-type decomposable submodular minimization problem (see Theorem 4.4.7), which can be efficiently performed using a number of different algorithms [79, 19, 80, 20, 70].

An SDP Method for Minimizing $\mathcal{R}_2(x)$

The $\mathcal{R}_2(x)$ minimization problem introduced in Equation (4.4) may be rewritten as

$$\min_{x: Ux \perp \mathbf{1}} \frac{Q_2(x)}{\|x\|^2_{\ell_2,\mu}}, \tag{4.5}$$

where we observe that $Q_2(x) = \sum_{e\in E} \vartheta_e f_e^2(x) = \sum_{e\in E} \vartheta_e \max_{y\in\mathcal{E}(\mathcal{B}_e)} \langle y, x\rangle^2$. This problem is, in turn, equivalent to the nonconvex optimization problem

$$\begin{aligned} \min_{x\in\mathbb{R}^N} & \sum_e \vartheta_e \left(\max_{y\in\mathcal{E}(\mathcal{B}_e)} \langle y, x\rangle \right)^2 \\ \text{s.t.} & \sum_{v\in V} \mu_v x_v^2 = 1, \ \sum_{v\in V} \mu_v x_v = 0. \end{aligned} \tag{4.6}$$

Following an approach proposed for homogeneous hypergraphs [8], one may try to solve an SDP relaxation of (4.6) instead. To describe the relaxation, let each vertex v of the graph be associated with a vector $x'_v \in \mathbb{R}^n$, $n \geq \zeta(E)$. The assigned vectors are collected into a matrix of the form $X = (x'_1, .., x'_N)$. The SDP relaxation reads as

$$\begin{aligned} \min_{X\in\mathbb{R}^{n\times N},\, \eta\in\mathbb{R}^{|E|}} & \sum_e \vartheta_e \eta_e^2 \\ \text{s.t.} \quad & \|Xy\|_2^2 \leq \eta_e^2 \quad \forall y \in \mathcal{E}(\mathcal{B}_e), e \in E \\ & \sum_{v\in V} \mu_v \|x'_v\|_2^2 = 1, \sum_{v\in V} \mu_v x'_v = 0. \end{aligned} \tag{4.7}$$

Note that $\mathcal{E}(\mathcal{B}_e)$ is of size $O(|e|!)$, and the above problem can be solved efficiently if $\zeta(E)$ is small.

Algorithm 4.1 lists the steps of an SDP-based algorithm for minimizing $\mathcal{R}_2(x)$, and it comes with approximation guarantees stated in Lemma 4.4.4.

In contrast to homogeneous hypergraphs [8], for which the approximation factor equals $O(\log \zeta(E))$, the guarantees for general submodular hypergraphs are $O(\zeta(E))$. This is due to the fact that the underlying base polytope $\mathcal{B}_e$ for a submodular function is significantly more complex than the corresponding polytope for the homogeneous case. We conjecture that this approximation guarantee is optimal for SDP methods.

Algorithm 4.1: Minimization of $\mathcal{R}_2(x)$ using SDP

Input: A submodular hypergraph $G = (V, E, \mathbf{w}, \boldsymbol{\mu})$

1: Solve the SDP (4.7).

2: Generate a random Gaussian vector $g \sim N(0, I_n)$, where I_n denotes the identity matrix of order n.

3: Output $x = X^T g$.

Lemma 4.4.4. *Let x be as in Algorithm 4.1, and let the optimal value of (4.7) be SDPopt. Then, with high probability,*

$$\mathcal{R}_2(x) \leq O(\zeta(E))\, SDPopt \leq O(\zeta(E))\, \min \mathcal{R}_2.$$

This result immediately leads to the following theorem.

Theorem 4.4.5. *Suppose that x is the output of Algorithm 4.1. Then, $c(x) \leq O(\sqrt{\zeta(E)\tau\, h_2})$ with high probability.*

We describe next Algorithm 4.2 for optimizing $\mathcal{R}_1(x)$ which has guaranteed convergence properties.

Algorithm 4.2: IPM-based minimization of $\mathcal{R}_1(x)$

Input: A submodular hypergraph $G = (V, E, \mathbf{w}, \boldsymbol{\mu})$

Find nonconstant $x^0 \in \mathbb{R}^N$ s.t. $0 \in \arg\min_c \|x^0 - c\mathbf{1}\|_{\ell_1,\mu}$

initialize $\hat{\lambda}^0 \leftarrow \mathcal{R}_1(x^0)$, $k \leftarrow 0$

1: **Repeat**:

2: For $v \in V$, $g_v^k \leftarrow \begin{cases} \operatorname{sgn}(x_v^k)\mu_v, & \text{if } x_v^k \neq 0 \\ -\frac{\mu_1^+(x^k) - \mu_1^-(x^k)}{\mu^0(x^k)}\mu_v, & \text{if } x_v^k = 0 \end{cases}$

3: $z^{k+1} \leftarrow \arg\min_{z:\|z\|\leq 1} Q_1(z) - \hat{\lambda}^k \langle z, g^k \rangle$

4: $c^{k+1} \leftarrow \arg\min_c \|z^{k+1} - c\mathbf{1}\|_{\ell_1,\mu}$

5: $x^{k+1} \leftarrow z^{k+1} - c^{k+1}\mathbf{1}$

6: $\hat{\lambda}^{k+1} \leftarrow \mathcal{R}_1(x^{k+1})$

7: Until $|\hat{\lambda}^{k+1} - \hat{\lambda}^k|/\hat{\lambda}^k < \epsilon$

8: **Output** x^{k+1}

Theorem 4.4.6. *The sequence $\{x^k\}$ generated by Algorithm 4.2 satisfies*

$$\mathcal{R}_1(x^{k+1}) \leq \mathcal{R}_1(x^k).$$

The computationally demanding part of Algorithm 4.2 is the optimization procedure in Step 3. The optimization problem is closely related to the problem of submodular function minimization (SFM) due to the defining properties of the Lovász extension. Theorem 4.4.7 describes different equivalent formulations of the optimization problem in Step 3.

Theorem 4.4.7. *If the norm of the vector z in Step 3 is $\|z\|_2$, the underlying optimization problem is the dual of the following ℓ_2 minimization problem:*

$$\min_{y_e} \| \sum_{e \in E} y_e - \hat{\lambda}^k g^k \|_2^2, \quad y_e \in \vartheta_e \mathcal{B}_e, \ \forall\, e \in E, \tag{4.8}$$

where the primal and dual variables are related according to $z = \frac{\hat{\lambda}^k g^k - \sum_{e \in E} y_e}{\|\hat{\lambda}^k g^k - \sum_{e \in E} y_e\|_2}$.

If the norm of the vector z in Step 3 is $\|z\|_\infty$, the underlying optimization problem is equivalent to the following SFM problem:

$$\min_{S \subseteq V} \sum_e \vartheta_e w_e(S) - \hat{\lambda}^k g^k(S), \tag{4.9}$$

where the the primal and dual variables are related according to $z_v = 1$ if $v \in S$, and $z_v = -1$ if $v \notin S$.

For special forms of submodular weights, different algorithms for the optimization problems in Theorem 4.4.7 may be used instead. For graphs and homogeneous hypergraphs with hyperedges of small size, the min-cut algorithms [81, 6] allow one to efficiently solve the discrete problem (4.9). Continuous optimization methods such as alternating projections (AP) [80] and coordinate descent methods (CDM) [20] can be used to solve (4.8) by "tracking" minimum norm points of base polytopes corresponding to individual hyperedges, where for general submodular weights, Wolfe's algorithm [25] can be used. When the submodular weights have some special properties, such as that they depend only on the cardinality of the input, there exist algorithms that operate efficiently even when $|e|$ is extremely large [19].

In our experimental evaluations, we use a random coordinate descent method (RCDM) [20], which ensures an expected $(1+\epsilon)-$approximation by

solving an expected number of $O(|V|^2|E|\log\frac{1}{\epsilon})$ min-norm-point problems. Note that when performing continuous optimization, one does not need to solve the inner-loop optimization problem exactly and is allowed to exit the loop as long as the objective function value decreases. Algorithm 4.3 lists the step of an RCDM algorithm in which one submodular hyperedge is sampled in one iteration, and the corresponding value of y_e is updated. (Clearly, multiple values of y_e can be updated simultaneously if and only if the corresponding hyperedges do not intersect, and this parallelization step further improves the convergence rate of the method.)

Algorithm 4.3: A RCDM for Solving the problem (8)

Input: Submodular hypergraph $G = (V, E, \mathbf{w}, \boldsymbol{\mu})$, $\hat{\lambda}^k$, g^k.
0: Initialize $y_e^0 \in \vartheta_e \mathcal{B}_e$ for $e \in E$, $k \leftarrow 0$
1: In iteration k:
2: Sample one hyperedge $e \in E$ uniformly at random.
3: $y_e^{k+1} \leftarrow \arg\min_{y_e \in \vartheta_e \mathcal{B}_e} \|y_e + \sum_{e' \in E/\{e\}} y_{e'} - \hat{\lambda}^k g^k\|_2^2$
4: Set $y_{e'}^{k+1} \leftarrow y_{e'}^k$ for $e' \neq e$.
Output $\frac{\hat{\lambda}^k g^k - \sum_{e \in E} y_e}{\|\hat{\lambda}^k g^k - \sum_{e \in E} y_e\|_2}$

4.5 Data clustering with large hyperedges

In what follows, we compare the algorithms for submodular hypergraph clustering described in the previous section to two methods: The IPM for homogeneous hypergraph clustering [9] and the clique expansion method (CEM) for submodular hypergraph clustering [71]. We focus on 2-way graph partitioning problems related to the University of California Irvine (UCI) datasets selected for analysis in [9], described in Table 4.1. The datasets include 20Newsgroups, Mushrooms and Covertype. In all datasets, $\zeta(E)$ was roughly 10^3, and each of these datasets describes multiple clusters. Since we are interested in 2-way partitioning, we focused on two pairs of clusters in Covertype, denoted by $(4, 5)$ and $(6, 7)$, and paired the four 20Newsgroups clusters, one of which includes *Comp.* and *Sci*, and another one which includes *Rec.* and *Talk*. The Mushrooms and 20Newsgroups datasets contain only categorical features, while Covertype also includes numerical features. We adopt the same approach as the one described in [9] to construct hyperedges: Each

feature corresponds to one hyperedge; hence, each categorical feature is captured by one hyperedge, while numerical features are first quantized into 10 bins of equal size, and then mapped to hyperedges. To describe the submodular weights, we fix $\vartheta_e = 1$ for all hyperedges and parametrize w_e using a variable $\alpha \in (0, 0.5]$

$$w_e(S;\alpha) = \frac{1}{2} + \frac{1}{2}\min\left\{1, \frac{|S|}{\lceil\alpha|e|\rceil}, \frac{|e/S|}{\lceil\alpha|e|\rceil}\right\},\ \forall S \subseteq e.$$

The intuitive explanation behind our choice of weights is that it allows one to accommodate categorization errors and outliers: In contrast to the homogeneous case in which any partition of a hyperedge has weight one, the chosen submodular weights allow a smaller weight to be used when the hyperedge is partitioned into small parts, i.e., when $\min\{|S|, |e/S|\} < \lceil\alpha|e|\rceil$. In practice, α is chosen to be relatively small – in all experiments, we set $\alpha \leq 0.04$, with α close to zero producing homogeneous hyperedge weights.

Table 4.1: The UCI datasets used for experimental testing.

Dataset	20Newsgroups	Mushroom	Covertype45	Covertype67
$\lvert V\rvert$	16242	8124	12240	37877
$\lvert E\rvert$	100	112	127	136
$\sum_{e\in E}\lvert e\rvert$	65451	170604	145999	451529

The results are shown in Figure 4.1. As may be observed, both in terms of the Clustering error (i.e., the total number of erroneously classified vertices) and the values of the Cheeger constant, IPM-based methods outperform CEM. This is due to the fact that for large hyperedge sizes, CEM incurs a high distortion when approximating the submodular weights ($O(\zeta(E))$ [71]). Moreover, as $w_e(S)$ depends merely on $|S|$, the submodular hypergraph CEM reduces to the homogeneous hypergraph CEM [7], which is an issue that the IPM-based method does not face. Comparing the performance of IPM on submodular hypergraphs (IPM-S) with that on homogeneous hypergraphs (IPM-H), we see that IPM-S achieves better clustering performance on both 20Newsgroups and Covertypes, and offers the same performance as IPM-H on the Mushrooms dataset. This indicates that it is practically useful to use submodular hyperedge weights for clustering purposes. A somewhat unexpected finding is that for certain cases, one observes that when α increases (and thus, when w_e decreases), the corresponding Cheeger constant

increases. This result may be caused by the fact that the IPM algorithm can get trapped in local minima.

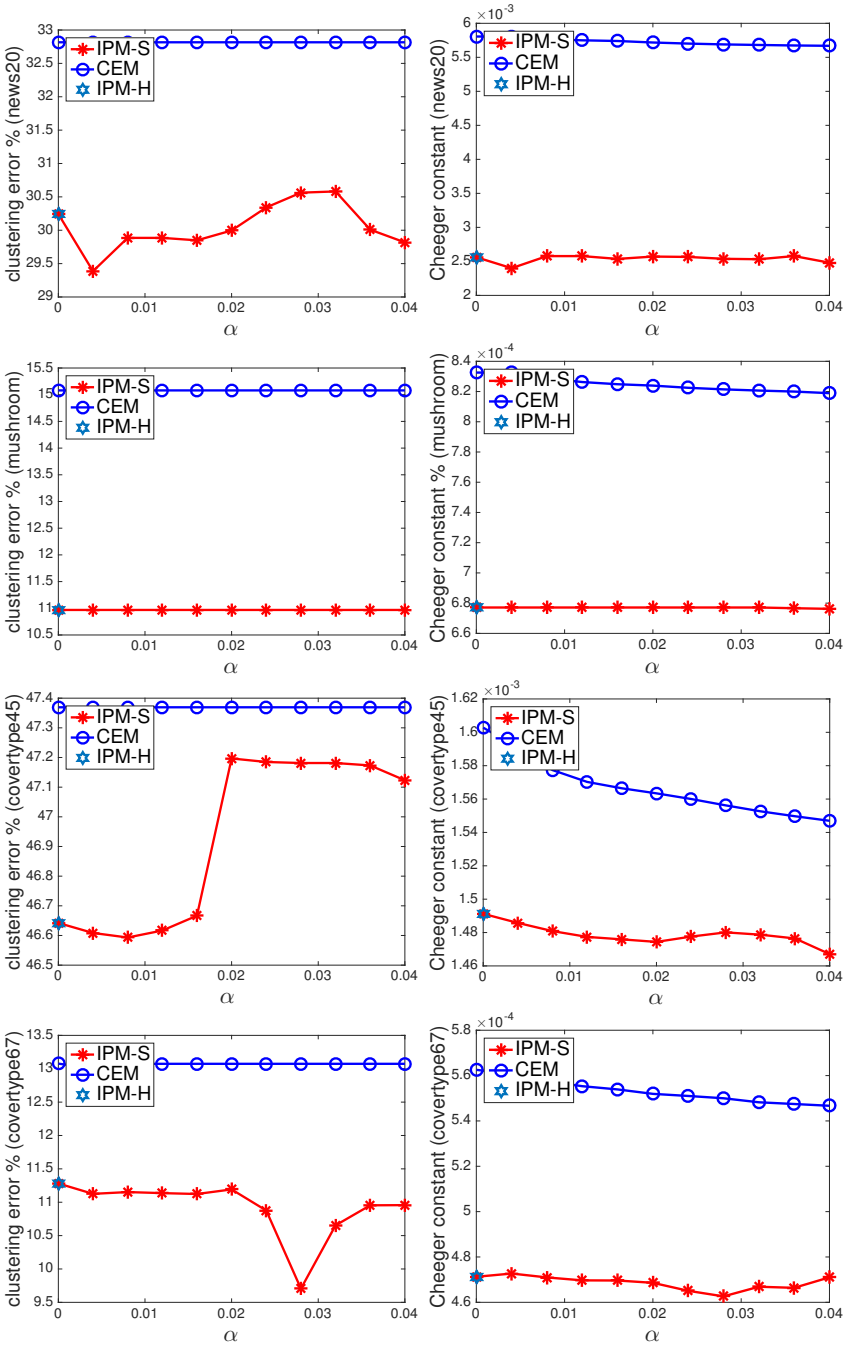


Figure 4.1: Experimental clustering results for four UCI datasets, displayed in pairs of figures depicting the Clustering error and the Cheeger constant versus α. Fine tuning the parameter α may produce significant performance improvements in several datasets. For example, on the Covertype67 dataset, choosing $\alpha = 0.028$ results in visible drops of the clustering error and the Cheeger constant. *Both the use of* 1*-Laplacians and submodular weights may be credited for improving clustering performance.*

CHAPTER 5

DECOMPOSABLE SUBMODULAR FUNCTION MINIMIZATION — MIN-CUTS

In this chapter, we consider another problem defined over submodular hypergraphs: Rather than approximate the submodular hypergraph conductance (3.1), we are to solve a problem by removing the volume of set S used as the denominator in the conductance $c(S)$. The new problem corresponds to the min-cut problem over submodular hypergraphs without normalization, which follows the form $\min_S \sum_{e\in E} w_e(S)$. However, directly solving this problem is trivial because $w_e(\cdot) \geq 0$ makes $S = \emptyset$ a trivial solution. Typically a more meaningful problem is to further fix two sets of vertices such that the solution always puts the two sets in the two different sides of the cut. A more general setting can be used to encode this consideration, where vertices are associated with some unary potential $\boldsymbol{\mu}' : V \to \mathbb{R}$ where μ'_v implies the cost to put a vertex v into the solution set S. Note that this unary potential is different from non-negative unary potential $\boldsymbol{\mu}$ defined in Chapter 2. Then, the more meaningful problem becomes $\min_S \sum_{e\in E} w_e(S) + \sum_{v\in S} \mu'_v$. To view the problem in a more general way, the unary potential, which is essentially a modular function, can be also viewed as a submodular function. Considering the first term is also a collection of submodular functions defined over hyperedges, we may view the problem in a more uniform way: We are to minimize a submodular function $F(\cdot)$ that is defined on the ground set $[N]$. Moreover, we have additional structure of F: Suppose F can be written as the sum of a collection of submodular functions $\{F_r\}_{r\in[R]}$, i.e., $F = \sum_{r\in[R]} F_r$. We term F as a decomposable submodular function and the minimization problem as decomposable submodular function minimization problem (DSFM) that follows:

$$\text{DSFM:} \quad \min_S \sum_{r\in[R]} F_r(S). \tag{5.1}$$

DSFM has attracted much research attentions in the recent decade since

Stobbe and Krause 's first work [18]. There are two important motivations: First, it naturally arises in many applications including image segmentation and circuit segmentation. Second, the generic submodular function minimization algorithms tend to have extremely high polynomial order in their complexity, where the current fastest known SFM algorithm has complexity $O(N^4 \log^{O(1)} N + \tau N^3)$, where τ denotes the time needed to evaluate the submodular function [82]. Since in practice each F_r is much "simpler" than the original F, leveraging such decomposable structures may accelerate the minimization procedure.

Algorithmic solutions for the DSFM problem fall into two categories, combinatorial optimization approaches [79, 83] and continuous function optimization methods [28]. In the latter setting, using the crucial concept the Lovász extension of the submodular function (2.5) lends the DSFM problem to a norm-regularized convex optimization framework. Prior work in continuous DSFM has focused on devising efficient algorithms for solving the convex problem and deriving matching convergence results. The best known approaches include the alternating projection (AP) methods [19, 80] and the coordinate descent (CD) methods [20].

Although these previous works leverage simplifications offered through decomposibility, they still derive convergence guarantees that are suboptimal, because they all miss to leverage the basic fact that each decomposed part F_r may not depend on all the entities in the ground set $[N]$. Inheriting from the terminology of the incidence matrix of a hyperedge to describe the dependence between vertices and hyperedges, we term the dependence between one decomposed part and one entity as *an incidence relation*. It is crucial to utilize *incidence relations* to further accelerate the algorithms for DSFM problems. Often, incidences involve relatively small subsets of elements, which leads to desirable sparsity constraints. This is especially the case for min-cut problems on graphs and hypergraphs (where each submodular component involves two or several vertices) [81, 6] and MAP inference with higher-order potentials (where each submodular component involves variables corresponding to adjacent pixels) [18].

In this chapter, we revisit two benchmark algorithms for continuous DSFM – AP and CD – and describe how to modify them to exploit incidence relations that allow for significantly improved computational complexity. Furthermore, we provide a complete theoretical analysis of the algorithms

parametrized by incidence relations with respect to their convergence rates. AP-based methods that leverage incidence relations achieve better convergence rates than classical AP algorithms both in the sequential and parallel optimization scenario. The random CD method (RCDM) and accelerated CD method (ACDM) that incorporate incidence information can be parallelized. The complexity of sequential CD methods cannot be improved using incidence relations, but the convergence rate of parallel CD methods strongly depends on how the incidence relations are used for coordinate sampling: while a new specialized combinatorial sampling based on equitable coloring [84] is optimal, uniformly at random sampling produces a 2-approximation. It also leads to a greedy method that empirically outperforms random sampling. A summary of these and other findings is presented in Table 5.1.

Table 5.1: Overview of known and new results: each entry contains the required number of iterations to achieve an ϵ-optimal solution (the dependence on ϵ is the same for all algorithms and hence omitted). Here, $\|\mu\|_1 = \sum_{i\in[N]} \mu_i$, where for all $i \in [N]$, μ_i equals the number of submodular functions that involve element i; K is a parallelization parameter that equals the number of min-norm points problems that have to be solved within each iteration.

	Prior work		This work	
	Sequential	Parallel	Sequential	Parallel
AP	$O(N^2R^2)$	$O(\frac{N^2R^2}{K})$	$O(N\|\mu\|_1 R)$	$O(\frac{N\|\mu\|_1 R}{K})$
RCDM	$O(N^2R)$	-	$O(N^2R)$	$O\left(\left(\frac{R-K}{R-1}N^2 + \frac{K-1}{R-1}N\|\mu\|_1\right)\frac{R}{K}\right)$
ACDM	$O(NR)$	-	$O(NR)$	$O\left(\left(\frac{R-K}{R-1}N^2 + \frac{K-1}{R-1}N\|\mu\|_1\right)^{1/2}\frac{R}{K}\right)$

5.1 Background and problem formulation

We start our exposition by reviewing several recent lines of work for solving the DSFM problem, and focus on approaches that transform the DSFM problem into a continuous optimization problem. We let $\mathcal{B}_r$ and f_r denote the base polytope and the Lovász extention (2.5) of F_r. Then, the DSFM problem can be solved through continuous optimization, $\min_{x\in[0,1]^N} \sum_r f_r(x)$. To counter the nonsmoothness of the objective function, a proximal formulation of a generalization of the above optimization problem is considered

instead [19],

$$\min_{x\in\mathbb{R}^N} \sum_{r\in[R]} f_r(x) + \frac{1}{2}\|x\|_2^2. \tag{5.2}$$

As the problem (5.2) is strongly convex, it has a unique optimal solution, denoted by x^*. The exact discrete solution to the DSFM problem equals $S^* = \{i \in [N] |\, x_i^* > 0\}$.

For convenience, we denote the product of base polytopes as $\mathcal{B} = \otimes_{r=1}^R \mathcal{B}_r$, and write $y = (y_1, y_2, ..., y_R) \in \mathcal{B}$. Also, we let A be a simple linear mapping $\otimes_{r=1}^R \mathbb{R}^N \to \mathbb{R}^N$, which given a point $a = (a_1, a_2, ..., a_R) \in \otimes_{r=1}^R \mathbb{R}^N$ outputs $Aa = \sum_{r\in[R]} a_r$. The AP and CD algorithms for solving (5.2) use the dual form of the problem, described in the next lemma.

Lemma 5.1.1 ([19])**.** *The dual problem of* (5.2) *reads as*

$$\min_{a,y} \|a - y\|_2^2 \quad s.t. \quad Aa = 0,\ y \in \mathcal{B}. \tag{5.3}$$

Moreover, problem (5.3) *may be written in the more compact form*

$$\min_{y} \|Ay\|_2^2 \quad s.t. \quad y \in \mathcal{B}. \tag{5.4}$$

For both problems, the primal and dual variables are related according to $x = -Ay$*. In what follows, for notational simplicity, we write* $g(y) = \frac{1}{2}\|Ay\|_2^2$*.*

The AP [80] and RCD algorithms [20] described below provide solutions to the problems (5.3) and (5.4), respectively. They both rely on repeated projections $\Pi_{\mathcal{B}_r}(\cdot)$ onto the base polytopes $\mathcal{B}_r$, $r \in [R]$. These projections are typically less computationally intense than projections onto the complete base polytope of F as they involve fewer data dimensions. The projection operation $\Pi_{\mathcal{B}_r}(\cdot)$ requires one to solve a min-norm problem by either exploiting the special forms of F_r or by using the general purpose algorithm of Wolfe [25]. The complexity of the method is typically characterized by the number of required projections $\Pi_{\mathcal{B}_r}(\cdot)$.

The AP algorithm. Starting with $y = y^{(0)}$, iteratively compute a sequence $(a^{(k)}, y^{(k)})_{k=1,2,...}$ such that for all $r \in [R]$, $a_r^{(k)} = y_r^{(k-1)} - Ay^{(k-1)}/R$, $y_r^{(k)} = \Pi_{\mathcal{B}_r}(a_r^{(k)})$, until a stopping criteria is met.

The RCDM algorithm. In each iteration k, chose uniformly at ran-

dom a subset of elements in y associated with one atomic function in the decomposition (5.1), say the one with index r_k. Then, compute the sequence $(y^{(k)})_{k=1,2,\ldots}$ according to $y^{(k)}_{r_k} = \Pi_{B_{r_k}}\left(-\sum_{r\neq r_k} y^{(k-1)}_r\right)$, $y^{(k)}_r = y^{(k-1)}_r$, for $r \neq r_k$.

Finding an ϵ-optimal solution for both the AP and RCD methods requires $O(N^2 R \log(\frac{1}{\epsilon}))$ iterations. In each iteration, the AP algorithm computes the projections onto all R base polytopes, while the RCDM only computes one projection. Therefore, as may be seen from Table 5.1, the sequential AP solver, which computes one projection in each iteration, requires $O(N^2 R^2 \log(\frac{1}{\epsilon}))$ iterations. However, the projections within one iteration of the AP method can be generated in parallel, while the projections performed in the RCDM have to be generated sequentially.

5.1.1 Incidence relations and related notations

We next formally introduce one of the key concepts used in this work: *incidence relations* between elements of the ground set and the component submodular functions.

We say that an element $i \in [N]$ is *incident* to a submodular function F iff there exists a $S \subseteq [N]/\{i\}$ such that $F(S \cup \{i\}) \neq F(S)$; similarly, we say that the submodular function F is *incident* to an element i iff i is incident to F. To verify whether an element i is incident to a submodular function F, one needs to verify that $F(\{i\}) = 0$ and that $F([N]) = F([N]/\{i\})$ since for any $S \subseteq [N]/\{i\}$

$$F(\{i\}) \geq F(S \cup \{i\}) - F(S) \geq F([N]) - F([N]/\{i\}).$$

Furthermore, note that if $i \in [N]$ is not incident to F_r, then for any $y_r \in \mathcal{B}_r$, one has $y_{r,i} = 0$. Let S_r be the set of all elements incident to F_r. For each element i, denote the number of submodular functions that are incident to i by $\mu_i = |\{r \in [R] : i \in S_r\}|$. We also refer to μ_i as the degree of element i. We find it useful to partition the set of submodular functions into different groups. Given a group $C \subseteq [R]$ of submodular functions, we define the degree of the element i within C, μ^C_i, as $\mu^C_i = |\{r \in C : i \in S_r\}|$.

We also define a skewed norm involving two vectors $w \in \mathbb{R}^N_{>0}$ and $z \in \mathbb{R}^N$ according to $\|z\|_{2,w} \triangleq \sqrt{\sum_{i\in[N]} w_i z_i^2}$. With a slight abuse of notation, for two

vectors $\theta = (\theta_1, \theta_2, ..., \theta_R) \in \otimes_{r=1}^{R} \mathbb{R}^N_{>0}$ and $y \in \otimes_{r=1}^{R} \mathbb{R}^N$, we also define the norm $\|y\|_{2,\theta} \triangleq \sqrt{\sum_{r\in[R]} \|y_r\|^2_{2,\theta_r}}$. Which of the norms we refer to should be clear from the context. In addition, we let $\|\theta\|_{1,\infty} = \sum_{i\in[N]} \max_{r\in[R]:i\in S_r} \theta_{r,i}$. For a closed set $\mathcal{K} \subseteq \otimes_{r=1}^{R} \mathbb{R}^N$ and a positive vector $\theta \in \otimes_{r=1}^{R} \mathbb{R}^N_{>0}$, the distance between y and $\mathcal{K}$ is defined as $d_\theta(y, \mathcal{K}) = \min\{\|y - z\|_{2,\theta} | z \in \mathcal{K}\}$. Also, given a set $\Omega \subseteq \mathbb{R}^N$, we let $\Pi_{\Omega,w}(\cdot)$ denote the projection operation onto Ω with respect to the norm $\|\cdot\|_{2,w}$.

Given a vector $w \in \mathbb{R}^N_{>0}$, we also make use of an induced vector $I(w) \in \otimes_{r=1}^{R} \mathbb{R}^N$ whose r-th entry satisfies $(I(w))_r = w$. It is easy to check that $\|I(w)\|_{1,\infty} = \|w\|_1$. Of special interest are induced vectors based on pairs of N-dimensional vectors, $\mu = (\mu_1, \mu_2, ..., \mu_N)$, $\mu^C = (\mu_1^C, \mu_2^C, ..., \mu_N^C)$. Finally, for $w, w' \in \mathbb{R}^N$, we denote the element-wise power of w by $w^\alpha = (w_1^\alpha, w_2^\alpha, ..., w_N^\alpha)$, for some $\alpha \in \mathbb{R}$, and the element-wise product of w and w' by $w \odot w' = (w_1 w'_1, w_2 w'_2, ..., w_N w'_N)$.

Next, recall that x^* is the unique optimal solution of the problem (5.2) and let $\mathcal{Z} = \{\xi \in \otimes_{r=1}^{R} \mathbb{R}^N | A\xi = -x^*, \xi_{r,i} = 0, \forall i \in S_r, \forall r \in [R]\}$. Then, due to the duality relationship of Lemma 5.1.1, $\Xi = \mathcal{Z} \cap \mathcal{B}$ is the set of optimal solutions $\{y\}$.

5.2 Continuous DSFM algorithms with incidence relations

In what follows, we revisit the AP and CD algorithms and describe how to improve their performance and analytically establish their convergence rates. Our first result introduces a modification of the AP algorithm (5.3) that exploits incidence relations so as to decrease the required number of iterations from $O(N^2R)$ to $O(N\|\mu\|_1)$. Our second result is an example that shows that the convergence rates of CD algorithms [83] cannot be directly improved by exploiting the functions' incidence relations even when the incidence matrix is extremely sparse. Our third result is a new algorithm that relies on coordinate descent steps but can be parallelized. In this setting, incidence relations are essential to the parallelization process.

To analyze solvers for the continuous optimization problem (5.2) that exploit the incidence structure of the functions, we make use of the skewed norm $\|\cdot\|_{2,w}$ with respect to some positve vector w that accounts for the fact

that incidences are, in general, nonuniformly distributed. In this context, the projection $\Pi_{\mathcal{B}_r,w}(\cdot)$ reduces to solving a classical min-norm problem after a simple transformation of the underlying space which does not incur significant complexity overheads. To see this, note that in order to solve a generic min-norm point problem, one typically uses either Wolfe's algorithm (continuous) or a divide-and-conquer procedure (combinatorial). The complexity of the former is at most quadratic in $F_{r,\max} \triangleq \max_{v,S} |F_r(S \cup \{v\}) - F_r(S)|$ [85], while the complexity of the latter merely depends on $\log F_{r,\max}$. This is because of the following Lemma (5.2.1) which describes how the projections $\Pi_{\mathcal{B}_r,w}(\cdot)$ can be performed via discrete optimization. It is unclear if including the weight vector w into the projection procedure increases or decreases $F_{r,\max}$. In either case, given that in our derivations all elements of w are contained in $[1, \max_{i\in[N]} \mu_i]$ instead of N or R, we do not expect to see significant changes in the complexity of the projection operation. Hence, throughout the remainder of our exposition, we regard the projection operation as an oracle and measure the complexity of all algorithms in terms of the number of projections performed.

Lemma 5.2.1. *The optimization problem to compute the project* $\Pi_{\mathcal{B}_r,w}(\cdot)$*, i.e.,* $\min_{y_r\in\mathcal{B}_r} \|z - y_r\|_{2,w}^2$*, is the dual of the problem* $\min_{x\in\mathbb{R}^N} f_r(x) - \langle x, z\rangle + \frac{1}{2}\|x\|_{2,w^{-1}}^2$*. A solution with coordinate accuracy* ϵ *for the latter setting can be obtained by solving the discrete problem*

$$\min_S F_r(S) - z(S) + \lambda \sum_{i\in S_r\cap S} w_i^{-1},$$

where

$$\lambda \in \left[\min_{i\in[N]}[-F_r(\{i\}) + z(\{i\})]w_i,\ \max_{i\in[N]}[F_r([N]/\{i\}) - F_r([N]) + z(\{i\})]w_i\right],$$

at most $\min\{|S_r|, \log 1/\epsilon\}$ *times. The parameter* λ *is chosen based on a binary search procedure which requires solving the discrete problem* $O(\log 1/\epsilon)$ *times.*

Proof. The first statement follows from $f_r(x) = \max_{y_r\in\mathcal{B}_r}\langle y_r, x\rangle$ and some simple algebra. The second claim follows from the divide and conquer algorithm described in Appendix B of [19]. □

Also, observe that one may avoid computing projections in skewed-norm spaces by introducing in (5.2) a weighted rather than an unweighted proximal

term. This gives another continuous objective that still provides a solution to the discrete problem (5.1). Even in this case, we can prove that the numbers of iterations used in the different methods listed Table 5.1 remain the same. Furthermore, by combining projections in skewed-norm spaces and weighted proximal terms, it is possible to actually reduce the number of iterations given in Table 5.1. However, for simplicity, we focus on the objective (5.2) and projections in skewed-norm spaces. Methods using weighted proximal terms with and without skewed-norm projections are analyzed in a similar manner in Section 5.4.

We make frequent use of the following result which generalizes Lemma 4.1 of [83].

Lemma 5.2.2. *Let $\theta \in \otimes_{r=1}^{R}\mathbb{R}^{N}_{>0}, w \in \mathbb{R}^{N}_{>0}$ be two positive vectors. Let $y \in \mathcal{B}$ and let z be in the base polytope of the submodular function F. Then, there exists a point $\xi \in \mathcal{B}$ such that $A\xi = z$ and $\|\xi - y\|_{2,\theta} \leq \sqrt{\frac{\|\theta\|_{1,\infty}}{2}}\|Ay - z\|_1$. Moreover, $\|\xi - y\|_{2,\theta} \leq \sqrt{\frac{\|\theta\|_{1,\infty}\|w^{-1}\|_1}{2}}\|Ay - z\|_{2,w}$.*

5.2.1 The incidence relation AP (IAP)

The following result establishes the basis of our improved AP method leveraging incidence structures.

Lemma 5.2.3. *The following problem is equivalent to problem* (5.3)*:*

$$\min_{a,y} \|a - y\|^2_{2,I(\mu)} \quad s.t. \quad y \in \mathcal{B}, Aa = 0, \text{ and } a_{r,i} = 0, \ \forall (r,i) : i \notin S_r, r \in [R]. \tag{5.5}$$

Let $\mathcal{A} = \{a \in \otimes_{r=1}^{R}\mathbb{R}^N | Aa = 0, a_{r,i} = 0, \ \forall (r,i) : i \notin S_r\}$ and $\mathcal{A}' = \{a \in \otimes_{r=1}^{R}\mathbb{R}^N | Aa = 0\}$. The AP algorithm for problem (5.5) consists of alternatively computing projections between $\mathcal{A}$ and $\mathcal{B}$, as opposed to those between $\mathcal{A}'$ and $\mathcal{B}$ used in the problem (5.3). However, as already pointed out, unlike for the classical AP problem (5.3), the distance in (5.5) is not Euclidean, and hence the projections may not be orthogonal.

The IAP method for solving (5.5) proceeds as follows. We begin with $a = a^{(0)} \in \mathcal{A}$, and iteratively compute a sequence $(a^{(k)}, y^{(k)})_{k=1,2,\ldots}$ as follows: for all $r \in [R]$, $y_r^{(k)} = \Pi_{\mathcal{B}_r,\mu}(a_r^{(k)})$, $a_{r,i}^{(k)} = y_{r,i}^{(k-1)} - \mu_i^{-1}(Ay^{(k-1)})_i, \ \forall \ i \in S_r$.

The key difference between the AP and IAP algorithms is that the latter effectively removes "irrelevant" components of y_r by fixing the irrelevant components of a to 0. In the AP method of Nishihara [80], these components are never zero as they may be "corrupted" by other components during AP iterations. Removing irrelevant components results in projecting y into a subspace of lower dimensions, which significantly accelerates the convergence of IAP (see illustration in Figure. 5.1).

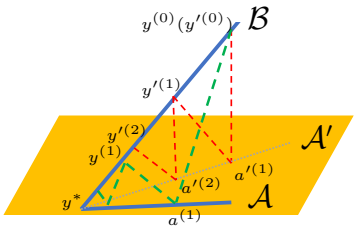


Figure 5.1: Illustration of the IAP method for solving problem (5.5): The space $\mathcal{A}$ is a subspace of $\mathcal{A}'$, which leads to faster convergence of the IAP method when compared to AP.

The analysis of the convergence rate of the IAP method follows a similar outline as that used to analyze (5.3) in [80]. Following Nishihara et al. [80], we define the following parameter that plays a key role in determining the rate of convergence of the AP algorithm:

$$\kappa_* \triangleq \sup_{y \in \mathcal{Z} \cup \mathcal{B}/\Xi} \frac{d_{I(\mu)}(y, \Xi)}{\max\{d_{I(\mu)}(y, \mathcal{Z}), d_{I(\mu)}(y, \mathcal{B})\}}.$$

Lemma 5.2.4 ([80]). *If $\kappa_* < \infty$, the AP algorithm converges linearly with rate $1 - \frac{1}{\kappa_*^2}$. At the k-th iteration, the algorithm outputs a value $y^{(k)}$ that satisfies*

$$d_{I(\mu)}(y^{(k)}, \Xi) \leq 2 d_{I(\mu)}(y^{(0)}, \Xi) \left(1 - \frac{1}{\kappa_*^2}\right)^k.$$

To apply the above lemma in the IAP setting, one first needs to establish an upper bound on κ_*. This bound is given in Lemma 5.2.5 below.

Lemma 5.2.5. *The parameter κ_* is upper bounded as $\kappa_* \leq \sqrt{N\|\mu\|_1/2} + 1$.*

By using the above lemma and the bound on κ_*, one can establish the following convergence rate for the IAP method.

Theorem 5.2.6. *After $O(N\|\mu\|_1 \log(1/\epsilon))$ iterations, the IAP algorithm for solving problem (5.5) outputs a pair of points (a, y) that satisfies $d_{I(\mu)}(y, \Xi) \leq \epsilon$.*

Note that in practice, one often has $\|\mu\|_1 \ll NR$, which shows that the convergence rate of the AP method for solving the DSBM problem may be significantly improved.

5.2.2 Sequential coordinate descent algorithms

Unlike the AP algorithm, the CD algorithms by Ene and Nguyen [20] remain unchanged given (5.4). Our first goal is to establish whether the convergence rate of the CD algorithms can be improved using a parameterization that exploits incidence relations.

The convergence rate of CD algorithms is linear if the objective function is component-wise smooth and ℓ-strong convex. In our case, $g(y)$ is component-wise smooth as for any $y, z \in \mathcal{B}$ that only differ in the r-th block (i.e., $y_r \neq z_r$, $y_{r'} = z_{r'}$ for $r' \neq r$), one has

$$\|\nabla_r g(y) - \nabla_r g(z)\|_2 \leq \|y - z\|_2. \tag{5.6}$$

Here, $\nabla_r g$ denotes the gradient vector associated with the r-th block.

Definition 5.2.7. *We say that the function $g(y)$ is* ℓ-strongly convex *in $\|\cdot\|_2$, if for any $y \in \mathcal{B}$*

$$g(y^*) \geq g(y) + \langle \nabla g(y), y^* - y \rangle + \frac{\ell}{2}\|y^* - y\|_2^2, \tag{5.7}$$
$$\textit{or equivalently,} \quad \|Ay - Ay^*\|_2^2 \geq \ell \|y^* - y\|_2^2,$$

where $y^* = \arg\min_{z \in \Xi} \|z - y\|_2^2$. Moreover, we let

$$\ell_* = \sup\{\ell : g(y) \textit{ is } \ell\textit{-strongly convex in } \|\cdot\|_2\}.$$

Note that the above definition essentially establishes a form of weak-strong convexity [86]. Then, using standard analytical tools for CD algorithms [87], we can prove the following result [20].

Theorem 5.2.8. *The RCDM for problem* (5.4) *outputs a point y that satisfies $\mathbb{E}[g(y)] \leq g(y^*) + \epsilon$ after $O(\frac{R}{\ell_*}\log(1/\epsilon))$ iterations. The ACDM applied to the problem* (5.4) *outputs a point y that satisfies $\mathbb{E}[g(y)] \leq g(y^*) + \epsilon$ after $O(\frac{R}{\sqrt{\ell_*}}\log(1/\epsilon))$ iterations.*

To precisely characterize the convergence rate, we need to find an accurate estimate of ℓ_*. Ene et al. [83] derived $\ell_* \geq \frac{1}{N^2}$ without taking into account the incidence structure. As sparse incidence side information improves the performance of the AP method, it is of interest to determine if the same can be accomplished for the CD algorithms. Example 5.2.1 establishes that this is not possible in general if one only relies on ℓ_*.

Example 5.2.1. *Consider a DSFM problem with a extremely sparse incidence structure with $|S_r| = 2$. More precisely, let $N = 2n + 1$, $R = 2n$, and $\|\mu\|_1 = \sum_{r\in[R]} |S_r| = 4n \ll NR$. Let F_r be incident to the elements $\{r, r+1\}$, for all $r \in [R]$, and be such that $F_r(\{r\}) = F_r(\{r+1\}) = 1, F_r(\emptyset) = F_r(\{r, r+1\}) = 0$. Then, $\ell_* < \frac{7}{N^2}$.*

Note that the optimal solution of problem (5.4) for this particular setting equals $y^* = 0$. Let us consider a point $y \in \mathcal{B}$ specified as follows. First, due to the given incidence relations, the block y_r has two components corresponding to the elements indexed by r and $r + 1$. For any $r \in [R]$,

$$y_{r,r} = -y_{r,r+1} = \begin{cases} \frac{r}{n} & r \leq n, \\ \frac{2n+1-r}{n} & r \geq n+1. \end{cases} \tag{5.8}$$

Therefore, $g(y) = \frac{1}{n}$, $\|y\|_2^2 > \frac{4}{3}n$, which results in $\ell_* < \frac{3}{2n^2} \leq \frac{7}{N^2}$ for all $n \geq 3$.

Example 5.2.1 only illustrates that an important parameter of CDMs cannot be improved using incidence information; but this does not necessarily imply that a sequential RCDM that uses incidence structures cannot offer better convergence rates than $O(N^2R)$. In Figure 5.2, we present additional experimental evidence that supports our observation, using the setting of Example 5.2.1: As the accuracy threshold increases, the slope approaches the value 3, which indicates that the required number of iterations equals $O(N^2R)$.

As a final remark, note that Nishihara et al. [80] also proposed a lower bound that does not make use of sparse incidence structures and only works for the AP method.

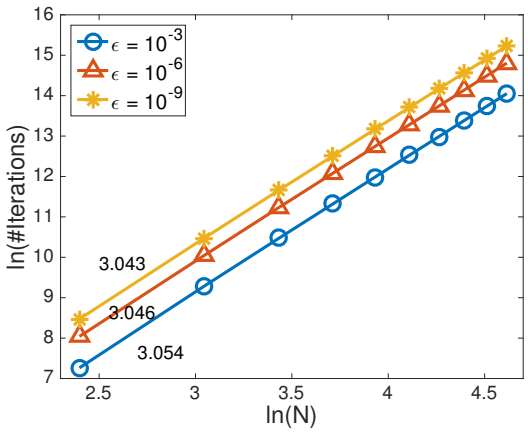


Figure 5.2: Simulations accompanying Example 3.1: ln(the number of iterations) vs ln(N). We constructed a DSFM problem following Example 5.2.1 and initialized y according to equation (5.8). We used the number of iterations k required to attain $g(y^{(k)}) \leq \epsilon g(y^{(0)})$ as a measure for the speed of convergence. We ran the simulations for $n \in [5, 50]$ and averaged the results for each n over 10 independent runs. The values next to the curves are their slopes obtained via a linear regression involving $\ln(\#\text{ Iterations}) \sim \ln(N)$.

5.2.3 New parallel CD methods

In what follows, we propose two CDMs which rely on parallel projections and incidence relations.

The following observation is key to understanding the proposed approach. Suppose that we have a nonempty group of blocks $C \subseteq R$. Let $y, h \in \otimes_{r=1}^{R} \mathbb{R}^N$. If $h_{r,i}$ is nonzero only for block $r \in C$ and $i \in S_r$, then

$$\begin{aligned} g(y+h) &= g(y) + \langle \nabla g(y), h \rangle + \frac{1}{2}\|Ah\|_2^2 \\ &\leq g(y) + \sum_{r \in C} \langle \nabla_r g(y), h_r \rangle + \sum_{r \in C} \frac{1}{2}\|h_r\|_{2,I(\mu^C)}^2. \end{aligned} \tag{5.9}$$

Hence, for all $r \in C$, if we perform projections onto $\mathcal{B}_r$ with respect to the norm $\|\cdot\|_{2,\mu^C}$ simultaneously in each iteration of the CDM, convergence is guaranteed as the value of the objective function remains bounded. The smaller the components of μ^C, the faster the convergence. Note that the components of μ^C are the numbers of incidence relations of elements restricted to the set C. Hence, in each iteration, blocks that ought to be updated in parallel are those that correspond to submodular functions that have supports

with smallest possible intersections.

One can select blocks that are to be updated in parallel in a combinatorially specified fashion or in a randomized fashion, as dictated by what we call an α-proper distribution. To describe our parallel RCDM, we first introduce the notion of an α-proper distribution.

Definition 5.2.9. *Let P be a distribution used to sample a group of C blocks. Define $\theta^P = (\theta_1^P, \theta_2^P, ..., \theta_R^P)$ such that for $r \in [R]$, $\theta_r^P \triangleq \mathbb{E}_{C\sim P}\left[\mu^C | r \in C\right]$. We say that P is an α-proper distribution, if for any $r \in [R]$ and a given $\alpha \in (0,1)$, we have $\mathbb{P}(r \in C) = \alpha$.*

We are now ready to describe the parallel RCDM algorithm – Algorithm 5.1; the description of the parallel ACDM is postponed to Section 5.2.4.

Algorithm 5.1: Parallel RCDM for Solving (5.4)

Input: $\mathcal{B}$, α
0: Initialize $y^{(0)} \in \mathcal{B}$, $k \leftarrow 0$
1: Do the following steps iteratively until the dual gap $< \epsilon$:
2: Sample C_{i_k} using some α-proper distribution P
3: For $r \in C_{i_k}$:
4: $y_r^{(k+1)} \leftarrow \Pi_{\mathcal{B}_r, \theta_r^P}(y_r^{(k)} - (\theta_r^P)^{-1} \odot \nabla_r g(y^{(k)}))$
5: Set $y_r^{(k+1)} \leftarrow y_r^{(k)}$ for $r \notin C_{i_k}$, $k \leftarrow k+1$
6: Output $y^{(k)}$

Next, we establish strong convexity results for the space $\|\cdot\|_{2,\theta^P}$ by invoking Lemma 5.2.2.

Lemma 5.2.10. *For any $y \in \mathcal{B}$, let $y^* = \arg\min_{\xi\in\Xi} \|\xi - y\|_{2,\theta^P}^2$. Then,*

$$\|Ay - Ay^*\|_2^2 \geq \frac{2}{N\|\theta^P\|_{1,\infty}} \|y - y^*\|_{2,\theta^P}^2.$$

The convergence rate of Algorithm 5.1 is established in the next theorem.

Theorem 5.2.11. *At each iteration of Algorithm 5.1, $y^{(k)}$ satisfies*

$$\mathbb{E}\left[g(y^{(k)}) - g(y^*) + \frac{1}{2} d_{\theta^P}^2(y^k, \xi)\right]$$
$$\leq \left[1 - \frac{4\alpha}{(N\|\theta^P\|_{1,\infty} + 2)}\right]^k \left[g(y^{(0)}) - g(y^*) + \frac{1}{2} d_{\theta^P}^2(y^0, \xi)\right].$$

The parameter $N\|\theta^P\|_{1,\infty}$ is obtained by combining the strong convexity constant and the properties of the sampling distribution P. Small values of $\|\theta^P\|_{1,\infty}$ ensure better convergence rates, and we next bound this value.

Lemma 5.2.12. *For any α-proper distribution P and an element $i \in [N]$, $\max_{r\in[R]:i\in S_r} \theta^P_{r,i} \geq \max\{\alpha\mu_i, 1\}$. Consequently, $\|\theta^P\|_{1,\infty} \geq \max\{\alpha\|\mu\|_1, N\}$.*

Without considering incidence relations, i.e., by setting $\|\mu\|_1 = NR$, one always has $\|\theta^P\|_{1,\infty} \geq \alpha NR$, which shows that parallelization cannot improve the convergence rate of the RCDM.

The next lemma characterizes an achievable $\|\theta^P\|_{1,\infty}$ obtained by choosing P to be a uniform distribution, which, when combined with Theorem 5.2.11, proves the result of the last column in Table 5.1.

Lemma 5.2.13. *If C is a set of size $0 < K \leq R$ obtained by sampling the K-subsets of $[R]$ uniformly at random, then $\theta^P_r = \frac{K-1}{R-1}\mu + \frac{R-K}{R-1}1$. Moreover, $\|\theta^P\|_{1,\infty} = \frac{K-1}{R-1}\|\mu\|_1 + \frac{R-K}{R-1}N$.*

Comparing Lemma 5.2.12 and Lemma 5.2.13, we see that the $\|\theta^P\|_{1,\infty}$ achieved by sampling uniformly at random is at most a factor of two of the lower bound since $\alpha = K/R$. A natural question is if it is possible to devise a better sampling strategy. This question is further addressed in Section 5.2.5, where we related the sampling problem to equitable coloring [84]. By using Hajnal-Szemerédi's theorem [88], we derived a sufficient condition under which an α-proper distribution P that achieves the lower bound in Lemma 5.2.12 can be found in polynomial time. We also described a greedy algorithm for minimizing $\|\theta^P\|_{1,\infty}$ that empirically convergences faster than sampling uniformly at random.

5.2.4 A parallel accelerated coordinate descent method

In the ACDM setting, we used the APPROX framework proposed by Fercoq and Richtárik in [89] and adapted it to this particular problem. In the general APPROX framework, the norm in (5.9) is chosen as follows: consider an arbitrary function ϕ with the component-wise smoothness and strong convexity property. For block r, one has $|\nabla_r\phi(x) - \nabla_r\phi(y)| \leq L_r\|x_r - y_r\|_{\nu_r}$, where $\|\cdot\|_{\nu_r}$ is a norm associated with the r-th block. If one wants to process

multiple blocks simultaneously, say those in a group C, one first needs to find a constant L_C such that for any h as defined in (5.9), it holds that

$$\phi(y+h) \;\leq\; \phi(y) + \sum_{r\in C}\langle \nabla_r \phi(y), h_r\rangle + \sum_{r\in C}\frac{L_C}{2}\|h_r\|^2_{2,\nu_r}.$$

The smaller the value of the multiplier L_C, the faster the convergence. Typically, L_C lies in $[\max_{r\in C} L_r, \sum_{r\in C} L_r]$.

Recall the smoothness property of g from equation (5.6). A direct application of APPROX to our problem gives

$$g(y+h) \;\leq\; g(y) + \sum_{r\in C}\langle \nabla_r g(y), h_r\rangle + \sum_{r\in C}\frac{\max_{i\in[N]}\mu_i^C}{2}\|h_r\|^2_2.$$

As $(\max_{i\in[N]}\mu_i^C) \geq \mu_j^C$ for all $j \in [N]$, we obtain convergence rates worse than those implied by inequality (5.9). To actually obtain the guarantees in (5.9), one needs to dispose with the $\|\cdot\|_2$ norm at the block level and break the blocks into components corresponding to the individual elements. The elements are evaluated independently through the use of the norm $\|\cdot\|_{2,\mu^C}$.

Algorithm 5.2: Parallel ACDM for Solving (5.4)

Input: $\mathcal{B}$, α, some constant $c > 0$

0: Initialize $y^{(0)} \in \mathcal{B}$, $k \leftarrow 0$

1: $c' \leftarrow \left\lceil (1+c)\frac{\sqrt{2N\|\theta^P\|_{1,\infty}}}{\alpha} + c \right\rceil$

2: Do the following steps iteratively until the dual gap $< \epsilon$:

3: If $k = lc'$ for some $l \in \mathbb{Z}$, $z^{(k)} \leftarrow y^{(k)}$, $\lambda_k \leftarrow 1$

4: $p^{(k)} \leftarrow (1-\lambda_k)y^{(k)} + \lambda_k z^{(k)}$

5: Sample C_{i_k} using some α-proper distribution P

6: $z^{(k+1)} \leftarrow z^{(k)}$

7: For $r \in C_{i_k}$:

8: $z_r^{(k+1)} \leftarrow \Pi_{\mathcal{B}_r,\theta_r^P}(z_r^{(k)} - \frac{\alpha}{\lambda_k}(\theta_r^P)^{-1} \odot \nabla_r g(p^{(k)}))$

9: $y^{(k+1)} \leftarrow p^{(k)} + \frac{\lambda_k}{\alpha}(z^{(k+1)} - z^{(k)})$

10: $\lambda_{k+1} \leftarrow \frac{\sqrt{\lambda_k^4 + 4\lambda_k^2} - \lambda_k^2}{2}$

11: $k \leftarrow k+1$

12: Output $y^{(k)}$

Similar to the APPROX method [89], the parallel ACDM can also be implemented to avoid full-dimensional vector operations (see Section A.4.9). The following theorem characterizes the convergence property of Algorithm

5.2.

Theorem 5.2.14. *Given $c > 0$, let $c' = \left\lceil (1+c)\frac{\sqrt{2N\|\theta^P\|_{1,\infty}}}{\alpha} + c \right\rceil$. Consider the iterations $k = lc'$ for $l \in \mathbb{Z}_{\geq 0}$. Then, $y^{(k)}$ of Algorithm 5.4 satisfies*

$$\mathbb{E}\left[g(y^{(k)} - g(y^*)\right] \leq \frac{1}{(1+c)^l}\left[g(y^{(0)}) - g(y^*)\right].$$

5.2.5 Minimization of $\|\theta^P\|_{1,\infty}$

Algorithm 5.3: A Greedy Algorithm to minimize $\|\theta^P\|_{1,\infty}$

Input: $\{S_r\}_{r\in[R]}$, K
0: Initialize the partition $\mathcal{C} = \{C_i\}_{1\leq i\leq m}$, $C_i \leftarrow \emptyset$, vectors $\{\mu^{C_i}\}_{1\leq i\leq m}$, $\mu^{C_i} \in \mathbb{R}^N$, and $\mu^{\max} \in \mathbb{R}^N$, $\mu^{\max} \leftarrow 0$.
1: For r from 1 to R:
2: For i from 1 to m:
3: If $|C_i| < K$:
4: $\triangle\mu^{C_i} \leftarrow 0$
5: For v in S_r, if $\mu_v^{C_i}$ is equal to $\mu_v^{\max}$, $\triangle\mu^{C_i} \leftarrow \triangle\mu^{C_i} + 1$
6: else: $\triangle\mu^{C_i} \leftarrow \infty$
7: $i^* \leftarrow \arg\min_i \triangle\mu^{C_i}$
8: $C_{i^*} \leftarrow C_{i^*} \cup \{r\}$
9: For v in S_r, $\mu_v^{C_{i^*}} \leftarrow \mu_v^{C_{i^*}} + 1$, $\mu_v^{\max} \leftarrow \max\{\mu_v^{\max}, \mu_v^{C_{i^*}}\}$.
10: Output $\mathcal{C}$.

We first define $\triangle_* \triangleq \max_{r\in[R]} |\{r' \in [R] | S_{r'} \cap S_r \neq \emptyset\}|$, which we use in our subsequent derivations.

As shown in Theorem 5.2.11 and Theorem 5.2.14, $\|\theta^P\|_{1,\infty}$ plays an important role in the convergence rate of CDMs. Hence, we are interested in identifying the optimal sampling strategy P that minimizes $\|\theta^P\|_{1,\infty}$.

For this purpose, consider a partition of $[R]$ into $m = \lceil \frac{1}{\alpha} \rceil$ parts $\{C_i\}_{1\leq i\leq m}$, such that $|C_i| \in \{K-1, K\}$. We refer to such a partition as a *balanced partition*. In this case, every block r is in exactly one component C_i and $\|\theta^P\|_{1,\infty} = \sum_{v\in[N]} \max_{i\in[m]} \mu_v^{C_i}$. As a result, the problem of minimizing $\|\theta^P\|_{1,\infty}$ is closely related to the so called *equitable coloring problem* first proposed by Meyer [84].

Definition 5.2.15 (Meyer [84])**.** *Given a graph, an* equitable coloring *is an assignment of colors to the vertices that satisfies the following two properties:*

no two adjacent vertices share the same color and the number of vertices in any two color classes differs by at most one. Moreover, the minimum number of colors in any equitable coloring is termed the equitable coloring number.

Hajnal-Szemerédi's theorem [88] established one of the most important results in equitable graph coloring: a graph is equitably k-colorable if k is strictly greater than the maximum vertex degree. This bound is tight. We can construct a graph based on the incidence structure of DSFM problem so that a vertex corresponds to a component submodular function and two vertices are connected iff the corresponding submodular functions are incident to at least one common point. An equitable coloring of this graph can be used to assign submodular functions of the same color class to a set C_i in $\mathcal{C}$. This guarantees that $\mu_v^{C_i} \leq 1$ for all C_i and all $v \in [N]$. Note that the maximal degree of this graph is $\triangle_*$. By directly applying Hajnal-Szemerédi's theorem, we have the following lemma.

Lemma 5.2.16. *There exists a balanced-partition distribution P such that $\|\theta^P\|_{1,\infty} = N$, provided that $\lceil \frac{1}{\alpha} \rceil \geq \triangle_* + 1$.*

As in many applications, such as image segmentation [18], the value of $\triangle_*$ is small, and hence using a balanced-partition instead of one obtained through sampling uniformly at random may produce significantly better results. Unfortunately, finding the equitable coloring number is an NP-hard problem; still, a polynomial time algorithm for finding $\triangle_* + 1$ equitable colorings was described in [90], with complexity $O(\triangle_* R^2)$. We describe a greedy algorithm that outputs a balanced-partition distribution and aims to minimize $\|\theta^P\|_{1,\infty}$ in Algorithm 5.3. According to our experimental results, the sampling strategy P found by Algorithm 5.3 works better than sampling uniformly at random.

5.3 Experiments on images and networks segmentation

In what follows, we illustrate the performance of the newly proposed DSFM algorithms on a benchmark datasets used for MAP inference in image segmentation [18] and used for semi-supervised learning over graphs and hypergraphs.

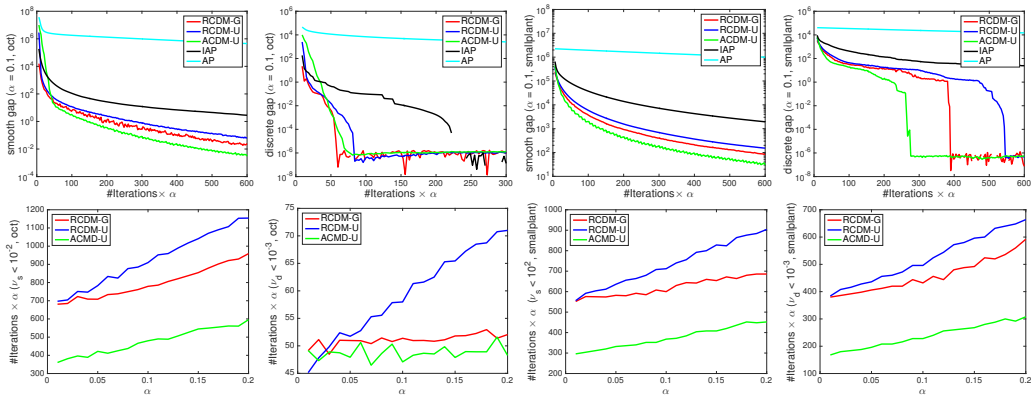


Figure 5.3: Image segmentation example. First row: Gap vs the number of iterations $\times\alpha$. Second row: The number of iterations $\times\alpha$ vs α. Here, α is the parallelization parameter, while $K = \alpha R$ equals the number of projections that have to be computed in each iteration.

In all the experiments, we evaluated the convergence rate of the algorithms by using the smooth duality gap ν_s and the discrete duality gap ν_d. The primal problem solution equals $x = -Ay$ so that the smooth duality gap can be computed according to $\nu_s = \sum_r f_r(x) + \frac{1}{2}\|x\|^2 - (-\frac{1}{2}\|Ay\|^2)$. Moreover, as the level set $S_\lambda = \{v \in [N] | x_v > \lambda\}$ can be easily found based on x, the discrete duality gap can be written as $\nu_d = \min_\lambda F(S_\lambda) - \sum_{v\in[N]} \min\{-x_v, 0\}$.

MAP inference. We used two images – *oct* and *smallplant* – adopted from [19].[1] The images comprise 640×427 pixels so that $N = 273,280$. The decomposable submodular functions are constructed following a standard procedure. The first class of functions arises from the 4-neighbor grid graph over the pixels. Each edge corresponds to a pairwise potential between two adjacent pixels i, j that follows the formula $\exp(-\|v_i - v_j\|_2^2)$, where v_i is the RGB color vector of pixel i. We split the vertical and horizontal edges into rows and columns that result in $639 + 426 = 1065$ components in the decomposition. Note that within each row or each column, the edges have no overlapping pixels, so the projections of these submodular functions onto the base polytopes reduce to projections onto the base polytopes of edge-like submodular functions. The second class of submodular functions contain clique potentials corresponding to the superpixel regions; specifically, for region r, $F_r(S) = |S|(|S_r| - |S|)$ [91]. These functions give another 500 decomposition components. We apply the divide and conquer method in [19] to compute

[1]Downloaded from the website of Professor Stefanie Jegelka: http://people.csail.mit.edu/stefje/code.html

the projections required for this type of submodular functions. Note that in each experiment, all components of the submodular function are of nearly the same size, and thus the projections performed for different components incur similar computational costs. As the projections represent the primary computational units, for comparative purposes we use the number of iterations (similarly to [19, 20]).

We compared five algorithms: RCDM with a sampling distribution P found by the greedy algorithm (RCDM-G), RCDM with uniform sampling (RCDM-U), ACDM with uniform sampling (ACDM-U), AP based on (5.5) (IAP) and AP based on (5.3) (AP). Figure 5.3 depicts the results. In the first row, we compared the convergence rates of different algorithms for a fixed parallelization parameter $\alpha = 0.1$. The values on the horizontal axis correspond to # iterations $\times\, \alpha$, the total number of projections performed divided by R. The results are averaged over 10 independent experiments. We observe that the CD-based methods outperform AP-based methods, and that ACDM-U is the best performing CD-based method. IAP significantly outperforms AP. Similarly, RCDM-G outperforms RCDM-U. We also investigated the relationship between the number of iterations and the parameter α. We recorded the number of iterations needed to achieve a smooth and discrete gap below a certain given threshold. The results are shown in the second row of Figure 5.3. We did not plot the curves for the AP-based methods as they are essentially horizontal lines. Among the CD-based methods, ACDM-U performs best. RCDM-G offers a much better convergence rate than RCDM-U since the sampling probability P produced by the greedy algorithm leads to a smaller value of $\|\theta^P\|_{1,\infty}$ compared to uniform sampling. The reason behind this finding is that the supports of the components in the decomposition are localized, which makes the sampling P obtained from the greedy algorithm highly effective. For RCDM-U, the total number of iterations increases almost linearly with α $(= K/R)$, which confirms the results of Lemma 5.2.13.

Note that in the above examples of MAP inference, another way to decompose the submodular functions is available: as there are three natural layers of non-overlapping incidence sets, we can merge all vertical edges, all horizontal edges, and all superpixel regions into three components respectively. Then, each of this component is incident to all pixels, and the derived results in this work will reduce to those of the former works [19, 20]. However, such

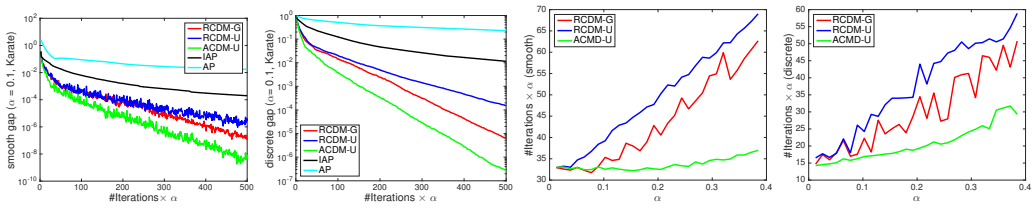


Figure 5.4: Zachary's Karate Club. Left two: Gap vs the number of iterations $\times\alpha$. Right two: The number of iterations $\times\alpha$ vs α. Here, α is the parallelization parameter, while $K = \alpha R$ equals the number of projections that have to be computed in each iteration.

a way to decompose submodular function strongly depends on the particular structure and thus is not general for DSFM problems. The following example on semi-supervised learning over graphs does not contain natural layers for decomposition.

Semi-supervised learning. We tested our algorithms over the dataset of Zachary's karate club [92]. This dataset is used as a benchmark example for evaluating semisupervised learning algorithms over graphs [93]. It includes $N = 34$ vertices and $R = 78$ submodular functions in the decomposition, each corresponding to one edge in the network. The objective function of both semi-supervised learning problems may be written as

$$\min_x \tau \sum_{r\in[R]} f_r(x) + \frac{1}{2}\|x - x_0\|_2^2, \tag{5.10}$$

where τ is a parameter that needs to be tuned, and $x_0 \in \{-1, 0, 1\}^N$, so that the nonzero components correspond to the labels that are known a priori. In our case, as we are only concerned with the convergence rate of the algorithm, we fix $\tau = 0.1$. In the experiments for Zachary's karate club, we set $x_0(1) = 1$, $x_0(34) = -1$ and let all other components of x_0 be equal to zero.

Figure 5.4 shows the results of the experiments pertaining to Zachary's karate club. In the left two subfigures, we compared the convergence rates of different algorithms for a fixed parallelization parameter $\alpha = 0.1$. The values on the horizontal axis correspond to # iterations $\times\,\alpha$, the total number of projections performed divided by R. In the right two subfigures, we controlled the numbers of projections executed within one iteration by tuning the parameter α and recorded the number of iterations needed to achieve smooth/discrete gaps below 10^{-3}. The values depicted on the vertical axis

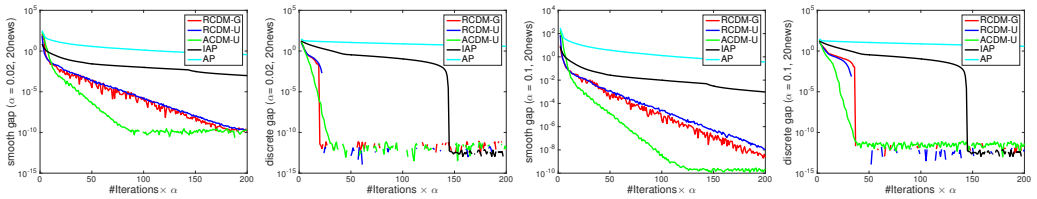


Figure 5.5: 20Newsgrounp: Smooth/discrete gap vs the (number of iterations $\times\alpha$).

correspond to # iterations $\times\alpha$, describing the total number of projections needed to achieve the given accuracy. In all cases, we see a similar tendency to that of the MAP inference. As may be seen, AP-based methods require more projections than CD-based methods, but IAP consistently outperforms AP, which is consistent with our theoretical results. Among the CD-based methods, ACDM-U offers the best performance in general, and RCDM-G slightly outperforms RCDM-U, since the greedy algorithm used for sampling produces a smaller $\|\theta^P\|_{1,\infty}$ than uniform sampling. As the AP-based methods are completely parallelizable, increasing the parameter α does not increase the total number of projections. However, for RCDM-U, the total number of iterations required increases almost linearly with α, which is supported by the result in Lemma 3.12. The performance curve for RCDM-G exhibits large oscillations due to the discrete problem component, needed for finding a balanced partition.

We also evaluate the proposed approaches over the 20Newsgroups from the University of California Irvine (UCI) data repository. This dataset is used as a benchmark example for evaluating semisupervised learning algorithms over hypergraphs [9, 94]. Here, for simplicity, we focused on binary classification tasks and thus paired the four 20Newsgroups classes, so that one group includes "Comp." and "Sci", and the other one includes "Rec." and "Talk". The 20Newsgroups dataset consists of categorical features and we adopt the same approach as the one described in [9] to construct hyperedges: each feature corresponds to one hyperedge and contributes one submodular function to the decomposition. Hence, 20Newsgroups contains $N = 16242$ elements and $R = 100$ submodular functions.

In the experiments for 20Newsgroups, we uniformly at random picked 200 elements and set their corresponding components in x_0 of equation (5.10) to the true labels and set all other entries to zero. Figure 5.5 shows the

results of the experiments pertaining to 20Newsgroup. We compared the convergence rate of different algorithms for different values of the parameter $\alpha \in \{0.02, 0.1\}$. The value on the horizontal axis, # iterations $\times\alpha$, equals the total number of projections, scaled by R. The results are averaged over 10 independent experiments. Once again, we observe that CD-based methods outperform AP-based methods. ACDM-U offers the best performance among all CD-based methods and IAP significantly outperforms AP. Similarly, RCDM-G has better performance than RCDM-U, due to the use of the greedy algorithm for the sampling procedure.

5.4 Using weighted proximal terms for acceleration

The AP and RCDM solvers discussed in the main text are designed to solve the convex optimization (5.2), but also produce a solution to the discrete optimization problem (5.1). To solve the discrete optimization problem (5.1), another convex optimization formulation may be considered instead:

$$\min_{x\in\mathbb{R}^N} \sum_{r\in[R]} f_r(x) + \frac{1}{2}\|x\|^2_{2,w}, \tag{5.11}$$

where the choice of $w \in \mathbb{R}^N_{>0}$ will be described later. By using the arguments in [95] or in Chapter 8.1-8.2 of [28], we know that the solution of the discrete optimization problem (5.1) can be obtained as $S = \{i \in [N] | x^*_i > 0\}$, where x^* is a solution of (5.11).

Next, we describe how a proper choice of w allows one to avoid compute oblique projections in the AP and parallel CDM algorithms. If oblique projections are allowed, a good choice for w may also decrease the computational complexities listed in Table 5.1. The results obtained based on weighted proximal terms are summarized in Table 5.2.

We now analyze the new objective (5.11) in more detail. The proof techniques used in the main text carry over to the setting involving weighted proximal terms.

By using a dual strategy similar to those described in Lemma 5.1.1 and Lemma 5.2.3, we arrive at the dual formulation of problem (5.11) described in the next lemma. Note that the derivation of (5.12) takes into account the underlying incidence relations.

Table 5.2: New complexity results based on weighted proximal terms: here, complexity refers to the required number of iterations needed to achieve an $\epsilon-$optimal solution (the dependence on ϵ is the same for all algorithms and hence omitted). As before, K is the parallelization parameter and it equals the number of min-norm points problems that are solved within each iteration; $K = 1$ reduces to the sequential case.

	Using Orthogonal Projection $\Pi_{\mathcal{B}_r}(\cdot)$	
	The Value of w	Complexity
AP	$w = \mu$	$O(N\|\mu\|_1 \frac{R}{K})$
RCDM	$w = \frac{R-K}{R-1}1 + \frac{K-1}{R-1}\mu$	$O\left(\left(\frac{R-K}{R-1}N^2 + \frac{K-1}{R-1}N\|\mu\|_1\right)\frac{R}{K}\right)$
ACDM	$w = \frac{R-K}{R-1}1 + \frac{K-1}{R-1}\mu$	$O\left(\left(\frac{R-K}{R-1}N^2 + \frac{K-1}{R-1}N\|\mu\|_1\right)^{\frac{1}{2}}\frac{R}{K}\right)$
	Using Oblique Projection $\Pi_{\mathcal{B}_r,w^{1/2}}(\cdot)$	
	The Value of w	Complexity
AP	$w = \mu^{\frac{1}{2}}$	$O(\|\mu^{\frac{1}{2}}\|_1^2 \frac{R}{K})$
RCDM	$w = \left(\frac{R-K}{R-1}1 + \frac{K-1}{R-1}\mu\right)^{\frac{1}{2}}$	$O\left(\left\|\left(\frac{R-K}{R-1}1 + \frac{K-1}{R-1}\mu\right)^{\frac{1}{2}}\right\|_1^2 \frac{R}{K}\right)$
ACDM	$w = \left(\frac{R-K}{R-1}1 + \frac{K-1}{R-1}\mu\right)^{\frac{1}{2}}$	$O\left(\left\|\left(\frac{R-K}{R-1}1 + \frac{K-1}{R-1}\mu\right)^{\frac{1}{2}}\right\|_1 \frac{R}{K}\right)$

Lemma 5.4.1. *The dual problem of* (5.11) *reads as*

$$\min_{a,y} \|a - y\|^2_{2,I(w^{-1}\odot\mu)} \quad s.t. \quad y \in \mathcal{B}, Aa = 0,$$
$$and\ a_{r,i} = 0,\ \forall (r,i) : i \notin S_r, r \in [R]. \tag{5.12}$$

Moreover, problem (5.3) *may be written in a more compact form as*

$$\min_{y} \|Ay\|^2_{2,w^{-1}} \quad s.t. \quad y \in \mathcal{B}. \tag{5.13}$$

For both problems, the primal and dual variables are related according to $x = -w^{-1} \odot Ay$.

5.4.1 The incidence relations AP (IAP) method for solving (5.12)

The steps of the IAP method are listed in Algorithm 5.4.

The convergence properties of Algorithm 5.4 can be characterized similarly as those of IAP for solving (5.5). The latter relies on a finite upper bound for $\kappa_* \triangleq \sup_{y \in \mathcal{Z}\cup\mathcal{B}/\Xi} \frac{d_{I(w^{-1}\odot\mu)}(y,\Xi)}{\max\{d_{I(w^{-1}\odot\mu)}(y,\mathcal{Z}), d_{I(w^{-1}\odot\mu)}(y,\mathcal{B})\}}$.

Algorithm 5.4: The IAP Method for Solving (5.12)

0: For all r, initialize $y_r^{(0)} \in \mathcal{B}_r$, and $k \leftarrow 0$
1: In iteration k:
2: For all $r \in [R]$:
3: $a_{r,i}^{(k+1)} \leftarrow y_{r,i}^{(k)} - \mu_i^{-1}(Ay^{(k)})_i$ for all $i \in S_r$
4: $y_r^{(k+1)} \leftarrow \Pi_{\mathcal{B}_r, w^{-1}\odot\mu}(a_r^{(k+1)})$

Lemma 5.4.2. *One has* $\kappa_* \leq \sqrt{\frac{\|w^{-1}\odot\mu\|_1\|w\|_1}{2}} + 1$. *When* $w = \mu$, $\kappa_* \leq \sqrt{\frac{N\|\mu\|_1}{2}} + 1$.

Proof. The result follows using the same strategy as the one used to prove Lemma 5.2.5. Note that when using Lemma 5.2.2, one should set θ to $I(w^{-1}\odot\mu)$ and replace w by w^{-1}. □

By setting $w = \mu$, Step 4 of Algorithm 5.4 reduces to orthogonal projections. In this case, based on Lemma 5.4.2, Algorithm 5.4 requires $O(N\|\mu\|_1 \log\frac{1}{\epsilon})$ iterations to achieve an ϵ-optimal solution. By setting $w = \mu^{\frac{1}{2}}$ for all $i \in [R]$, Step 4 of Algorithm 5.4 reduces to the projections $\Pi_{\mathcal{B}_r, w^{\frac{1}{2}}}(\cdot)$. In this case, Algorithm 5.4 requires $O\left(\left\|\mu^{\frac{1}{2}}\right\|^2 \log\frac{1}{\epsilon}\right)$ iterations to achieve an ϵ-optimal solution. The latter result is slightly better because $\left\|\mu^{\frac{1}{2}}\right\|^2 \leq N\|\mu\|_1$.

5.4.2 A parallel RCD method for solving (5.13)

As discussed in Section 5.2.3, RCDM strongly depends on an α-proper distribution P that characterizes the parallel coordinate sampling strategy. In what follows, we choose P to be a uniform distribution. From Lemma 5.2.13, we know that when P is uniform, one has $\theta_r^P = \frac{K-1}{R-1}\mu + \frac{R-K}{R-1}1$ for all $r \in [R]$, where K denotes the number of projections computed in parallel as part of each iteration. In Algorithm 5.2, θ_r^P defines the normed space over which to minimize $g(y)$. As our goal is to minimize $g_w(y) = \frac{1}{2}\|Ay\|_{2,w^{-1}}^2$, the vector used to define the normed space is

$$\nu = w^{-1} \odot \theta_r^P = w^{-1} \odot (\frac{K-1}{R-1}\mu + \frac{R-K}{R-1}1).$$

The parallel RCDM procedure in this setting is described in Algorithm 5.5.

Algorithm 5.5: Parallel RCDM for Solving (5.13)

Input: $\mathcal{B}$, K

0: Initialize $y^{(0)} \in \mathcal{B}$, $k \leftarrow 0$

1: Do the following steps iteratively until the dual gap $< \epsilon$:

2: Uniformly sample $C_{i_k} \subseteq [R]$ so that $|C_{i_k}| = K$.

3: For $r \in C_{i_k}$:

4: $y_r^{(k+1)} \leftarrow \Pi_{\mathcal{B}_r,\nu}(y_r^{(k)} - (\nu^{-1}) \odot \nabla_r g_w(y)\ )$

5: Set $y_r^{(k+1)} \leftarrow y_r^{(k)}$ for $r \notin C_{i_k}$

6: $k \leftarrow k + 1$

7: Output $y^{(k)}$

Similarly to what was done in Lemma 5.2.10, we can establish weak strong convexity of $g_w(y)$ with respect to the norm $\|\cdot\|_{2,\nu}$ by invoking Lemma 5.2.2.

Lemma 5.4.3. *For any $y \in \mathcal{B}$, let $y^* = \arg\min_{\xi\in\Xi} \|\xi - y\|_{2,\nu}^2$. Then,*

$$\|Ay - Ay^*\|_{2,w^{-1}}^2 \geq \frac{2}{\|w\|_1\|\nu\|_1}\|y - y^*\|_{2,\nu}^2.$$

Therefore, using a strategy similar to the one outlined in the proof of Theorem 5.2.11, the convergence rates of Algorithm 5.5 can be derived as summarized in the next theorem.

Theorem 5.4.4. *At each iteration of Algorithm 5.5, $y^{(k)}$ satisfies*

$$\mathbb{E}\left[g_w(y^{(k)}) - g_w(y^*) + \frac{1}{2}d_{I(\nu)}^2(y^k, \xi)\right] \leq \left[1 - \frac{4K}{R(\|w\|_1\|\nu\|_1 + 2)}\right]^k \left[g_w(y^{(0)}) - g_w(y^*) + \frac{1}{2}d_{I(\nu)}^2(y^0, \xi)\right].$$

By setting $w = \frac{K-1}{R-1}\mu + \frac{R-K}{R-1}1$, we reduce the projections in Step 4 of Algorithm 5.5 to orthogonal projections. In this case, based on Theorem 5.4.4, Algorithm 5.5 requires $O\left(\left(\frac{K-1}{R-1}N\|\mu\|_1 + \frac{R-K}{R-1}N^2\right)\frac{R}{K}\log\frac{1}{\epsilon}\right)$ iterations to achieve an ϵ-optimal solution.

By setting $w = \left(\frac{K-1}{R-1}\mu + \frac{R-K}{R-1}\right)^{1/2}$ for all $i \in [R]$, the projections in Step 4 of Algorithm 5.5 reduce to oblique projections $\Pi_{\mathcal{B}_r,w^{\frac{1}{2}}}(\cdot)$. In this case, Algorithm 5.5 requires $O\left(\left\|\left(\frac{K-1}{R-1}\mu_i + \frac{R-K}{R-1}\right)^{1/2}\right\|_1^2 \log\frac{1}{\epsilon}\right)$ iterations to achieve an ϵ-optimal solution, which is slightly better than the previous case. The accelerated methods can be analyzed in the same manner.

5.4.3 Simulations

We now describe simulation results that empirically evaluate Algorithms 5.4 and 5.5. The DSFM problem is designed as follows. We consider $N = 100$ vertices. The unary potentials of different elements are iid standard Gaussian variables. We construct a network over these vertices based on the Barabási-Albert model (BA) [96], initialized with a single edge between vertices 1 and 2. Each edge in the network gives a pairwise potential for the corresponding vertices. We use the BA model so that the number of incidence relations corresponding to different vertices varies to a large extent. As we are using weighted proximal terms, the continuous objectives are not consistent for different w. However, here, we are only interested in generating solutions for the discrete problem (5.1) and thus regard the discrete gap ν_d as the relevant metric for characterizing convergence properties. The following results are obtained from 100 independent experiments.

In IAP (Algorithm 5.4), we set $w \in \{1, \mu, \mu^{1/2}\}$, corresponding to three cases: *unweighted proximal term + oblique projections*, *weighted proximal term + orthogonal projections*, *weighted proximal term + oblique projections*, respectively. In RCDM-U (Algorithm 5.5), we set $w \in \{1, \frac{K-1}{R-1}\mu + \frac{R-K}{R-1}1, (\frac{K-1}{R-1}\mu + \frac{R-K}{R-1}1)^{1/2}\}$, corresponding to the same three cases. We control the number of parallel projection operations in each iteration by choosing $K \in \{10, 20, 30, 40, 50\}$. Figure 5.6 shows the convergence curve of the discrete gap for different solvers and different choices of w. We only plotted results for $K = 10, 50$ as other values of K produce similar patterns. For both IAP and RCDM-U, when w corresponds to the weighted proximal term + orthogonal projections case, we obtain the best convergence rates. The value $w = 1$, corresponding to the case unweighted proximal term + oblique projections, results in the worst convergence rates. Albeit somewhat inconsistent with the results listed in Table 5.2, the simulations simply imply that using weighted proximal terms can reduce the complexity of the algorithms at hand and that the weighted proximal term with orthogonal projections in the inner loop may represent the best choice in practice.

In Table 5.3, we also list the number of iterations needed by different solvers to obtain a solution for the discrete problem (5.1). Again, the w corresponding to the weighted proximal term + orthogonal projections case results in the smallest number of iterations, while the w corresponding to the

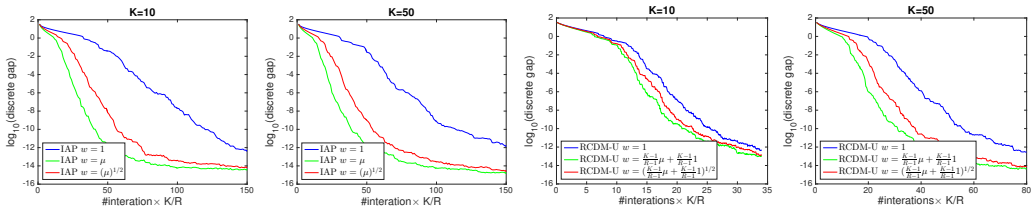


Figure 5.6: Simulations for Algorithm 5.4 and 5.5: $\log_{10}$ (discrete gap) vs (number of iterations $\times K/R$).

Table 5.3: The number of iterations $\times K/R$ needed to find an optimal solution to the discrete problem (5.1). MN: mean; MD: median.

Solvers	w	$K=10$		$K=20$		$K=30$		$K=40$		$K=50$	
		MN	MD	MN	MD	MN	MD	MN	MD	MN	MD
IAP	1	109	103	109	103	109	103	109	103	109	103
	μ	43	34	43	34	43	34	43	34	43	34
	$\mu^{1/2}$	59	50	59	50	59	50	59	50	59	50
RCDM-U	1	27	22	34	28	43	38	51	46	54	49
	$\frac{K-1}{R-1}\mu + \frac{R-K}{R-1}1$	22	17	25	20	29	24	32	24	33	25
	$\left(\frac{K-1}{R-1}\mu + \frac{R-K}{R-1}1\right)^{1/2}$	25	19	28	23	33	28	37	31	38	32

unweighted proximal term + oblique projections case results in the largest number of iterations. Note that as K increases, the number of iterations $\times K/R$ in IAP does not change as IAP is fully parallelizable, while the number of operations in RCDM-U slightly increases due to the overlapping incidence sets of different submodular functions.

CHAPTER 6

QUADRATIC DECOMPOSABLE SUBMODULAR FUNCTION MINIMIZATION — PAGERANKS

In the previous chapter, we considered how to efficiently solve the min-cut problems over submodular hypergraphs. Although the min-cut type of solutions have been found very successful in many applications such as image segmentation, they have some drawbacks. The combinatorial solutions are discrete, which cannot yield soft values that are necessary in many other real life applications like information retrieval and recommendation algorithm design. These applications need soft values to rank the entities in the pool and send ranking lists as reponse. Actually, more than the combinatorial solutions, the solutions in continuous space, i.e., x in (5.2), tend to be discrete in their values because the regularizer based on Lovász extensions impose the sparsity within the difference of values of x. Such drawbacks motivate the work in this chapter, which considers computing the PageRank type of solutions that still capture topology induced by submodular hypergraphs while they may be suitable for ranking-related tasks. Recall that PageRank vectors in the graph case can be obtained by solving an optimization problem with quadratic terms as regularizers. We propose a new convex optimization problem that uses quadratic terms of Lovász extensions as regularizers and give it the name *quadratic decomposable submodular function minimization* (QDSFM).

To specify the QDSFM problem, consider that we still have a submodular hypergraph whose hyperedges are associated with a collection of submodular functions $\{F_r\}_{r\in[R]}$ defined over the ground set $[N]$, and denote their Lovász extensions and base polytopes by $\{f_r\}_{r\in[R]}$ and $\{B_r\}_{r\in[R]}$, respectively. Let $S_r \subseteq [N]$ denote the set of variables incident to F_r and assume that the functions F_r are normalized and nonnegative, i.e., that $F_r(\emptyset) = 0$ and $F_r \geq 0$. These two mild constraints are satisfied by almost all submodular functions that arise in practice.

The QDSFM problem may be formally stated as follows:

$$\text{QDSFM:} \qquad \min_{x\in\mathbb{R}^N} \|x-a\|_W^2 + \sum_{r\in[R]} \left[f_r(x)\right]^2, \tag{6.1}$$

where $a \in \mathbb{R}^N$ is a given vector and $W \in \mathbb{R}^{N\times N}$ is a positive diagonal matrix. Note that the QDSFM problem is convex because the Lovász extensions f_r are convex and nonnegative. In fact, it is also strongly convex, implying that there exists a unique optimal solution, denoted by x^*.

Recall in the previous chapter, we introduced the continuous version of DSFM problems (5.2). The key difference between QDSFM and DSFM is that the regularizers of QDSFM are quadratic forms of Lovász extensions. QDSFM appears naturally in a wide spectrum of applications, including learning on graphs and hypergraphs, and in particular, semi-supervised learning and PageRank analysis, to be detailed later. Moreover, it has been demonstrated both theoretically [97] and empirically [98, 9] that using quadratic regularizers in (6.1) offers significantly improved predictive performance for semi-supervised learning when compared to DSFM. Despite the importance of the QDSFM problem, it has not received the same level of attention as the DSFM problem, both from the theoretic and algorithm perspective. To the best of our knowledge, only a few reported works [9, 99] have provided solutions for *specific instances* of QDSFMs with sublinear convergence guarantees. For the general QDSFM problem, no analytical results are currently known.

This chapter takes a substantial step towards solving the QDSFM problem in its most general form by developing a family of algorithms with *linear convergence rates* and *small iteration cost*, including the randomized coordinate descent (RCD) and alternative projection (AP) algorithms. Our contributions are as follows.

1. First, we derive a new dual formulation for the QDSFM problem since an analogue of the dual transformation for the DSFM problem is not applicable. Interestingly, the dual QDSFM problem requires one to find the best approximation of a hyperplane via a product cone as opposed to a product polytope, encountered in the dual DSFM problem.

2. Second, we develop linearly convergent RCD and AP algorithms for solving the dual QDSFM. Because of the special underlying conic struc-

ture, a new analytic approach is needed to prove the weak strong-convexity of the dual QDSFM, which essentially guarantees linear convergence.

3. Third, we develop generalized Frank-Wolfe (FW) and min-norm-point methods for efficiently computing the conic projection required in each step of RCD and AP and provide a $1/k$-rate convergence analysis. These FW-type algorithms and their corresponding analysis do not rely on the submodularity assumption and apply to general conic projections, and are hence of independent interest.

4. Forth, we introduce a novel application of QDSFM problems, in terms of a PageRank (PR) process for hypergraphs. The underlying PR state vectors can be efficiently computed by solving QDSFM problems. We also show that many important properties of the PR process for graphs carry over to the hypergraph case, including mixing and local hypergraph partitioning results.

5. Finally, we evaluate our methods on semi-supervised learning over hypergraphs using synthetic and real datasets, and demonstrate superior performance both in convergence rate and prediction accuracy compared to existing general-purpose methods.

6.1 Dual formulation

Despite being convex, the objective is in general nondifferentiable. This implies that only sublinear convergence can be obtained when directly applying the subgradient method. To address this issue, we revert to the dual formulation of the QDSFM problem. A natural idea is to mimic the approach used for DSFM by exploiting the characterization of the Lovász extension, $f_r(x) = \max_{y_r \in B_r} \langle y_r, x \rangle$, $\forall r$. However, this leads to a semidefinite programming problem for the dual variables $\{y_r\}_{r\in[R]}$, which is too expensive to solve for large problems. Instead, we establish a new dual formulation that overcomes this obstacle.

The dual formulation hinges upon the following simple, yet key observa-

tion:

$$[f_r(x)]^2 = \max_{\phi_r \geq 0} \phi_r f_r(x) - \frac{\phi_r^2}{4} = \max_{\phi_r \geq 0} \max_{y_r \in \phi_r B_r} \langle y_r, x \rangle - \frac{\phi_r^2}{4}. \tag{6.2}$$

Let $y = (y_1, y_2, ..., y_R)$ and $\phi = (\phi_1, \phi_2, ..., \phi_R)$. For each r, we define a convex cone

$$C_r = \{(y_r, \phi_r) | \phi_r \geq 0, y_r \in \phi_r B_r\},$$

which represents the feasible set of the variables (y_r, ϕ_r). Furthermore, we denote the product cone by

$$\mathcal{C} = \bigotimes_{r \in [R]} C_r := \{(y, \phi) : \phi_r \geq 0, y_r \in \phi_r B_r, \forall r \in [R]\}.$$

Invoking equation (6.2), we arrive at the following two dual formulations for the original QDSFM problem in Lemma (6.1.1).

Lemma 6.1.1. *The following optimization problem is dual to* (6.1)*:*

$$\min_{y,\phi} \; g(y, \phi) := \| \sum_{r \in [R]} y_r - 2Wa \|_{W^{-1}}^2 + \sum_{r \in [R]} \phi_r^2, \; s.t. \; (y, \phi) \in \mathcal{C}. \tag{6.3}$$

By introducing $\Lambda = (\lambda_r) \in \bigotimes_{r \in [R]} \mathbb{R}^N$*, the previous optimization problem takes the form*

$$\min_{y,\phi,\Lambda} \sum_{r \in [R]} \left[\| y_r - \frac{\lambda_r}{\sqrt{R}} \|_{W^{-1}}^2 + \phi_r^2 \right], \; s.t. \; (y, \phi) \in \mathcal{C}, \sum_{r \in [R]} \lambda_r = 2Wa. \tag{6.4}$$

The primal variables in both cases may be computed as $x = a - \frac{1}{2} W^{-1} \sum_{r \in [R]} y_r$.

The proof of Lemma 6.1.1 is presented in Section A.5.1. The dual formulations for the DSFM problem were described in Lemma 2 of [19]. Similarly to the DSFM problem, the dual formulations (6.3) and (6.4) are simple, nearly separable in the dual variables, and may be solved by algorithms such as those used for solving dual DSFM, including the Douglas-Rachford splitting method (DR) [19], the alternative projection method (AP) [80] and the random coordinate descent method (RCD) [20].

However, there are also some notable differences. Our dual formulations for the QDSFM problem are defined on a product cone constructed from the base polytopes of the submodular functions. The optimization problem (6.4)

essentially asks for the best approximation of an affine space in terms of the product cone, whereas in the DSFM problem one seeks an approximation in terms of a product polytope. In the next section, we propose to solve the dual problem (6.3) using the random coordinate descent method (RCD), and to solve (6.4) using the alternative projection method (AP), both tailored to the conic structure. The conic structures make the convergence analysis and the computations of projections for these algorithms challenging.

Before proceeding to the algorithms, we first provide a relevant geometric property of the product cone $\bigotimes_{r\in[R]} C_r$, characterized in the following lemma. This property is essential in establishing the linear convergence of RCD and AP algorithms.

Lemma 6.1.2. *Consider a feasible solution of the problem $(y,\phi) \in \bigotimes_{r\in[R]} C_r$ and a nonnegative vector $\phi' = (\phi'_r) \in \bigotimes_{r\in[R]} \mathbb{R}_{\geq 0}$. Let s be an arbitrary point in the base polytope of $\sum_{r\in[R]} \phi'_r F_r$, and let $W^{(1)}, W^{(2)}$ be two positive diagonal matrices. Then, there exists a $y' \in \bigotimes_{r\in[R]} \phi'_r B_r$ satisfying $\sum_{r\in[R]} y'_r = s$ and*

$$\|y-y'\|^2_{I(W^{(1)})} + \|\phi-\phi'\|^2 \leq \mu(W^{(1)},W^{(2)}) \left[\|\sum_{r\in[R]} y_r - s\|^2_{W^{(2)}} + \|\phi-\phi'\|^2 \right],$$

where

$$\mu(W^{(1)},W^{(2)}) = \max\left\{ \sum_{i\in[N]} W^{(1)}_{ii} \sum_{j\in[N]} 1/W^{(2)}_{jj},\ \frac{9}{4}\rho^2 \sum_{i\in[N]} W^{(1)}_{ii} + 1 \right\}, \quad (6.5)$$

and $\rho = \max_{y_r\in B_r, \forall r\in[R]} \sqrt{\sum_{r\in[R]} \|y_r\|_1^2}$.

The proof of Lemma 6.1.2 is relegated to Section A.5.1.

6.2 Linearly convergent algorithms for solving the QDSFM problem

Next, we introduce and analyze the random coordinate descent method (RCD) for solving the dual problem (6.3), and the alternative projection method (AP) for solving (6.4). Both methods exploit the separable structure of the feasible set. It is worth mentioning that our results may be easily extended to apply to the Douglas-Rachford splitting method, as well as to

Algorithm 6.1: The RCD method for Solving (6.3)

0: For all r, initialize $y_r^{(0)} \leftarrow 0$, $\phi_r^{(0)}$ and $k \leftarrow 0$
1: In iteration k:
2: Uniformly at random pick an $r \in [R]$.
3: $(y_r^{(k+1)}, \phi_r^{(k+1)}) \leftarrow \Pi_{C_r}(2Wa - \sum_{r' \neq r} y_{r'}^{(k)}; W^{-1})$
4: Set $y_{r'}^{(k+1)} \leftarrow y_{r'}^{(k)}$ for $r' \neq r$

accelerated and parallel variants of the RCD method which were used to solve DSFM problems [19, 20, 70].

We define the projection Π onto a convex cone C_r as follows: for a point b in $\mathbb{R}^N$ and a positive diagonal matrix W' in $\mathbb{R}^{N \times N}$, we set

$$\Pi_{C_r}(b; W') = \arg \min_{(y_r, \phi_r) \in C_r} \|y_r - b\|_{W'}^2 + \phi_r^2.$$

Throughout the remainder of this section, we treat the projections as provided by an oracle. Later in Section 6.3, we provide some efficient methods for computing the projections.

6.2.1 The randomized coordinate descent algorithm

Consider the dual formulation (6.3). For each coordinate r, optimizing over the dual variables (y_r, ϕ_r) is equivalent to computing a projection onto the cone C_r. This gives rise to the RCD method of Algorithm 6.1.

The objective $g(y, \phi)$ described in (6.3) is not strongly convex in general. However, with some additional work, Lemma 6.1.2 can be used to establish the weak strong convexity of $g(y, \phi)$; this essentially guarantees a linear convergence rate of the RCD algorithm. To proceed, we need some additional notation. Denote the set of solutions of (6.3) by

$$\Xi = \{(y,\, \phi) | \sum_{r \in [R]} y_r = 2W(a - x^*), \phi_r = \inf_{y_r \in \theta B_r} \theta, \forall r\}. \tag{6.6}$$

This representation is a consequence of the relationship between the optimal primal and dual solutions stated in Lemma 6.1.1. For convenience, we denote the optimal value of the objective function of interest over $(y,\, \phi) \in \Xi$ by

$g^* = g(y, \phi)$, and also define a distance function

$$d((y, \phi), \Xi) = \sqrt{\min_{(y', \phi') \in \Xi} \|y - y'\|^2_{I(W^{-1})} + \|\phi - \phi'\|^2}.$$

Lemma 6.2.1 (Weak Strong Convexity)**.** *Suppose that* $(y, \phi) \in \bigotimes_{r \in [R]} C_r$ *and that* $(y^*, \phi^*) \in \Xi$ *minimizes* $\|y - y'\|^2_{I(W^{-1})} + \|\phi - \phi'\|^2$ *over* $(y', \phi') \in \Xi$*, i.e.,* (y^*, ϕ^*) *is the projection of* (y, ϕ) *onto* Ξ*. Then,*

$$\| \sum_{r \in [R]} (y_r - y_r^*) \|^2_{W^{-1}} + \|\phi - \phi^*\|^2 \geq \frac{d^2((y, \phi), \Xi)}{\mu(W^{-1}, W^{-1})}. \tag{6.7}$$

Lemma 6.2.1 can be proved by applying Lemma 6.1.2 with $\phi' = \phi^*, s = 2W(a - x^*)$, $W^{(1)}, W^{(2)} = W^{-1}$ and the definition of y^*. Note that the claim in (6.7) is significantly weaker than the usual strong convexity condition. We show that such a condition is sufficient to ensure linear convergence of the RCD algorithm and provides the results in the following theorem.

Theorem 6.2.2 (Linear Convergence of RCD)**.** *Running* k *iterations of Algorithm 6.1 produces a pair* $(y^{(k)}, \phi^{(k)})$ *that satisfies*

$$\begin{aligned} &\mathbb{E}\left[g(y^{(k)}, \phi^{(k)}) - g^* + d^2((y^{(k)}, \phi^{(k)}), \Xi)\right] \\ &\leq \left[1 - \frac{2}{R[1 + \mu(W^{-1}, W^{-1})]}\right]^k \left[g(y^{(0)}, \phi^{(0)}) - g^* + d^2((y^{(0)}, r^{(0)}), \Xi)\right]. \end{aligned}$$

Theorem 6.2.2 asserts that at most $O(R\mu(W^{-1}, W^{-1}) \log \frac{1}{\epsilon})$ iterations are required to obtain an ϵ-optimal solution in terms of expectation for the QDSFM problem. The proofs of Lemma 6.2.1 and Theorem 6.2.2 are postponed to Section A.5.2.

6.2.2 The alternative projection algorithm

The AP method can be used to solve the dual problem (6.4), which is of the form of a *best-approximation problem*, by alternatively performing projections between the product cone and a hyperplane. Furthermore, for some incidence relations, S_r may be a proper subset of $[N]$, which consequently requires the ith component of y_r, i.e., $y_{r,i}$ to be zero if $i \notin S_r$. Enforcing $\lambda_{r,i} = 0$ for $i \notin S_r$ allows the AP method to avoid redundant computations and have

Algorithm 6.2: The AP Method for Solving (6.8)

0: For all r, initialize $y_r^{(0)} \leftarrow 0, \phi_r^{(0)} \leftarrow 0$, and $k \leftarrow 0$
1: In iteration k:
2: $\quad \alpha^{(k+1)} \leftarrow 2W^{-1} \sum_r y_r^{(k)} - 4a$.
3: $\quad$ For all $r \in [R]$:
4: $\quad\quad \lambda_{r,i}^{(k+1)} \leftarrow y_{r,i}^{(k)} - \frac{1}{2}(\Psi^{-1} W \alpha^{(k+1)})_i$ for $i \in S_r$
5: $\quad\quad (y_r^{(k+1)}, \phi_r^{(k+1)}) \leftarrow \Pi_{C_r}(\lambda_r^{(k+1)}; \Psi W^{-1})$

better convergence rates. This phenomenon has also been observed for the DSFM problem in [70]. To avoid redundant computations, we use the AP approach to solve the following dual problem:

$$\begin{aligned} &\min_{y,\phi,\Lambda} \quad \sum_{r\in[R]} \left[\|y_r - \lambda_r\|^2_{\Psi W^{-1}} + \phi_r^2 \right], \\ &\text{s.t. } (y,\phi) \in \mathcal{C}, \sum_{r\in[R]} \lambda_r = 2Wa, \text{ and } \lambda_{r,i} = 0 \text{ for all } i \notin S_r. \end{aligned} \tag{6.8}$$

Here, $\Psi \in \mathbb{R}^{N\times N}$ is a positive diagonal matrix in which $\Psi_{ii} = |\{r \in [R] | i \in S_r\}|$ equals the number of submodular functions that i is incident to.

Lemma 6.2.3. *Problem* (6.8) *is equivalent to problem* (6.3).

The proof of Lemma 6.2.3 is presented in Section A.5.3. The AP method for solving (6.8) is listed in Algorithm 6.2. Observe that Step 5 is a projection onto cones defined based on the positive diagonal matrix equal to ΨW^{-1} which differs from the one used in the RCD method. Note that compared to the RCD algorithm, AP requires one to compute projections onto *all* cones C_r in each iteration. Thus, in most cases, AP has larger computation cost than the RCD algorithm. On the other hand, AP naturally lends itself to parallelization since the projections can be decoupled and implemented very efficiently.

Before going on, we first prove the uniqueness of optimal ϕ in the following lemma.

Lemma 6.2.4. *The optimal value of ϕ to problem* (6.8) *is unique.*

Proof. We know (6.8) is equivalent to (6.3). Suppose there are two optimal solutions $(\bar{y}, \bar{\phi})$ and $(\tilde{y}, \tilde{\phi})$. As the optimal x^* is unique, $\sum_{r\in[R]} \bar{y}_r = \sum_{r\in[R]} \tilde{y}_r = 2W(a - x^*) = \sum_{r\in[R]} \frac{\bar{y}_r + \tilde{y}_r}{2}$. If $\bar{\phi} \neq \tilde{\phi}$, we have $\sum_{r\in[R]} \left(\frac{\bar{\phi}_r + \tilde{\phi}_r}{2}\right)^2 <$

$\sum_{r\in[R]} \bar{\phi}_r^2 = \sum_{r\in[R]} \tilde{\phi}_r^2$, which makes $g(\frac{\bar{y}+\tilde{y}}{2}, \frac{\bar{\phi}+\tilde{\phi}}{2})$ smaller than $g(\bar{y}, \bar{\phi}) = g(\tilde{y}, \tilde{\phi})$ and thus causes contradiction. □

To determine the convergence rate of the AP method, we adapt the result of [80] on the convergence rate of APs between two convex bodies. In our setting, the two convex bodies of interest are the the cone $\mathcal{C}$ and the hyperplane

$$\mathcal{Z} = \{(y, \phi) | \sum_{r\in[R]} y_r = 2W(a - x^*), \phi_r = \phi_r^*, y_{r,i} = 0, \forall\, i \notin S_r\},$$

where $\phi^* = (\phi_r^*)_{r\in[R]}$ is the unique optimal solution of (6.8).

Lemma 6.2.5 ([80])**.** *Let Ξ be as defined in* (6.6)*. In addition, define the distance function*

$$d_{\Psi W^{-1}}((y, \phi), \Xi) = \sqrt{\min_{(y',\phi')\in\Xi} \|y - y'\|^2_{I(\Psi W^{-1})} + \|\phi - \phi'\|^2}.$$

In the k-th iteration of Algorithm 6.5, the pair $(y^{(k)}, \phi^{(k)})$ satisfies

$$d_{\Psi W^{-1}}((y^{(k)}, \phi^{(k)}), \Xi) \leq 2d_{\Psi W^{-1}}((y^{(0)}, \phi^{(0)}), \Xi)(1 - \frac{1}{\kappa_*^2})^k,$$

where

$$\kappa_* = \sup_{(y,\phi)\in\mathcal{C}\cup\mathcal{Z}/\Xi} \frac{d_{\Psi W^{-1}}((y, \phi), \Xi)}{\max\{d_{\Psi W^{-1}}((y, \phi), \mathcal{C}), d_{\Psi W^{-1}}((y, \phi), \mathcal{Z})\}}.$$

The next Lemma establishes a finite upper bound on κ_*. This guaratees linear convergence rates for the AP algorithm.

Lemma 6.2.6. *One has $\kappa_*^2 \leq 1 + \mu(\Psi W^{-1}, W^{-1})$.*

The proof of Lemma 6.2.6 may be found in Section A.5.3.

Lemma 6.2.6 implies running the AP algorithm with $O(\mu(\Psi W^{-1}, W^{-1}) \log \frac{1}{\epsilon})$ iterations guarantees that $d_{\Psi W^{-1}}((y^{(k)}, \phi^{(k)}), \Xi) \leq \epsilon$. Moreover, as $\Psi_{ii} \leq R$, the iteration complexity of the AP method is smaller than that of the RCD algorithm. However, each iteration of the AP solver requires performing projections on all cones C_r, $r \in [R]$, while each iteration of the RCD method requires computing only one projection onto a single cone. Other methods used as generic QDSFM solvers in semi-supervised learning on hypergraphs [9]

such as the DR and primal-dual hybrid gradient descent (PDHG) [100], also require computing a total of R projections during each iteration. Thus, from the perspective of iteration cost, RCD is significantly more efficient, especially when R is large and computing $\Pi(\cdot)$ is costly. This phenomenon is also observed in practice, as illustrated by the experiments in Section 6.5.1.

The following corollary summarizes the previous discussion and will be used in our subsequent derivations in Section 6.4.1.

Corollary 6.2.7. *Suppose that $W = \beta D$, where β is a hyper-parameter, and D is a diagonal matrix such that*

$$D_{ii} = \sum_{r \in [R]:\, i \in S_r} \max_{S \subseteq V} [F_r(S)]^2.$$

Recall that $\Psi_{ii} = |\{r \in [R] | i \in S_r\}|$. Then, Algorithm 6.1 (RCD) requires an expected number of

$$O(N^2 R \max\{1, 9\beta^{-1}\} \max_{i,j \in [N]} \frac{D_{ii}}{D_{jj}} \log \frac{1}{\epsilon})$$

iterations to return a solution (y, ϕ) that satisfies $d((y, \phi), \Xi) \leq \epsilon$. Algorithm 6.2 (AP) requires a number of

$$O(N^2 R \max\{1, 9\beta^{-1}\} \max_{i,j \in [N]} \frac{\Psi_{jj}}{R} \frac{D_{ii}}{D_{jj}} \log \frac{1}{\epsilon})$$

iterations to return a solution (y, ϕ) that satisfies $d_{\Psi W^{-1}}((y, \phi), \Xi) \leq \epsilon$.

The proof of Corollary 6.2.7 is provided in Section A.5.4. Note that the term $N^2 R$ also appears in the expression for the complexity of the RCD method and the AP method for solving the DSFM problem [83]. The term $\max\{1, 9\beta^{-1}\}$ implies that whenever β is small, the convergence rate is slow. In Section 6.4.1, we provide a case-specific analysis showing that β is related to the mixing time of and underlying Markov process in the PageRank application. Depending on the application domain, other interpretations of β exist, but will not be discussed here to avoid topical clutter. Finally, we point out that the term $\max_{i,j \in [N]} \frac{D_{ii}}{D_{jj}}$ in the expression for the RCD method and the term $\max_{i,j \in [N]} \frac{\Psi_{jj}}{R} \frac{D_{ii}}{D_{jj}}$ for the AP method arise from degree-based normalization.

6.3 Computing the conic projections

In this section, we describe efficient routines for computing a projection onto the conic set $\Pi_{C_r}(\cdot)$. As the procedure works for all values of $r \in [R]$, we drop the subscript r for simplicity of notation. Let $C = \{(y, \phi) | y \in \phi B, \phi \geq 0\}$, where B denotes the base polytope of the submodular function F. Recall that the conic projection onto C is defined as

$$\Pi_C(a; \tilde{W}) = \arg\min_{(y,\phi)} h(y, \phi) \triangleq \|y - a\|^2_{\tilde{W}} + \phi^2 \quad \text{s.t. } (y, \phi) \in C. \tag{6.9}$$

Let h^* and (y^*, ϕ^*) be the optimal value of the objective function and the optimal solution, respectively. When performing projections, one only needs to consider the variables incident to F, and set all other variables to zero. For simplicity, we assume that all variables in $[N]$ are incident to F. The results can easily extend to the case that the incidences are of a general form.

Unlike the QDSFM problem, solving the DSFM involves the computation of projections onto the base polytopes of submodular functions. Two algorithms, the Frank-Wolfe (FW) method [24] and the Fujishige-Wolfe minimum norm algorithm (MNP) [101], are used for this purpose. Both methods assume inexpensive linear minimization oracles on polytopes and guarantee a $1/k$-convergence rate. The MNP algorithm is more sophisticated yet empirically more efficient. Nonetheless, neither of these methods can be applied directly to conic projections. To this end, we modify these two methods by adjusting them to the conic structure in (6.9) and show that they still achieve a $1/k$-convergence rate. We refer to the procedures as *the conic FW method* and *the conic MNP method*, respectively.

6.3.1 The conic MNP algorithm

Detailed steps of the conic MNP method are given in Algorithm 6.3. The conic MNP algorithm keeps track of an *active set* $S = \{q_1, q_2, ...\}$ and searches for the best solution in its conic hull. Denote the cone of an active set S by $\text{cone}(S) = \{\sum_{q_i \in S} \alpha_i q_i | \alpha_i \geq 0\}$ and its linear set by $\text{lin}(S) = \{\sum_{q_i \in S} \alpha_i q_i | \alpha_i \in \mathbb{R}\}$. Similar to the original MNP algorithm, Algorithm 6.3 also includes two level-loops: the MAJOR and MINOR loop. In the MAJOR loop, one greedily adds a new active point $q^{(k)}$ to the set S obtained from the linear

Algorithm 6.3: The Conic MNP Method for Solving (6.9)

Input: $\tilde{W}$, a, B and a small positive constant δ.
Maintain $\phi^{(k)} = \sum_{q_i \in S^{(k)}} \lambda_i^{(k)}$
Choose an arbitrary $q_1 \in B$.
Set $S^{(0)} \leftarrow \{q_1\}$, $\lambda_1^{(0)} \leftarrow \frac{\langle a, q_1 \rangle_{\tilde{W}}}{1+\|q_1\|^2_{\tilde{W}}}$, $y^{(0)} \leftarrow \lambda_1^{(0)} q_1$, $k \leftarrow 0$
1. Iteratively execute (**MAJOR LOOP**):
2. $q^{(k)} \leftarrow \arg\min_{q \in B} \langle \nabla_y h(y^{(k)}, \phi^{(k)}), q \rangle_{\tilde{W}}$
3. **If** $\langle y^{(k)} - a, q^{(k)} \rangle_{\tilde{W}} + \phi^{(k)} \geq -\delta$, **break**;
4. **Else** $S^{(k)} \leftarrow S^{(k)} \cup \{q^{(k)}\}$.
5. Iteratively execute (**MINOR LOOP**):
6. $\alpha \leftarrow \arg\min_{\alpha} \| \sum_{q_i^{(k)} \in S^{(k)}} \alpha_i q_i^{(k)} - a \|^2_{\tilde{W}} + (\sum_{q_i^{(k)} \in S} \alpha_i)^2$,
7. $z^{(k)} \leftarrow \sum_{q_i^{(k)} \in S} \alpha_i q_i^{(k)}$
8. **If** $\alpha_i \geq 0$ for all i, **break**;
9. **Else** $\theta = \min_{i:\alpha_i<0} \lambda_i^{(k)} / (\lambda_i^{(k)} - \alpha_i)$, $\lambda_i^{(k+1)} \leftarrow \theta \alpha_i + (1-\theta)\lambda_i^{(k)}$,
10. $y^{(k+1)} \leftarrow \theta z^{(k)} + (1-\theta) y^{(k)}$,
11. $S^{(k+1)} \leftarrow \{i : \lambda^{(k+1)} > 0\}$, $k \leftarrow k+1$
12. $y^{(k+1)} \leftarrow z^{(k)}$, $\lambda^{(k+1)} \leftarrow \alpha$, $S^{(k+1)} \leftarrow \{i : \lambda^{(k+1)} > 0\}$, $k \leftarrow k+1$

minimization oracle w.r.t. the base polytope (Step 2). By the end of the MAJOR loop, we obtain $y^{(k+1)}$ that minimizes $h(y, \phi)$ over cone(S) (Step 3-12). The MINOR loop is activated when lin(S) contains some point z that guarantees a smaller value of the objective function than the optimal point in *cone*(S), provided that some active points from S may be removed. It is important to point out that compared to the original MNP method, Steps 2 and 6 as well as the termination Step 3 are specialized for the conic structure.

Compared to the linearly convergent Away-steps Frank-Wolfe algorithm [102], the size of the active set S in the (conic) MNP algorithm is always bounded by R rather than the number of extreme points of the polytope. Furthermore, compared to fully corrective non-negative matching pursuit [103], the (conic) MNP method does not require solving an inner loop nonnegative quadratic program. In practice, the MNP method exhibits empirical performance superior to that of alternative Frank-Wolfe variants; corresponding convergence rates have been recently established in [85].

Theorem 6.3.1 below shows that the conic MNP algorithm preserves the $1/k$-convergence rate of the original MNP method.

Theorem 6.3.1. *Let B be an arbitrary polytope in $\mathbb{R}^N$ and let $C = \{(y, \phi) | y \in \phi B, \phi \geq 0\}$ be the cone induced by the polytope. For some positive diagonal*

Algorithm 6.4: The Conic FW Algorithm for Solving (6.9)

Input: $\tilde{W}$, a, B and a small positive δ

Initialize $y^{(0)} \leftarrow 0$, $\phi^{(0)} \leftarrow 0$ and $k \leftarrow 0$

1. **Iteratively execute the following steps:**
2. $\quad q^{(k)} \leftarrow \arg\min_{q \in B} \langle \nabla_y h(y^{(k)}, \phi^{(k)}), q \rangle$
3. $\quad$ **If** $\langle y^{(k)} - a, q^{(k)} \rangle_{\tilde{W}} + \phi^{(k)} \geq -\delta$, **break**.
4. $\quad$ **Else**: $(\gamma_1^{(k)}, \gamma_2^{(k)}) \leftarrow \arg\min_{\gamma_1 \geq 0, \gamma_2 \geq 0} h(\gamma_1 y^{(k)} + \gamma_2 q^{(k)}, \gamma_1^{(k)} \phi^{(k)} + \gamma_2^{(k)})$
5. $\quad\quad y^{(k+1)} \leftarrow \gamma_1^{(k)} y^{(k)} + \gamma_2^{(k)} q^{(k)}$, $\phi^{(k+1)} \leftarrow \gamma_1^{(k)} \phi^{(k)} + \gamma_2^{(k)}$, $\quad k \leftarrow k+1$.

matrix $\tilde{W}$, define $Q = \max_{q \in B} \|q\|_{\tilde{W}}$. Algorithm 6.3 produces a sequence of pairs $(y^{(k)}, \phi^{(k)})_{k=1,2,\ldots}$ such that $h(y^{(k)}, \phi^{(k)})$ decreases monotonically. It terminates when $k = O(N\|a\|_{\tilde{W}} \max\{Q^2, 1\}/\delta)$, with

$$h(y^{(k)}, \phi^{(k)}) \leq h^* + \delta \|a\|_{\tilde{W}}.$$

Note that the result is essentially independent of the submodularity assumption and applies to general cones induced by arbitrary polytopes. Proving Theorem 6.3.1 requires a careful modification of the arguments in [85] to handle the conic structure. We provide a detailed proof in Section A.5.5.

6.3.2 The conic Frank-Wolfe algorithm

We now introduce the conic FW method, which is summarized in Algorithm 6.4. Note that Steps 2, 4 are specialized for cones. The main difference between the FW and MNP methods is the size of active set: FW only maintains two active points, while MNP may maintain as many as N points. A similar algorithm was developed in [104] for solving a more general objective. For completeness, we establish a $1/k$-convergence rate for the conic FW algorithm in Theorem 6.3.2.

Theorem 6.3.2. *Let B be an arbitrary polytope in $\mathbb{R}^N$ and let $C = \{(y, \phi) | y \in \phi B, \phi \geq 0\}$ denote its corresponding cone. For some positive diagonal matrix $\tilde{W}$, define $Q = \max_{q \in B} \|q\|_{\tilde{W}}$. Then, the point $(y^{(k)}, \phi^{(k)})$ generated by Algorithm 6.4 satisfies*

$$h(y^{(k)}, \phi^{(k)}) \leq h^* + \frac{2\|a\|_{\tilde{W}}^2 Q^2}{k+2}.$$

The proof of Theorem 6.3.2 is deferred to Section A.5.6.

6.3.3 Time complexity of the conic MNP and FW methods

Let EO stand for the time needed to query the function value of $F(\cdot)$. The linear program in Step 2 of both the MNP and FW algorithms requires $O(N \log N + N \times EO)$ operations if implemented as a greedy method. Step 6 of the MNP method requires solving a quadratic program with no constraints, and the solution may be obtained in closed form using $O(N|S|^2)$ operations. The remaining operations in the MNP method introduce a $O(N)$ complexity term. Hence, the execution of each MAJOR or MINOR loop requires $O(N \log N + N \times EO + N|S|^2)$ operations. In the FW method, the optimization problem in Step 4 is relatively simple, since it reduces to solving a nonnegative quadratic program with only two variables and consequently introduces a complexity term $O(N)$. Therefore, the complexity of the FW algorithm is dominated by Step 2, which requires $O(N \log N + N \times EO)$ operations.

Although the conic FW is computationally efficient in each iteration, the conic MNP method empirically requires fewer iterations in order to achieve a high-quality approximation (see, for example, the experimental results in Section 6.3.4). Moreover, the conic MNP method can compute exact projections in finite time, thereby paralleling a similar property of the MNP method pertaining to polytopes [101]. Because of the conic structure used in our setting, the optimal projections may be expressed as linear combinations of the extreme points of the polytopes. As the number of such combinations is finite, the conic MNP can find the exact projection in finitely many iterations. Note that for the conic FW method, this may not be the case. This is further evidenced from the numerical experiments in Section 6.3.4, where using the conic MNP method as the projection routine leads to better accuracy than using the conic FW method.

6.3.4 Numerical illustration: convergence of QRCD-MNP and QRCD-FW

We next illustrate the convergence behaviors of the RCD algorithm when applying the conic MNP method and the conic FW method as subroutine for computing the conic projection, referred to as QRCD-MNP and QRCD-FW, respectively. We generate a synthetic QDSFM problem (6.1) as follows.

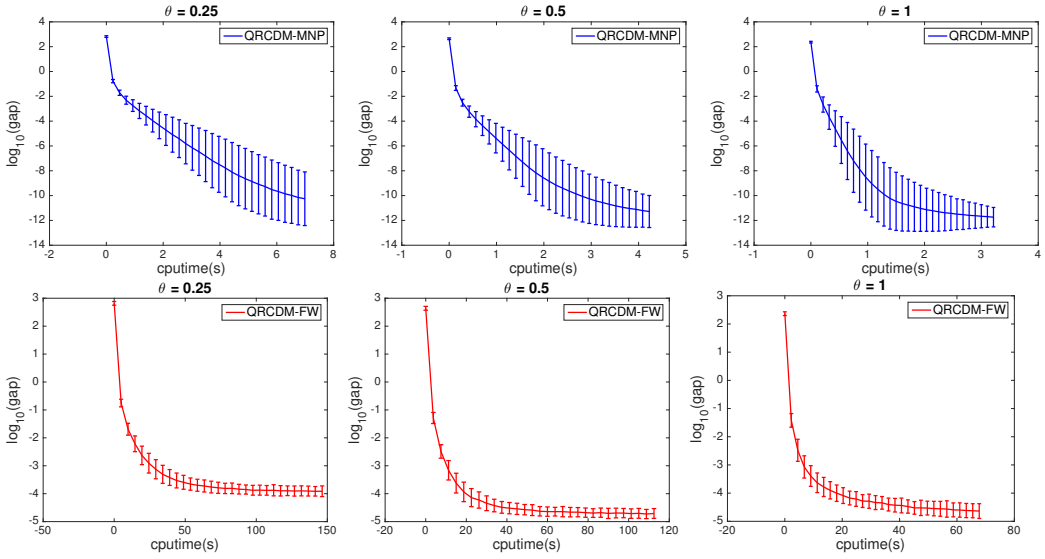


Figure 6.1: Convergence of QRCD-MNP and QRCD-FW for general submodular functions.

We fix $N = 100$, $R = 100$, and $W_{ii} = 1$, for all $i \in [N]$. We generate each incidence set S_r by choosing uniformly at random a subset of $[N]$ of cardinality 10, and set the entries of a to be iid standard Gaussian. We consider the following submodular functions: for $r \in [R]$, $S \subseteq S_r$

$$F_r(S) = \frac{\min\{|S|, |S_r/S|\}^\theta}{(|S_r|/2)^\theta}, \qquad \theta \in \{0.25,\, 0.5,\, 1\}.$$

The number of RCD iterations is set to $300R = 3 \times 10^4$. The convergence results are shown in Figure 6.1. QRCD-FW requires longer CPU-time to converge than QRCD-MNP. Moreover, QRCD-FW cannot achieve high accuracy because of the inaccurate inner-loop projections.

6.4 Applications to PageRank

We now introduce one important application of the QDSFM problem — PageRank (PR). Our treatment of the PR process relies on diffusion processes over hypergraphs [105, 23]. We demonstrate that the underlying PR vector can be efficiently computed via our proposed QDSFM solvers. We also show that the newly introduced PR retains important properties of the standard PR over graphs, pertaining to mixing and local partitioning [106].

6.4.1 Background: PageRank(PR)

PR is a well-known algorithm over graphs, first introduced in the context of webpage search [21]. Given a graph $G = (V, E)$, with $V = [N]$ and E denoting the edge set over V, let A and D denote the adjacency matrix and diagonal degree matrix of G, respectively. PR essentially reduces to finding a fixed point $p \in \mathbb{R}^N$ via the iterative procedure

$$p^{(t+1)} = \alpha p^{(0)} + (1 - \alpha) A D^{-1} p^{(t)},$$

where $p^{(0)} \in \mathbb{R}^N$ is a fixed vector and $\alpha \in (0, 1)$. By using different exploration vectors $p^{(0)}$, PR can be adapted to different graph-based applications. For example, for ranking of Web pages [21], $p^{(0)}$ may be a uniformly at random chosen vertex. For finding local graph partitioning around a vertex i [106], $p^{(0)}$ is typically set to e_i, i.e., a zero-one vector with a single 1 in its i-th component.

It is easy to verify that p is a solution to the problem

$$\min_p \frac{\alpha}{1-\alpha} \|p - p^{(0)}\|^2_{D^{-1}} + (D^{-1}p)^T (D - A)(D^{-1}p) = \|x - a\|^2_W + \langle x, L(x) \rangle, \tag{6.10}$$

where $x = D^{-1}p$, $a = D^{-1}p^{(0)}$, $W = \frac{\alpha}{1-\alpha} D$ and $L = D - A$.

Recent work illuminating the role of high-order functional relations among vertices in networks [39, 71, 107] brought forward the need to study PR-type of problems for hypergraphs [108, 109].[1] There exist several definitions of the PR algorithm for hypergraphs. For example, the approach in [108] relies on what is termed a *multilinear PR* [110] and is limited to the setting of uniform hypergraphs, in which all hyperedges have the same cardinality. In addition, computing a multilinear PR requires tensor multiplication and has complexity exponential in the size of the hyperedges. The PR formulation in [109] requires one to first *project hypergraphs onto graphs* and then run standard graph-based PR. The projection step usually causes large distortions when the hyperedges are large and leads to low-quality hypergraph partitioning [9, 111].

Using equation (6.10), we define a new formulation for PR on hypergraphs

[1] Hypergraphs are natural extensions of graphs in which edges are replaced by hyperedges, representing subsets of vertices in V of size ≥ 2.

Table 6.1: The term $\langle x, L(x)\rangle$ for different combinatorial structures. In the third column, whenever the stated conditions are not satisfied, it is assumed that $F_r = 0$. For directed hypergraphs, H_r and T_r are subsets of S_r which we subsequently term the *head* and the *tail set*. For $H_r = T_r = S_r$, one recovers the setting for undirected hypergraphs.

A single component of $\langle x, L(x)\rangle$	The combinatorial structure	The submodular function
$w_r(x_i - x_j)^2$, $S_r = \{i, j\}$	A graph [98, 13]	$F_r(S) = \sqrt{w_{ij}}$ if $\lvert S \cap \{i, j\}\rvert = 1$
$w_r \max_{i,j \in S_r}(x_i - x_j)^2$	A hypergraph [9]	$F_r(S) = \sqrt{w_r}$ if $\lvert S \cap S_r\rvert \in [1, \lvert S_r\rvert - 1]$
$w_r \max\limits_{(i,j) \in H_r \times T_r}(x_i - x_j)_+^2$	A directed hypergraph [99]	$F_r(S) = \sqrt{w_r}$ if $\lvert S \cap H_r\rvert \geq 1$, $\lvert([N]/S) \cap T_r\rvert \geq 1$
General $[f_r(x)]^2$	A submodular hypergraph [111]	Any symmetric submodular function

that leverages nonlinear Laplacian operators $L(\cdot)$ for (un)directed hypergraphs [105, 23] and submodular hypergraphs [111, 112]. The new PR may be applied to both uniform and non-uniform hypergraphs with arbitrarily large hyperedges. The stationary point is the solution to a QDSFM problem, where the term $\langle x, L(x)\rangle$ in (6.10) is represented by a sum of Lovás extensions of submodular functions defined over hyperedges (see Table 6.1). During the review of this article, many other interesting works based on this hypergraph Laplacian operator have appeared: [113] looked into polynomial-time algorithms to compute the inverse of this Laplacian operator; [114] considered the derived heat-kernel PageRank; [115] attempted to sparsify hypergraphs while keeping the spectrum of this Laplacian operator.

To demonstrate the utility of this particular form of PageRank, we show that the PR vector obtained as a solution of the QDSFM can be used to find a cut of a directed hypergraph (as well as undirected hypergraph) with small conductance, which is an important invariant for local partitioning and community detection [108, 109]. The results essentially generalize the mixing results for PR over graphs derived in [106] and [116].

Our main technical contribution is handling new technical challenges that arise due to the non-linearity of the operators involved. In what follows, we mainly focus on directed hypergraphs. Tighter and more general results may be obtained for submodular hypergraphs [111, 112], which will be left for the future study.

6.4.2 Terminology: boundary, volume, conductance

In Chapter 2, we have the definition for general submodular hypergraphs. Here, we specify our attention to directed hypergraphs. To be more clear, we reintroduce the following notations. Let $G = (V, E)$ be a hypergraph with $V = [N]$ and hyperedge in E equated with the incidence sets $\{S_r\}_{r\in[R]}$. More precisely, each hyperedge represents an incidence set S_r associated with a triplet (w_r, H_r, T_r), where w_r is a scalar weight and H_r and $T_r \subseteq S_r$ are the head and tail sets defined in Table 6.1. The pair (H_r, T_r) defines a submodular function F_r with incidence set S_r according to $F_r(S) = 1$ if $|S \cap H_r| \geq 1$ and $|\bar{S} \cap T_r| \geq 1$ (where $\bar{S}$ stands for the complement of S), and $F_r(S) = 0$ otherwise. The corresponding Lovász extension reads as $f_r(x) = \max_{(i,j)\in H_r\times T_r}(x_i - x_j)$. If S_r contains only two vertices for all values $r \in [R]$, the hypergraph clearly reduces to a graph; and similarly, if $H_r = T_r = S_r$, directed hypergraphs reduce to undirected hypergraphs. We also define the degree of a vertex i as

$$d_i = \sum_{r:i\in S_r} w_r,$$

and let D denote the diagonal matrix of degree values. The volume of a set $S \subseteq V$ is defined as

$$\mathrm{vol}(S) = \sum_{i\in S} d_i.$$

Furthermore, let $m = \sum_{i\in V} d_i = \mathrm{vol}(V)$ and define the *boundary* of a set S and its *volume* as

$$\partial S = \{r \in [R] | S \cap H_r \neq \emptyset, \bar{S} \cap T_r \neq \emptyset\}, \quad \mathrm{vol}(\partial S) = \sum_{r\in\partial S} w_r = \sum_{r\in[R]} w_r F_r(S).$$

Using the boundary and volume, we also define the *conductance* of the set S as

$$\Phi(S) = \frac{\mathrm{vol}(\partial S)}{\min\{\mathrm{vol}(S), \mathrm{vol}(\bar{S})\}}.$$

The boundary, volume and conductance are all standard and well-studied graph-theoretic concepts, but generalized above to accommodate hypergraphs.

Additionally, we also make use of a distribution π_S over V, $(\pi_S)_i = \frac{d_i}{\mathrm{vol}(S)}$.

6.4.3 The PageRank process as QDSFM optimization

Next, we formally define the PR process for directed hypergraphs and show how it relates to the QDSFM problem. We borrow the definition of a diffusion process (DP) over directed hypergraphs proposed in [105, 23] to describe our PageRank (PR) process. To this end, we first need to define a Markov operator as follows.

Suppose that $p \in \mathbb{R}^N$ is a potential over $[N]$ and let $x = D^{-1}p$. For all hyperedges S_r, we let $S_r^{\downarrow}(x) = \arg\min_{i \in T_r} x_i$ and $S_r^{\uparrow}(x) = \arg\max_{j \in H_r} x_j$.

1. For each hyperedge S_r, and for each pair of vertices $(i, j) \in S_r^{\uparrow}(x) \times S_r^{\downarrow}(x)$ that satisfy $x_i > x_j$, we introduce a positive scalar $a_r^{(ij)}$ such that $\sum_{(i,j) \in S_r^{\uparrow}(x) \times S_r^{\downarrow}(x)} a_r^{(ij)} = w_r$. For pairs of vertices (i, j) not satisfying the above constraint, we set $a_r^{(ij)} = 0$.

2. Using the above described pairwise weights, we create a symmetric matrix A such that $A_{ij} = \sum_{r \in [R]} a_r^{(ij)}$ and $A_{ii} = d_i - \sum_{i,j \in V, j \neq i} A_{ij}$.

Then, the Markov operator $M(p)$ based on A is defined according to

$$M(p) = Ax = AD^{-1}p.$$

The Markov operator $M(p)$ is closely related to the Lovász extension f_r, which may be seen as follows. Let $\nabla f_r(x)$ be the subdifferential set of f_r at x. We once again recall that as F_r is submodular, $f_r(x) = \arg\max_{y \in B_r} \langle y, x \rangle$ [28]. This gives rise to the following result.

Lemma 6.4.1. *For any $p \in \mathbb{R}^N$ and $x = D^{-1}p$, it holds that $p - M(p) \in \sum_{r \in [R]} w_r f_r(x) \nabla f_r(x)$.*

Proof. Consider a vertex $i \in V$. Then,

$$\begin{aligned}
&(p - M(p))_i \\
&= p_i - A_{ii}\frac{p_i}{d_i} - \sum_{j:\in V, j\neq i} A_{ij}\frac{p_j}{d_j} \\
&= \sum_{j:\in V, j\neq i} A_{ij}(x_i - x_j) = \sum_{r\in[R]} \sum_{j:\in V, j\neq i} a_r^{(ij)}(x_i - x_j) \\
&\overset{1)}{=} \sum_{r:\in[R], i\in S_r^{\uparrow}} w_r f_r(x) \sum_{j:\in S_r^{\downarrow}} a_r^{(ij)}/w_r + \sum_{r:\in[R], i\in S_r^{\downarrow}} w_r f_r(x) \sum_{j:\in S_r^{\uparrow}} (-a_r^{(ij)})/w_r,
\end{aligned}$$

where 1) holds due to the definitions of $a_r^{(ij)}$ and $f_r(x)$. It only remains to show that for a $r \in [R]$, the vector v

$$v_i = \begin{cases} \sum_{j:\in S_r^{\downarrow}} a_r^{(ij)}/w_r & i \in S_r^{\uparrow}, \\ -\sum_{j:\in S_r^{\uparrow}} a_r^{(ij)}/w_r & i \in S_r^{\downarrow}, \\ 0 & \text{otherwise} \end{cases}$$

satisfies $v \in \nabla f_r(x)$. Since $\sum_{(i,j)\in S_r^{\uparrow}\times S_r^{\downarrow}} a_r^{(ij)} = w_r$ and $a_r^{(ij)} \geq 0$, we know that $v \in B_r$. Moreover, since $\langle v, x\rangle = f_r(x)$, the claim clearly holds. □

The statement of Lemma 6.4.1 describes a definition of a Laplacian operator for submodular hypergraphs consistent with previous definitions [111]. We write the underlying Laplacian operator as $L(x) = (D-A)x = p - M(p)$. When $p = \pi_V$, the components of x are equal and hence $M(p) = AD^{-1}p = DD^{-1}p = p$. Therefore, π_V may be viewed as an analogue of a stationary distribution.

Given an initial potential $p_0 \in \mathbb{R}^N$ and an $\alpha \in (0, 1]$, the PR process is described by the following ordinary differential equation:

$$\textbf{The PR process} \quad \frac{dp_t}{dt} = \alpha(p_0 - p_t) + (1-\alpha)(M(p_t) - p_t).$$

The choice of $\alpha = 0$ reduces the PR process to the DP defined in [105, 23]. Tracking the PR process at every time point t is as difficult as tracking DP, which in turn requires solving a densest subset problem [105, 23]. However, we only need to examine and evaluate the stationary point which corresponds

to the PR vector. The PR vector $pr(\alpha, p_0)$ satisfies the equation

$$\alpha(p_0 - pr(\alpha, p_0)) + (1-\alpha)(M(pr(\alpha, p_0)) - pr(\alpha, p_0)) = 0. \tag{6.11}$$

From Lemma 6.4.1, we know that $pr(\alpha, p_0)$ can be obtained by solving a QDSFM. Let $x_{pr} = D^{-1}pr(\alpha, p_0)$. Then

$$\begin{aligned} &\alpha(p_0 - Dx_{pr}) - (1-\alpha)\sum_{r\in[R]} w_r f_r(x_{pr})\nabla f_r(x_{pr}) \ni 0 \\ \Leftrightarrow\ &x_{pr} = \arg\min_x \|x - x_0\|_W^2 + \sum_{r\in[R]} [f_r'(x)]^2, \end{aligned} \tag{6.12}$$

where $x_0 = D^{-1}p_0, W = \frac{\alpha}{1-\alpha}D$ and $f_r' = \sqrt{w_r}f_r$.

6.4.4 Complexity of computing PageRank over directed hypergraphs

We now analyze the computation complexity of computing PageRank over directed hypergraphs using QSDFM solvers. First, due to the specific structure of the Lovász extension f_r' in (6.12), we devise an algorithm of complexity $O(|S_r|\log|S_r|)$ that *exactly* computes the required projections onto the induced cones. This projection algorithm is much more efficient than the conic FW and MNP methods that apply to general polytopes. We focus on projections with respect to one particular value r; since the scaling w_r does not affect the analysis, we henceforth omit the subscript r and the superscript $'$ and simply use f instead of f_r'.

The key idea is to revert the projection $\Pi_C(a; \tilde{W})$ back to its primal form. In the primal setting, optimization becomes straightforward and only requires careful evaluation of the gradient values. First, following a similar strategy as described in Lemma 6.1.1, it is easy to show that

$$\min_z \|z - b\|_W^2 + [f(z)]^2$$

is the dual of the problem (6.9), where $W = \tilde{W}^{-1}$, $b = \frac{1}{2}W^{-1}a$, and f is the Lovász extension corresponding to the base polytope B. Then, one has $y = a - 2Wz$, $\phi = 2\langle y, z\rangle_{W^{-1}}$.

Next, recall that for a directed hyperedge, we introduced head and tail

Algorithm 6.5: Projection for a Directed Hyperedge

Input: W, b, H, T

1. Sort $\{b_i\}_{i\in H}$ and $\{b_j\}_{j\in T}$.
2. Initialize $\gamma \leftarrow \max_{i\in H} b_i$ and $\delta \leftarrow \min_{j\in T} b_j$.
3. **If** $\gamma \leq \delta$, **return** $z = b$.
4. Iteratively execute:
5. $\quad w_H \leftarrow \sum_{i\in S_H(\gamma)} W_v$, $w_T \leftarrow \sum_{j\in S_T(\delta)} W_v$
6. $\quad \gamma_1 \leftarrow \max_{i\in H/S_H(\gamma)} b_v$, $\delta_1 \leftarrow \delta + (\gamma - \gamma_1)w_H/w_T$
7. $\quad \delta_2 \leftarrow \min_{j\in T/S_T(\delta)} b_v$, $\gamma_2 \leftarrow \gamma - (\delta_2 - \delta)w_T/w_H$
8. $\quad k^* \leftarrow \arg\min_{k\in\{1,2\}} \delta_k$
9. $\quad$ **If** $\gamma_{k^*} \leq \delta_{k^*}$ or $\triangle_{\gamma_{k^*}} \leq 0$, **break**
10. $\quad (\gamma, \delta) \leftarrow (\gamma_{k^*}, \delta_{k^*})$
11. $(\gamma, \delta) \leftarrow (w_T, w_H)\frac{\triangle_\gamma}{w_T w_H + w_T + w_H}$
12. Set z_i to γ, if $i \in S_H(\gamma)$, to δ, if $i \in S_T(\delta)$, and to b_i otherwise.

sets H, T, respectively, and $f(z) = \max_{i\in H} z_i - \min_{j\in T} z_j$. Clearly, when H, T both equal to the corresponding incidence set, the directed hypergraph reduces to an undirected hypergraph. To solve this optimization problem, define two intermediate variables $\gamma = \max_{i\in H} z_i$ and $\delta = \min_{j\in T} z_j$. Denote the derivates with respect to γ and δ as $\triangle_\gamma$ and $\triangle_\delta$, respectively. The optimal values of γ and δ satisfy

$$\triangle_\gamma = \gamma - \delta + \sum_{i\in S_H(\gamma)} W_i(\gamma - b_i), \quad \triangle_\delta = \delta - \gamma + \sum_{j\in S_T(\delta)} W_j(\delta - b_j),$$

where $S_H(\gamma) = \{i | i \in H, b_i \geq \gamma\}$ and $S_T(\delta) = \{j | j \in T, b_j \leq \delta\}$. The optimal values of γ and δ are required to simultaneously satisfy $\triangle_\gamma = 0$ and $\triangle_\delta = 0$. Algorithm 6.5 can be used to find such values of γ and δ. The "search" for (γ, δ) starts from $(\max_{i\in H} b_i, \min_{j\in T} b_j)$, and one gradually decreases γ and increases δ while keeping $\triangle_\gamma = -\triangle_\delta$ (see Steps 5-10 in Algorithm 6.5). The complexity of Algorithm 6.5 is dominated by the sorting step, which requires $O(|S_r| \log |S_r|)$ operations, as w_H, w_T, $\triangle_\gamma$, and $\triangle_\delta$ can all be efficiently tracked within the inner loops.

Combining the projection method in Algorithm 6.5 and Corollary 6.2.7 with parameter choice $\beta = \frac{\alpha}{1-\alpha}$, we arrive at the following result summarizing the overall complexity of running the PageRank process on (un)directed hypergraphs.

Corollary 6.4.2. *The PageRank problem on (un)directed hypergraphs can be*

obtained by solving (6.12). *A combination of Algorithm 6.1 (RCD) and Algorithm 6.5 returns an ϵ-optimal solution in expectation with total computation complexity:*

$$O\left(N^2 \max\left\{1, 9\frac{1-\alpha}{\alpha}\right\} \max_{i,j\in[N]} \frac{D_{ii}}{D_{jj}} \sum_{r\in[R]} |S_r| \log |S_r| \log \frac{1}{\epsilon}\right).$$

A combination of Algorithm 6.2 (AP) and Algorithm 6.5 returns an ϵ-optimal solution with total computation complexity:

$$O\left(N^2 \max\left\{1, 9\frac{1-\alpha}{\alpha}\right\} \max_{i,j\in[N]} \frac{\Psi_{jj} D_{ii}}{D_{jj}} \sum_{r\in[R]} |S_r| \log |S_r| \log \frac{1}{\epsilon}\right).$$

6.4.5 Analysis of partitions based on PageRank

We are now ready to analyze hypergraph partitioning procedures based on our definition of PageRank. We first prove that our version of PageRank, similarly to the standard PageRank over graphs [106], satisfies the mixing result derived based on Lováz-Simonovits curves [117]. This result may be further used to prove that *Personalized* PageRank leads to partitions with small conductance.[2] The main parts of our proof are inspired by Andersen's proof for PageRank over graphs [106]. The novelty of our approach is that we need several specialized steps to handle the nonlinearity of the Markov operator $M(\cdot)$, as in the standard graph setting the Markov operator is linear. We postpone the proofs of all our results to Section A.5.7, but introduce all relevant concepts and terminology in what follows.

First, we define *sweep cuts* that are used to partition hypergraphs based on a distribution p. The sweep cut based on a given potential vector $p \in \mathbb{R}^N$ is used to partition the hypergraph. Recall that $x = D^{-1}p$ and sort the components of x so that

$$x_{i_1} \geq x_{i_2} \geq \cdots \geq x_{i_N}. \tag{6.13}$$

[2]Personalized PageRank is a PR process with initial distribution $p_0 = 1_i$, for some vertex $i \in V$.

Let $\mathcal{S}_j^p = \{x_{i_1}, x_{i_2}, ..., x_{i_N}\}$ and evaluate

$$j^* = \arg\min_{j\in[N]} \Phi(\mathcal{S}_j^p).$$

Then, $(\mathcal{S}_{j^*}^p, \bar{\mathcal{S}}_{j^*}^p)$ is what we refer to as the sweep cut used for partitioning the hypergraph. Moreover, let $\Phi_p = \Phi(\mathcal{S}_{j^*}^p)$. We are now ready to prove a mixing result for the PR process in terms of $\Phi_{pr(\alpha,p_0)}$.

Second, we introduce the Lováz-Simonovits curve to characterize the distribution of a potential. Given a potential vector $p \in \mathbb{R}^N$ and $x = D^{-1}p$, suppose that the order of the components in x follows equation (6.13). Let $\text{vol}(\mathcal{S}_0^p) = 0$. Define a piecewise linear function $I_p(\cdot) : [0, m] \to [0, 1]$ according to

$$I_p(z) = p(\mathcal{S}_{j-1}^p) + \frac{z - \text{vol}(\mathcal{S}_{j-1}^p)}{d_{v_j}} p_{v_j}, \text{ for } \text{vol}(\mathcal{S}_{j-1}^p) \le k \le \text{vol}(\mathcal{S}_j^p),\ j \in [N].$$

It is easy to check that $I_p(z)$ is continuous and concave in z. Moreover, for any set $S \subseteq [N]$, we have $p(S) \le I_p(\text{vol}(S))$. We further write $V_p = \{j : x_{i_j} > x_{i_{j+1}}\}$ and refer to $\{\text{vol}(\mathcal{S}_j^p)\}_{j\in V_p}$ as the (exact) break points of I_p.

The following Lemma establishes an upper bound on the break points.

Lemma 6.4.3. *Let $p = pr(\alpha, p_0)$, $x = D^{-1}p$ and $j \in V_p$. Then,*

$$\begin{aligned} I_p(vol(\mathcal{S}_j^p)) \le & \frac{\alpha}{2-\alpha} p_0(\mathcal{S}_j^p) \\ & + \frac{1-\alpha}{2-\alpha}[I_p(vol(\mathcal{S}_j^p) - vol(\partial\mathcal{S}_j^p)) + I_p(vol(\mathcal{S}_j^p) + vol(\partial\mathcal{S}_j^p)]. \end{aligned}$$

Furthermore, for $k \in [0, m]$,

$$I_p(k) \le p_0(k) \le I_{p_0}(k).$$

Remark 6.4.1. *In comparison with the undirected graph case (Lemma 5 [106]), where the result holds for arbitrary $S \subseteq V$, our claim is true only for $\mathcal{S}_j^p$ for which $j \in V_p$. However, this result suffices to establish all relevant PR results.*

Using the upper bound on the break points in I_p, we can now construct a curve that uniformly bounds I_p using Φ_p.

Theorem 6.4.4. *Let $p = pr(\alpha, p_0)$ be the previously defined PR vector, and let Φ_p be the minimal conductance achieved by a sweep cut. For any integer*

$t \geq 0$ and any $k \in [0, m]$, the following bound holds:

$$I_p(k) \leq \frac{k}{m} + \frac{\alpha}{2-\alpha}t + \sqrt{\frac{\min\{k, m-k\}}{\min_{i:(p_0)_i>0} d_i}} \left(1 - \frac{\Phi_p^2}{8}\right)^t.$$

For graphs [106], Theorem 6.4.4 may be used to characterize the graph partitioning property of sweep cuts induced by a personalized PageRank vector. Similarly, for general directed hypergraphs, we establish a similar result in Theorem 6.4.5.

Theorem 6.4.5. *Let S be a set of vertices such that $vol(S) \leq \frac{1}{2}M$ and $\Phi(S) \leq \frac{\alpha}{c}$, for some constants α, c. If there exists a distribution P for sampling vertices $v \in S$ such that $\mathbb{E}_{i\sim P}[pr(\alpha, 1_i)(\bar{S})] \leq \frac{c}{8} pr(\alpha, \pi_S)(\bar{S})$, then with probability at least $\frac{1}{2}$,*

$$\Phi_{pr(\alpha,1_i)} = O(\sqrt{\alpha \log \frac{vol(S)}{d_i}}),$$

where i is sampled according to P.

Remark 6.4.2. *Note that in Theorem 6.4.5, we require that the sampling distribution P satisfy $\mathbb{E}_{i\sim P}[pr(\alpha, 1_i)(\bar{S})] \leq \frac{c}{8} pr(\alpha, \pi_S)(\bar{S})$. For graphs, when we sample a vertex i with probability proportional to its degree d_i, this condition is naturally satisfied, with $c = 8$. However, for general (un)directed hypergraphs, the sampling procedure is non-trivial and harder to handle due to the non-linearity of the random-walk operator $M(\cdot)$. We relegate a more in-depth study of this topic as a future direction.*

6.5 Applications to semi-supervised learning

Another important application of QDSFM is semi-supervised learning (SSL). SSL is a learning paradigm that allows one to utilize the underlying structure or distribution of unlabeled samples whenever the information provided by labeled samples is insufficient for learning an inductive predictor [118, 119]. A standard setup for a K-class transductive learner is as follows: given N data points $\{z_i\}_{i\in[N]}$, along with labels for the first l ($\ll N$) samples $\{y_i | y_i \in [K]\ \}_{i\in[l]}$, the learner is asked to infer the labels of all the remaining data

points $i \in [N]/[l]$. The widely-used SSL problem with least square loss requires one to solve K regularized problems. For each class $k \in [K]$, one sets the scores of data points within the class to

$$\hat{x}^{(k)} = \arg\min_{x^{(k)}} \beta \|x^{(k)} - a^{(k)}\|^2 + \Omega(x^{(k)}),$$

where $a^{(k)}$ describes the information provided by the known labels, i.e., $a_i^{(k)} = 1$ if $y_i = k$, and 0 otherwise, β denotes a hyperparameter and Ω is a smoothness regularizer. The labels of the data points are estimated as

$$\hat{y}_i = \arg\max_k \{\hat{x}_i^{(k)}\}.$$

In typical graph and hypergraph learning problems, Ω is chosen to be a Laplacian regularizer constructed using $\{z_i\}_{i\in[N]}$; it is this regularization term that ties the above learning problems to PageRank (again, refer to Table 6.1). With Laplacian regularization, one has that each edge/hyperedge corresponds to one functional component in the QDSFM problem. The variables themselves may be normalized with respect to their degrees, in which case the normalized Laplacian is used instead. For example, in graph learning, one of the terms in Ω assumes the form $w_{ij}(x_i/\sqrt{d_i} - x_j/\sqrt{d_j})^2$, where d_i and d_j correspond to the degrees of the vertex variables i and j, respectively. Using some simple algebra, it can be shown that the normalization term is accounted for by the matrix W used in the definition of the QDSFM problem (6.1).

6.5.1 Experiment setup

We numerically evaluate our SSL learning framework for hypergraphs on both real and synthetic datasets. For the particular problem at hand, the QDSFM problem can be formulated as follows:

$$\min_{x\in\mathbb{R}^N} \beta\|x - a\|^2 + \sum_{r\in[R]} \max_{i,j\in S_r} \left(\frac{x_i}{\sqrt{W_{ii}}} - \frac{x_j}{\sqrt{W_{jj}}}\right)^2, \tag{6.14}$$

where $a_i \in \{-1, 0, 1\}$ indicates if the corresponding variable i has a negative, missing, or positive label, respectively, β is a parameter that balances out the influence of observations and the regularization term, $\{W_{ii}\}_{i\in[N]}$ defines

a positive measure over the variables and may be chosen to be the degree matrix D, with $D_{ii} = |\{r \in [R] : i \in S_r\}|$. Each part in the decomposition corresponds to one hyperedge. We compare eight different solvers falling into three categories: (a) our proposed general QDSFM solvers, *QRCD-SPE, QRCD-MNP, QRCD-FW and QAP-SPE*; (b) alternative specialized solvers for the given problem (6.14), including *PDHG* [9] and *SGD* [99]; (c) SSL solvers that do not use QDSFM as the objective, including *DRCD* [20] and *InvLap* [7]. The first three methods all have outer-loops that execute RCD, but with different inner-loop projections computed via the *exact projection algorithm* for undirected hyperedges (see Algorithm 6.5 in Section 6.4.4), or the generic MNP and FW. As it may be time consuming to find the precise projections via MNP and FW, we always fix the number of MAJOR loops of the MNP and the number of iterations of the FW method to $100|S_r|$ and $100|S_r|^2$, respectively. Empirically, these choices provide an acceptable trade-off between accuracy and complexity. The QAP-SPE method uses AP in the outer-loop and exact inner-loop projections (see Algorithm 6.5 in Section 6.4.4). PDHG and SGD are the only known solvers for the specific objective (6.14). PDHG and SGD depend on certain parameters that we choose in standard fashion: for PDHG, we set $\sigma = \tau = \frac{1}{\sqrt{1+\max_i D_{ii}}}$ and for SGD, we set $\eta_k = \frac{1}{k\beta \max_i W_{ii}}$.

DRCD is a state-of-the-art solver for DSFM and also uses a combination of outer-loop RCD and inner-loop projections. InvLap first transforms hyperedges into cliques and then solves a Laplacian-based linear problem. All the aforementioned methods, except InvLap, are implemented in C++ in a nonparallel fashion. InvLap is executed via matrix inversion operations in Matlab, and may be parallelized. The CPU times of all methods are recorded on a 3.2GHz Intel Core i5. The results are summarized for 100 independent tests. When reporting the results, we use the primal gap ("gap") to characterize the convergence properties of different solvers.

6.5.2 Synthetic data

We generated a hypergraph with $N = 1000$ vertices that belong to two equal-sized clusters. We uniformly at random generated 500 hyperedges within each cluster and 1000 hyperedges across the two clusters. Note that in higher-order

clustering, we do not need to have many hyperedges within each cluster to obtain good clustering results. Each hyperedge includes 20 vertices. We also uniformly at random picked $l = 1, 2, 3, 4$ vertices from each cluster to represent labeled datapoints. With the vector x obtained by solving (6.14), we classified the variables based on the Cheeger cut rule [9]: suppose that $\frac{x_{i_1}}{\sqrt{W_{i_1 i_1}}} \geq \frac{x_{i_2}}{\sqrt{W_{i_2 i_2}}} \geq \cdots \geq \frac{x_{i_N}}{\sqrt{W_{i_N i_N}}}$, and let $\mathcal{S}_j = \{i_1, i_2, ..., i_j\}$. We partition $[N]$ into two sets $(\mathcal{S}_{j^*}, \bar{\mathcal{S}}_{j^*})$, where

$$j^* = \arg\min_{j \in [N]} \Phi(\mathcal{S}_j) \triangleq \frac{|S_r \cap \mathcal{S}_j \neq \emptyset, S_r \cap \bar{\mathcal{S}}_j \neq \emptyset\}|}{\max\{\sum_{r\in[R]} |S_r \cap \mathcal{S}_j|, \sum_{r\in[R]} |S_r \cap \bar{\mathcal{S}}_j|\}}.$$

The classification error is defined as (# of incorrectly classified vertices)/N. In the experiment, we used $W_{ii} = D_{ii}$, $\forall\, i$, and tuned β to be nearly optimal for different objectives with respect to the classification error rates: for QDSFM as the objective, using QRCD-SPE, QAP-SPE, PDHG, and SGD as the methods of choice, we set $\beta = 0.02$; for DSFM as the objective, including the DRCD method, we set $\beta = 1$; for InvLap, we set $\beta = 0.001$.

The left subfigure of Figure 6.2 shows that QRCD-SPE converges much faster than all other methods when solving the problem (6.14) according to the gap metric (we only plotted the curve for $l = 3$ as all other values of l produce similar patterns). To avoid clutter, we moved the results for QRCD-MNP and QRCD-FW to the right two subfigures in Figure 6.2, as these methods are typically 100 to 1000 times slower than QRCD-SPE. Table 6.2 lists the performance of different methods with comparable CPU-times. Note that when QRCD-SPE converges (with primal gap 10^{-9} achieved after 0.83s), the obtained x leads to a much smaller classification error than other methods. QAP-SPE, PDHG and SGD all have large classification errors as they do not converge within short CPU time-frames. QAP-SPE and PDHG perform only a small number of iterations, but each iteration computes the projections for all the hyperedges, which is more time-consuming. The poor performance of DRCD implies that the DFSM is not a good objective for SSL. InvLap achieves moderate classification errors, but still does not match the performance of QRCD-SPE. Furthermore, note that InvLap uses Matlab, which is optimized for matrix operations, and is hence fairly efficient. However, for experiments on real datasets, where one encounters fewer but significantly larger hyperedges, InvLap does not offer performance

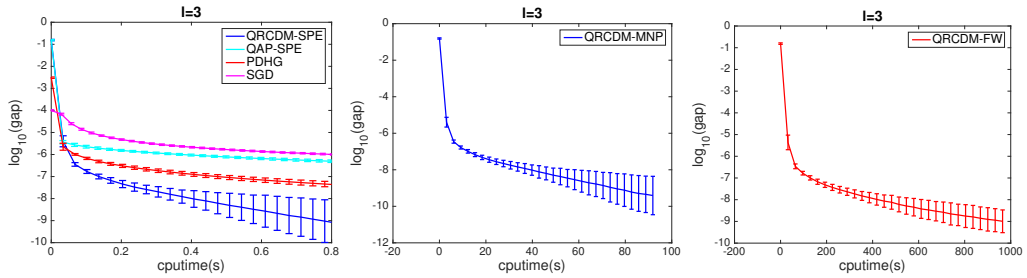


Figure 6.2: Convergence results on synthetic datasets: gap vs CPU-time of different QDSFM solvers (with an average ± standard deviation).

Table 6.2: Prediction results on synthetic datasets: classification error rates & Average 100 $c(\mathcal{S}_{j^*})$ for different solvers (MN: mean, MD: median).

Obj.	Solvers	Classification error rate (%)								Average $100c(\mathcal{S}_{j^*})$				#iterations	cputime(s)
		l=1		l=2		l=3		l=4							
		MN	MD	MN	MD	MN	MD	MN	MD	l=1	l=2	l=3	l=4		
QDSFM	QRCD-SPE	**2.93**	**2.55**	**2.23**	**0.00**	**1.47**	**0.00**	**0.78**	**0.00**	**6.81**	**6.04**	**5.71**	**5.41**	4.8×10^5	0.83
	QAP-SPE	14.9	15.0	12.6	13.2	7.33	8.10	4.07	3.80	9.51	9.21	8.14	7.09	2.7×10^2	0.85
	PDHG	9.05	9.65	4.56	4.05	3.02	2.55	1.74	0.95	8.64	7.32	6.81	6.11	3.0×10^2	0.83
	SGD	5.79	4.15	4.30	3.30	3.94	2.90	3.41	2.10	8.22	7.11	7.01	6.53	1.5×10^4	0.86
Oth.	DRCD	44.7	44.2	46.1	45.3	43.4	44.2	45.3	44.6	9.97	9.97	9.96	9.97	3.8×10^6	0.85
	InvLap	8.17	7.30	3.27	3.00	1.91	1.60	0.89	0.70	8.89	7.11	6.18	5.60	—	**0.07**

as good as that for synthetic data. The column "Average $100c(\mathcal{S}_{j^*})$" also illustrates that the QDSFM objective is a good choice for finding approximate balanced cuts of hypergraphs.

We also evaluated the influence of parameter choices on the convergence of QRCD methods. According to Theorem 6.2.2, the required number of RCD iterations for achieving an ϵ-optimal solution for (6.14) is roughly $O(N^2 R \max(1, 9/(2\beta)) \max_{i,j\in[N]} W_{ii}/W_{jj} \log 1/\epsilon)$ (see Section A.5.8). We mainly focus on testing the dependence on the parameters β and $\max_{i,j\in[N]} \frac{W_{ii}}{W_{jj}}$, as the term $N^2 R$ is also included in the iteration complexity of DSFM and was shown to be necessary given certain submodular structures [70]. To test the effect of β, we fix $W_{ii} = 1$ for all i, and vary $\beta \in [10^{-3}, 10^3]$. To test the effect of W, we fix $\beta = 1$ and randomly choose half of the vertices and set their W_{ii} values to lie in $\{1, 0.1, 0.01, 0.001\}$, and set the remaining ones to 1. Figure 6.3 show our results. The logarithm of gap ratios is proportional to $\log \beta^{-1}$ for small β, and $\log \max_{i,j\in[N]} \frac{W_{ii}}{W_{jj}}$, which is not as sensitive as predicted by Theorem 6.2.2. Moreover, when β is relatively large (> 1), the dependence on β levels out.

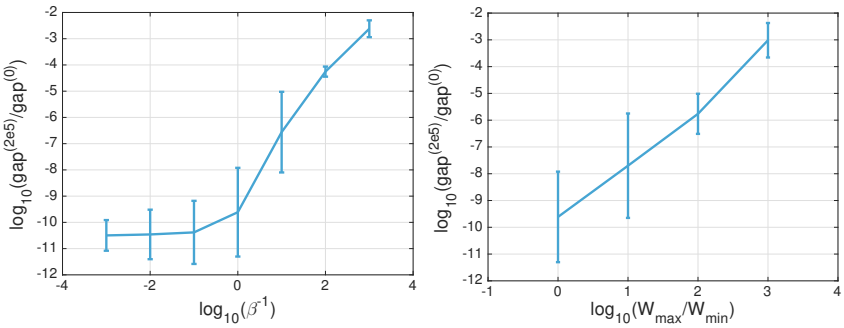


Figure 6.3: Parameter sensitivity: the rate of a primal gap of QRCD after 2×10^5 iterations with respect to different choices of the parameters β & $\max_{i,j\in[N]} W_{ii}/W_{jj}$.

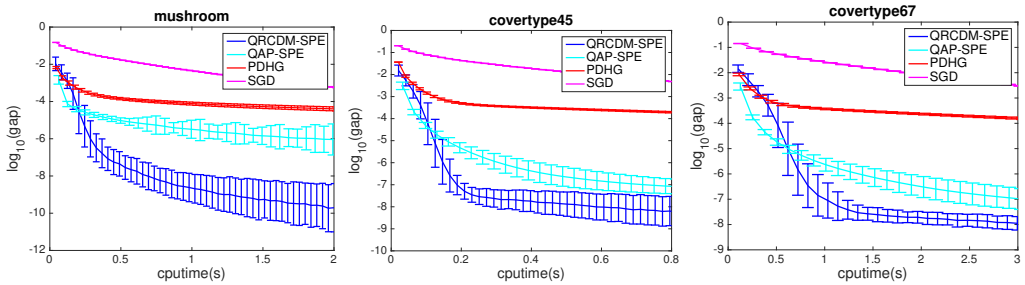


Figure 6.4: Convergence of different solvers for QDFSM over three different real datasets.

6.5.3 Real data

We also evaluated the proposed algorithms on three UCI datasets: *Mushroom*, *Covertype45*, *Covertype67*, used as standard test datasets for SSL on hypergraphs [7, 9, 99]. Each dataset corresponds to a hypergraph model described in [9]: entries correspond to vertices, while each categorical feature is modeled as one hyperedge; numerical features are first quantized into 10 bins of equal size, and then mapped to hyperedges. Compared to synthetic data, in the real datasets, the size of most hyperedges is significantly larger (≥ 1000) while the number of hyperedges is small (≈ 100). See Table 6.3 for a detailed description of the parameters of the generated hypergraphs. Previous work has shown that smaller classification errors can be achieved by using QDSFM as an objective instead of DSFM or InvLap [9]. In our experiment, we focused on comparing the convergence of different solvers for QDSFM. We set $\beta = 100$ and $W_{ii} = 1$, for all i, and set the number of observed labels to 100, which is a proper setting according to [9]. Figure 6.4 shows the results. Again, the proposed QRCD-SPE and QAP-SPE methods

Table 6.3: The UCI datasets used for experimental testing.

Dataset	Mushroom	Covertype45	Covertype67
N	8124	12240	37877
R	112	127	136
$\sum_{r\in[R]} \lvert S_r \rvert$	170604	145999	451529

both converge faster than PDHG and SGD, while QRCD-SPE performs the best. Note that we did not plot the results for QRCD-MNP and QRCD-FW as these methods converge extremely slowly due to the large sizes of the hyperedges. InvLap requires 22, 114 and 1802 seconds to run on the Mushroom, Covertype45 and Covertype67 datasets, respectively. Hence, the method does not scale well with the problem size.

APPENDIX A

SUPPLEMENTARY PROOFS, DERIVATIONS AND TABLES

A.1 Proof of some preliminary results

We find the following properties of the Lovász extension of normalized symmetric submodular functions useful in the derivations to follow.

Lemma A.1.1. *Consider two vectors $x, x' \in \mathbb{R}^N$. If F is a symmetric submodular function with $F([N]) = 0$, and $f(x)$ is the corresponding Lovász extension, then for any scalar $c \in \mathbb{R}$,*

1) $f(cx) = |c|f(x)$.

2) $\nabla f(cx) = sgn(c)\nabla f(x)$, *where sgn denotes the sign function defined in the main text.*

3) $\langle \nabla f(x), \mathbf{1} \rangle = 0$.

Proof. Given the definition of the Lovász extension and its subgradient, for any $c > 0$ we have $f(cx) = cf(x)$ and $\nabla f(cx) = \nabla f(x)$. As F is a symmetric submodular function, $f(x) = f(-x)$ is even, which establishes the first claim. Also, since $f(x)$ is even, $\nabla f(x)$ is odd, and thus, for some $c < 0$, we have $\nabla f(cx) = \nabla f((-c) - x) = \nabla f(-x) = -\nabla f(x)$. For $c = 0$, $\nabla f(0) = \mathcal{B} = [-1, 1]\mathcal{B} = \{ab : a \in [-1, 1], b \in \mathcal{B}\}$, since F is a symmetric submodular function. Hence, the second claim holds as well. The third claim follows from $\langle \nabla f(x), \mathbf{1} \rangle = F([N]) = 0$. □

Definition A.1.2. *Let $x, x' \in \mathbb{R}^N$. If $x_u > x_v \Rightarrow x'_u > x'_v$ for all $u, v \in [N]$, we write $x \rightharpoonup x'$.*

Lemma A.1.3. *Assume that F is a submodular function defined on $[N]$ and that f is its corresponding Lovász extension. If $x \rightharpoonup x'$, then $\nabla f(x') \subseteq \nabla f(x)$. Furthermore, $\langle \nabla f(x'), x \rangle = f(x)$.*

Proof. Consider a point $y' \in \nabla f(x')$. According to Lemma 2.1, we know that $y' \in \arg\max_{y\in B}\langle y, x'\rangle$. Suppose that a nonincreasing order of components in x' reads as $x'_{i_1} \geq x'_{i_2} \geq \cdots \geq x'_{i_N}$. By the duality result of Proposition 3.2 in [28], it is known that y' is an optimal solution to the above optimization problem if and only if $\sum_{j=1}^{k} y'_{i_j} = F(\{i_1, ..., i_k\})$ whenever $x'_{i_k} > x'_{i_{k+1}}$ or $k = N$. As $x \rightharpoonup x'$, $\sum_{j=1}^{k} y'_{i_j} = F(\{i_1, ..., i_k\})$ whenever $x_{i_k} > x_{i_{k+1}}$ or $k = N$, and thus y' is also an optimal solution for $\max_{y\in\mathcal{B}}\langle y, x\rangle$, i.e., $\nabla f(x') \subseteq \nabla f(x)$. Hence, $\langle \nabla f(x'), x\rangle \in \langle \nabla f(x), x\rangle = f(x)$, which concludes the proof. □

A.2 Proof for Chapter 3

A.2.1 Proof of Theorem 3.1.1

Here, we use $\mathcal{G}$ as the subscript to denote the variables that are associated with the projected graph $\mathcal{G}$. First, since $\mathrm{Vol}(\partial S) \leq \mathrm{Vol}_{\mathcal{G}}(\partial S)$ and $\mathrm{Vol}(S) = \mathrm{Vol}_{\mathcal{G}}(S)$ for any $S \subseteq V$, we have

$$c(S^*) = \frac{\mathrm{Vol}(S^*)}{\max\{\mathrm{Vol}(S^*), \mathrm{Vol}(\bar{S}^*)\}} \leq \frac{\mathrm{Vol}_{\mathcal{G}}(S^*)}{\max\{\mathrm{Vol}(S^*), \mathrm{Vol}(\bar{S}^*)\}} = c_{\mathcal{G}}(S^*).$$

Moreover, as $\mathrm{Vol}_{\mathcal{G}}(\partial S) \leq \beta^* \mathrm{Vol}(\partial S)$, we have

$$\begin{aligned} h_{2,\mathcal{G}} &= \min_{S\subseteq V} c_{\mathcal{G}}(S) \leq \min_{S\subseteq V} \beta^* c(S) = \beta^* h_2 \\ d_{u,\mathcal{G}} &= \sum_{e\in E} \sum_{v\in e/\{u\}} \vartheta_e w_{uv}^{(e)} \leq \sum_{e\in E} \beta^* \vartheta_e w_e(\{u\}) \leq \beta^* d_u. \end{aligned}$$

Further, combining the above results with those that characterize the spectral partitioning algorithm for graphs (2.2)(2.3), we have

$$c(S^*) \leq c_{\mathcal{G}}(S^*) \leq 2\sqrt{\tau_{\mathcal{G}} h_{2,\mathcal{G}}} = 2\sqrt{\max_{u\in V} \frac{d_{u,\mathcal{G}}}{\mu_u} h_{2,\mathcal{G}}} \leq 2\beta^* \sqrt{\max_{u\in V} \frac{d_u}{\mu_u} h_2} = 2\beta^* \sqrt{\tau h_2}.$$

A.2.2 Proof of Theorem 3.1.2

First, recall that $w_{v\tilde{v}}^{*(e)}$ denotes the projection weight of equation (3.7), for the case that $w_e(\cdot)$ is submodular:

$$w_{v\tilde{v}}^{*(e)} = \sum_{S \in 2^e/\{\emptyset,e\}} \left[\frac{w_e(S)}{2|S|(|e|-|S|)} 1_{|\{v,\tilde{v}\}\cap S|=1} \right. \tag{A.1}$$
$$\left. - \frac{w_e(S)}{2(|S|+1)(|e|-|S|-1)} 1_{|\{v,\tilde{v}\}\cap S|=0} - \frac{w_e(S)}{2(|S|-1)(|e|-|S|+1)} 1_{|\{v,\tilde{v}\}\cap S|=2} \right].$$

We start by proving that for a fixed pair of vertices v and $\tilde{v}$, the weights $w_{v\tilde{v}}^{*(e)}$ are nonnegative provided that the $w_e(\cdot)$ are submodular. Note that the sum on the right-hand side of (A.1) is over all proper subsets S. The coefficients of $w_e(S)$ are positive if and only if S contains exactly one of the endpoints v and $\tilde{v}$. The idea behind the proof is to construct bijections between the subsets with positive coefficients and those with negative coefficients and cancel negative and positive terms.

We partition the power set 2^e into four parts, namely

$$\mathbb{S}_1 \triangleq \{S \in 2^e : v \in S, \tilde{v} \notin S\},$$
$$\mathbb{S}_2 \triangleq \{S \in 2^e : v \notin S, \tilde{v} \in S\},$$
$$\mathbb{S}_3 \triangleq \{S \in 2^e : v \notin S, \tilde{v} \notin S\},$$
$$\mathbb{S}_4 \triangleq \{S \in 2^e : v \in S, \tilde{v} \in S\}.$$

Choose any $S_1 \in \mathbb{S}_1$ and construct the unique sets $S_2 = S_1/\{v\} \cup \{\tilde{v}\} \in \mathbb{S}_2$, $S_3 = S_1/\{v\} \in \mathbb{S}_3$, $S_4 = S_1 \cup \{\tilde{v}\} \in \mathbb{S}_4$. Consequently, each set may be reconstructed from another set in the group, and we denote this set of bijective relations by $S_1 \leftrightarrow S_2 \leftrightarrow S_3 \leftrightarrow S_4$. Let $s = |S_1|$. Due to the way the sets S_1 and S_2 are chosen, the corresponding coefficients of $w_e(S_1)$ and $w_e(S_2)$ in (3.7) are both equal to

$$\frac{1}{2s(|e|-s)}.$$

We also observe that the corresponding coefficients of $w_e(S_3)$ and $w_e(S_4)$ are

$$-\frac{1}{2s(|e|-s)}.$$

Note that the submodularity property

$$w_e(S_1) + w_e(S_2) \geq w_e(S_3) + w_e(S_4)$$

allows us to cancel out the negative terms in the sum (A.1). This proves the claimed result.

Next, we prove that the optimization problem (3.2) has a feasible solution.

Recall that $G_e = (V^{(e)}, E^{(e)}, w^{(e)})$ is the subgraph obtained by projecting e. Set $w^{(e)} = w^{*(e)}$. For simplicity of notation, we denote the volume of the boundary of S over G_e as

$$\mathrm{Vol}_{G_e}(\partial S) = \sum_{v \in S, \tilde{v} \in e/S} w^{(e)}_{v\tilde{v}}, \qquad \text{for } S \in 2^e.$$

The existence of a feasible solution of the optimization problem may be verified by checking that for any $S \in 2^e/\{\emptyset, e\}$, and for a given $\beta^{(e)}$, we have the following bounds on the volume of the boundary of S:

$$w_e(S) \leq \mathrm{Vol}_{G_e}(\partial S) \leq \beta^{(e)} w_e(S).$$

Due to symmetry, we only need to perform the verification for sets S of different cardinalities $|S| \leq |e|/2$. This verification is performed on a case-by-case bases, as we could not establish a general proof for arbitrary degree $|e| \geq 2$. In what follows, we show that the claim holds true for all $|e| \leq 7$; based on several special cases considered, we conjecture that the result is also true for all values of $|e|$ greater than seven.

For notational simplicity, we henceforth assume that the vertices in e are labeled by elements in $\{1, 2, 3, ..., |e|\}$.

First, note that by combining symmetry and submodularity, we can easily show that

$$\begin{aligned} &w_e(S_1) + w_e(S_2) = w_e(S_1) + w_e(\bar{S_2}) \\ \geq &w_e(S_1 \cup \bar{S_2}) + w_e(S_1 \cap \bar{S_2}) = w_e(S_2/S_1) + w_e(S_1/S_2). \end{aligned}$$

We iteratively use this equality in our subsequent proofs, following a specific

notational format for all relevant inequalities:

$$v_{i_1}, ..., v_{i_r} \in S_1,\ v_{j_1}, ..., v_{j_s} \in S_2,\ \textit{Weight inequality} \implies \textit{Volume inequality}.$$

The above line asserts that for all ordered subsets $(v_{i_1}, ..., v_{i_r})$ and $(v_{j_1}, ..., v_{j_s})$ chosen from S_1 and S_2 without replacement, respectively, we have that the *Weight inequalities* follow based on the properties of $w_e(\cdot)$. These *Weight inequalities* are consequently inserted into the formula for the volume $\text{Vol}_{G_e}(S)$ to arrive at the *Volume inequality* for $\text{Vol}_{G_e}(S)$.

For $|e| = 2$, the projection (A.1) is just a "self-projection": It is easy to check that for any singleton S, $\text{Vol}_{G_e}(\partial S) = w_e(S)$ and hence $\beta^{(e)} = 1$. We next establish the same claim for larger hyperedge sizes $|e|$.

$|e| = 3$, $\beta^{(e)} = 1$

By using the symmetry property of $w_e(\cdot)$ we have

$$w_{12}^{*(e)} = \frac{1}{2}(w_e(\{1\}) + w_e(\{2\})) - w_e(\{3\}).$$

Therefore, $\text{Vol}_{G_e}(\partial\{1\}) = w_{12}^{*(e)} + w_{13}^{*(e)} = w_e(\{1\})$ and hence $\beta^{(e)} = 1$.

$|e| = 4$, $\beta^{(e)} = 3/2$

By using the symmetry property of $w_e(\cdot)$ we have

$$\begin{aligned} & w_{12}^{*(e)} \\ =& \frac{1}{3}(w_e(\{1\}) + w_e(\{2\})) - \frac{1}{4}(w_e(\{3\}) + w_e(\{4\})) \\ +& \frac{1}{4}w_e(\{1,3\}) + w_e(\{1,4\})) - \frac{1}{3}w_e(\{1,2\}). \end{aligned}$$

The basic idea behind the proof of the equalities to follow is to carefully select subsets for which the submodular inequality involving $w_e(\cdot)$ may be used to eliminate the terms corresponding to the volumes $\text{Vol}_{G_e}(\partial S)$.

Case $S = \{1\}$:

$$\begin{aligned}&\mathrm{Vol}_{G_e}(\partial\{1\})\\ =&w_e(\{1\}) - \frac{1}{6}(w_e(\{2\}) + w_e(\{3\}) + w_e(\{4\}))\\ +&\frac{1}{6}(w_e(\{1,2\}) + w_e(\{1,3\}) + w_e(\{1,4\}))\end{aligned}$$

$$v_1 = 1,\ v_2, v_3 \in \{2,3,4\},\ w_e(\{v_1,v_2\}) + w_e(\{v_1,v_3\}) \geq w_e(\{v_2\}) + w_e(\{v_3\})$$
$$\implies \quad \mathrm{Vol}_{G_e}(\partial\{1\}) \geq w_e(\{1\}).$$

$$v_1 = 1,\ v_2 \in \{2,3,4\},\ w_e(\{v_1,v_2\}) \leq w_e(\{v_1\}) + w_e(\{v_2\})$$
$$\implies \quad \mathrm{Vol}_{G_e}(\partial\{1\}) \leq \frac{3}{2} w_e(\{1\}).$$

Case $S = \{1,2\}$:

$$\begin{aligned}&\mathrm{Vol}_{G_e}(\partial\{1,2\})\\ =&\frac{1}{6}(w_e(\{1\}) + w_e(\{2\}) + w_e(\{3\}) + w_e(\{4\}))\\ +&w_e(\{1,2\}) - \frac{1}{6}(w_e(\{1,3\}) + w_e(\{1,4\}))\end{aligned}$$

$$v_1 \in \{1,2\},\ v_2 \in \{3,4\},\ w_e(\{v_1\}) + w_e(\{v_2\}) \geq w_e(\{v_1,v_2\})$$
$$\implies \quad \mathrm{Vol}_{G_e}(\partial\{1,2\}) \geq w_e(\{1,2\}).$$

$$v_1, v_2 \in \{1,2\}, v_3 \in \{3,4\},\ w_e(\{v_1\}) + w_e(\{v_3\}) \leq w_e(\{v_1,v_2\}) + w_e(\{v_2,v_3\})$$
$$\implies \quad \mathrm{Vol}_{G_e}(\partial\{1,2\}) \leq \frac{4}{3} w_e(\{1,2\}).$$

$|e| = 5,\ \beta^{(e)} = 2$

By using the symmetry property of $w_e(\cdot)$ we have

$$\begin{aligned}
&w_{12}^{*(e)}\\
=&\frac{1}{4}(w_e(\{1\}) + w_e(\{2\})) - \frac{1}{6}(w_e(\{3\}) + w_e(\{4\}) + w_e(\{5\})) - \frac{1}{4}w_e(\{1,2\})\\
+&\frac{1}{6}(w_e(\{1,3\}) + w_e(\{1,4\}) + w_e(\{1,5\}) + w_e(\{2,3\})\\
+&w_e(\{2,4\}) + w_e(\{2,5\})) - \frac{1}{6}(w_e(\{3,4\}) + w_e(\{3,5\}) + w_e(\{4,5\})).
\end{aligned}$$

Case $S = \{1\}$:

$$\begin{aligned}
&\mathrm{Vol}_{G_e}(\partial\{1\})\\
=&w_e(\{1\}) - \frac{1}{4}(w_e(\{2\}) + w_e(\{3\}) + w_e(\{4\}) + w_e(\{5\}))\\
+&\frac{1}{4}(w_e(\{1,2\}) + w_e(\{1,3\}) + w_e(\{1,4\}) + w_e(\{1,5\})).
\end{aligned}$$

$$v_1 = 1,\ v_2, v_3 \in \{2,3,4,5\},\ w_e(\{v_1,v_2\}) + w_e(\{v_1,v_3\}) \geq w_e(\{v_2\}) + w_e(\{v_3\}) \quad \Longrightarrow \quad \mathrm{Vol}_{G_e}(\partial\{1\}) \geq w_e(\{1\}).$$

$$v_1 = 1,\ v_2 \in \{2,3,4,5\},\ w_e(\{v_1,v_2\}) \leq w_e(\{v_1\}) + w_e(\{v_2\}) \quad \Longrightarrow \quad \mathrm{Vol}_{G_e}(\partial\{1\}) \leq 2w_e(\{1\}).$$

Case $S = \{1,2\}$:

$$\begin{aligned}
&\mathrm{Vol}_{G_e}(\partial\{1,2\})\\
=&\frac{1}{4}(w_e(\{1\}) + w_e(\{2\})) - \frac{1}{6}(w_e(\{3\}) + w_e(\{4\}) + w_e(\{5\})) + w_e(\{1,2\})\\
-&\frac{1}{12}(w_e(\{1,3\}) + w_e(\{1,4\}) + w_e(\{1,5\}) + w_e(\{2,3\}) + w_e(\{2,4\}) + w_e(\{2,5\}))\\
+&\frac{1}{3}(w_e(\{3,4\}) + w_e(\{3,5\}) + w_e(\{4,5\})).
\end{aligned}$$

$$v_1, v_2, v_3 \in \{3,4,5\},\ w_e(\{v_2\}) + w_e(\{v_3\}) \leq w_e(\{v_1, v_2\}) + w_e(\{v_1, v_3\})$$
$$v_1 \in \{1,2\},\ v_2 \in \{3,4,5\},\ w_e(\{v_1, v_2\}) \leq w_e(\{v_1\}) + w_e(\{v_2\})$$
$$\Longrightarrow \quad \mathrm{Vol}_{G_e}(\partial\{1,2\}) \geq w_e(\{1,2\}).$$

$$v_1 \in \{1,2\},\ v_2, v_3 \in \{3,4,5\},\ w_e(\{v_1, v_2\}) + w_e(\{v_1, v_3\}) \geq w_e(\{v_2\}) + w_e(\{v_3\})$$
$$v_1, v_2 \in \{1,2\},\ v_3, v_4, v_5 \in \{3,4,5\},\ w_e(\{v_3, v_4\}) \leq w_e(\{v_1, v_2\}) + w_e(\{v_5\})$$
$$\Longrightarrow \quad \mathrm{Vol}_{G_e}(\partial\{1,2\}) \leq 2w_e(\{1,2\}).$$

$|e| = 6,\ \beta^{(e)} = 4$

By using the symmetry property of $w_e(\cdot)$ we have

$$\begin{aligned}
& w_{12}^{*(e)} \\
=& \frac{1}{5}(w_e(\{1\}) + w_e(\{2\})) - \frac{1}{8}(w_e(\{3\}) + w_e(\{4\}) + w_e(\{5\}) + w_e(\{6\})) \\
& \qquad - \frac{1}{5}w_e(\{1,2\}) \\
+& \frac{1}{8}(w_e(\{1,3\}) + w_e(\{1,4\}) + w_e(\{1,5\}) + w_e(\{1,6\}) + w_e(\{2,3\}) + w_e(\{2,4\}) \\
& \qquad + w_e(\{2,5\}) + w_e(\{2,6\})) \\
-& \frac{1}{9}(w_e(\{3,4\}) + w_e(\{3,5\}) + w_e(\{3,6\}) + w_e(\{4,5\}) + w_e(\{4,6\}) + w_e(\{5,6\})) \\
-& \frac{1}{9}(w_e(\{1,2,3\}) + w_e(\{1,2,4\}) + w_e(\{1,2,5\}) + w_e(\{1,2,6\})) \\
+& \frac{1}{8}(w_e(\{1,3,4\}) + w_e(\{1,3,5\}) + w_e(\{1,3,6\}) + w_e(\{1,4,5\}) \\
& \qquad + w_e(\{1,4,6\}) + w_e(\{1,5,6\})).
\end{aligned}$$

Case $S = \{1\}$:

$$
\begin{aligned}
&\mathrm{Vol}_{G_e}(\partial\{1\}) \\
=&w_e(\{1\}) - \frac{3}{10}(w_e(\{2\}) + w_e(\{3\}) + w_e(\{4\}) + w_e(\{5\}) + w_e(\{6\})) \\
+&\frac{3}{10}(w_e(\{1,2\}) + w_e(\{1,3\}) + w_e(\{1,4\}) + w_e(\{1,5\}) + w_e(\{1,6\})) \\
-&\frac{1}{12}(w_e(\{2,3\}) + w_e(\{2,4\}) + w_e(\{2,5\}) + w_e(\{2,6\}) + w_e(\{3,4\}) \\
&\qquad + w_e(\{3,5\}) + w_e(\{3,6\}) + w_e(\{4,5\}) + w_e(\{4,6\}) + w_e(\{5,6\})) \\
+&\frac{1}{12}(w_e(\{1,2,3\}) + w_e(\{1,2,4\}) + w_e(\{1,2,5\}) + w_e(\{1,2,6\}) \\
&\qquad + w_e(\{1,3,4\}) + w_e(\{1,3,5\}) + w_e(\{1,3,6\}) \\
&\qquad + w_e(\{1,4,5\}) + w_e(\{1,4,6\}) + w_e(\{1,5,6\}))
\end{aligned}
$$

$$
\begin{aligned}
&v_1 = 1,\ v_2, v_3 \in \{2,3,4,5,6\},\ w_e(\{v_1, v_2\}) + w_e(\{v_1, v_3\}) \geq w_e(\{v_2\}) + w_e(\{v_3\}) \\
&v_1 = 1,\ v_2, v_3, v_4, v_5 \in \{2,3,4,5,6\}, \\
&\qquad\qquad w_e(\{v_1, v_2, v_3\}) + w_e(\{v_1, v_4, v_5\}) \geq w_e(\{v_2, v_3\}) + w_e(\{v_4, v_5\}) \\
&\qquad\qquad\qquad \Longrightarrow \mathrm{Vol}_{G_e}(\partial\{1\}) \geq w_e(\{1\}).
\end{aligned}
$$

$$
\begin{aligned}
&v_1 = 1,\ v_2 \in \{2,3,4,5,6\},\ w_e(\{v_1, v_2\}) \leq w_e(\{v_1\}) + w_e(\{v_2\}) \\
&v_1 = 1,\ v_2, v_3 \in \{2,3,4,5,6\},\ w_e(\{v_1, v_2, v_3\}) \leq w_e(\{v_1\}) + w_e(\{v_2, v_3\}) \\
&\qquad\qquad\qquad \Longrightarrow \quad \mathrm{Vol}_{G_e}(\partial\{1\}) \leq \frac{10}{3} w_e(\{1\}).
\end{aligned}
$$

Case $S = \{1, 2\}$:

$$
\begin{aligned}
&\mathrm{Vol}_{G_e}(\partial\{1,2\}) \\
=&\frac{3}{10}(w_e(\{1\}) + w_e(\{2\})) - \frac{7}{20}(w_e(\{3\}) + w_e(\{4\}) + w_e(\{5\}) + w_e(\{6\})) \\
+&w_e(\{1,2\}) - \frac{1}{30}(w_e(\{1,3\}) + w_e(\{1,4\}) + w_e(\{1,5\}) + w_e(\{1,6\}) + w_e(\{2,3\}) \\
&\qquad + w_e(\{2,4\}) + w_e(\{2,5\}) + w_e(\{2,6\})) \\
+&\frac{1}{18}(w_e(\{3,4\}) + w_e(\{3,5\}) + w_e(\{3,6\}) + w_e(\{4,5\}) + w_e(\{4,6\}) + w_e(\{5,6\})) \\
+&\frac{5}{12}(w_e(\{1,2,3\}) + w_e(\{1,2,4\}) + w_e(\{1,2,5\}) + w_e(\{1,2,6\})) \\
-&\frac{1}{18}(w_e(\{1,3,4\}) + w_e(\{1,3,5\}) + w_e(\{1,3,6\}) \\
&\qquad + w_e(\{1,4,5\}) + w_e(\{1,4,6\}) + w_e(\{1,5,6\}))
\end{aligned}
$$

$$
\begin{aligned}
&v_1 \in \{1,2\},\ v_2 \in \{3,4,5,6\},\ w_e(\{v_1,v_2\}) \le w_e(\{v_1\}) + w_e(\{v_2\}) \\
&v_1 = 1,\ v_2 = 2,\ v_3, v_4 \in \{3,4,5,6\}, \\
&\qquad w_e(\{v_3\}) + w_e(\{v_4\}) \le w_e(\{v_1,v_2,v_3\}) + w_e(\{v_1,v_2,v_4\}) \\
&v_1 = 1,\ v_2, v_3 \in \{3,4,5,6\}, \\
&\qquad w_e(\{v_1,v_2,v_3\}) \le w_e(\{v_1\}) + w_e(\{v_2,v_3\}) \\
&\qquad\qquad \Longrightarrow \quad \mathrm{Vol}_{G_e}(\partial\{1,2\}) \ge w_e(\{1,2\}).
\end{aligned}
$$

$$
\begin{aligned}
&v_1 = 1,\ v_2 = 2,\ v_3 \in \{3,4,5,6\},\ w_e(\{v_3\}) \ge w_e(\{v_1,v_2,v_3\}) - w_e(\{v_1,v_2\}) \\
&v_1,\ v_2 \in \{1,2\},\ v_3 \in \{3,4,5,6\}, \\
&\qquad w_e(\{v_1,v_3\}) \ge w_e(\{v_1\}) + w_e(\{v_1,v_2,v_3\}) - w_e(\{v_1,v_2\}) \\
&v_1,\ v_2 \in \{1,2\},\ v_3, v_4, v_5, v_6 \in \{3,4,5,6\}, \\
&\qquad w_e(\{v_1,v_3,v_4\}) \ge w_e(\{v_5,v_6\}) + w_e(\{v_1\}) - w_e(\{v_1,v_2\}) \\
&\qquad\qquad \Longrightarrow \quad \mathrm{Vol}_{G_e}(\partial\{1,2\}) \le 3w_e(\{1,2\}).
\end{aligned}
$$

Case $S = \{1, 2, 3\}$:

$$
\begin{aligned}
&\mathrm{Vol}_{G_e}(\partial\{1,2,3\}) \\
=&-\frac{3}{20}(w_e(\{1\}) + w_e(\{2\}) + w_e(\{3\}) + w_e(\{4\}) + w_e(\{5\}) + w_e(\{6\})) \\
&+\frac{5}{12}(w_e(\{1,2\}) + w_e(\{1,3\}) + w_e(\{2,3\} + w_e(\{4,5\}) + w_e(\{4,6\}) + w_e(\{5,6\})) \\
&-\frac{13}{90}(w_e(\{1,4\}) + w_e(\{1,5\}) + w_e(\{1,6\}) + w_e(\{2,4\}) + w_e(\{2,5\}) + w_e(\{2,6\}) \\
&\qquad + w_e(\{3,4\}) + w_e(\{3,5\}) + w_e(\{3,6\})) \\
&+w_e(\{1,2,3\}) + \frac{1}{18}(w_e(\{1,2,4\}) + w_e(\{1,2,5\}) + w_e(\{1,2,6\}) + w_e(\{1,3,4\}) \\
&\qquad + w_e(\{1,3,5\}) + w_e(\{1,3,6\}) + w_e(\{1,4,5\}) + w_e(\{1,4,6\}) + w_e(\{1,5,6\}))
\end{aligned}
$$

$$
\begin{aligned}
&v_1, v_2, v_3 \in \{1,2,3\},\ v_4, v_5, v_6 \in \{4,5,6\}, \\
&\qquad w_e(\{v_2\}) + w_e(\{v_3\}) \le w_e(\{v_1, v_2\}) + w_e(\{v_1, v_3\}), \\
&\qquad w_e(\{v_5\}) + w_e(\{v_6\}) \le w_e(\{v_4, v_5\}) + w_e(\{v_4, v_6\}) \\
&v_1, v_2, v_3 \in \{1,2,3\},\ v_4, v_5, v_6 \in \{4,5,6\}, \\
&\qquad w_e(\{v_1, v_2\}) + w_e(\{v_4, v_5\}) \le w_e(\{v_3, v_6\}) \\
&v_1, v_2, v_3 \in \{1,2,3\},\ v_4, v_5, v_6 \in \{4,5,6\}, \\
&\qquad w_e(\{v_1, v_2, v_4\}) + w_e(\{v_1, v_4, v_5\}) \le w_e(\{v_1, v_4\}) + w_e(\{v_3, v_6\}) \\
&\qquad\qquad\qquad\qquad \Longrightarrow \quad \mathrm{Vol}_{G_e}(\partial\{1,2,3\}) \ge w_e(\{1,2,3\}).
\end{aligned}
$$

$$
\begin{aligned}
&v_1, v_2 \in \{1,2,3\},\ v_4, v_5 \in \{4,5,6\},\\
&\qquad w_e(\{v_1, v_4, v_5\}) \le w_e(\{v_1, v_4\}) + w_e(\{v_1, v_5\}) - w_e(\{v_1\}),\\
&\qquad w_e(\{v_1, v_2, v_4\}) \le w_e(\{v_1, v_4\}) + w_e(\{v_2, v_4\}) - w_e(\{v_4\})\\
&v_1, v_2, v_3 \in \{1,2,3\},\ v_4, v_5, v_6 \in \{4,5,6\},\\
&\qquad w_e(\{v_1, v_4\}) \ge w_e(\{v_1\}) + w_e(\{v_5, v_6\}) - w_e(\{v_1, v_2, v_3\})\\
&v_1, v_2, v_3 \in \{1,2,3\},\ v_4, v_5, v_6 \in \{4,5,6\},\\
&\qquad w_e(\{v_1\}) + w_e(\{v_2\}) \ge w_e(\{v_1, v_2\}),\\
&\qquad w_e(\{v_1\}) \ge w_e(\{v_2, v_3\}) - w_e(\{v_1, v_2, v_3\}),\\
&\qquad w_e(\{v_4\}) + w_e(\{v_5\}) \ge w_e(\{v_4, v_5\}),\\
&\qquad w_e(\{v_4\}) \ge w_e(\{v_5, v_6\}) - w_e(\{v_4, v_5, v_6\})\\
&\qquad\qquad\qquad\qquad \Longrightarrow \quad \mathrm{Vol}_{G_e}(\partial\{1,2,3\}) \le 4 w_e(\{1,2,3\}).
\end{aligned}
$$

$|e| = 7$, $\beta^{(e)} = 6$

By using the symmetry property of $w_e(\cdot)$ we have

$$
\begin{aligned}
& w_{12}^{*(e)} \\
=& \frac{1}{6}(w_e(\{1\}) + w_e(\{2\})) - \frac{1}{10}(w_e(\{3\}) + w_e(\{4\}) + w_e(\{5\}) + w_e(\{6\}) + w_e(\{7\})) \\
& - \frac{1}{6} w_e(\{1,2\}) + \frac{1}{10}(w_e(\{1,3\}) + w_e(\{1,4\}) + w_e(\{1,5\}) + w_e(\{1,6\}) + w_e(\{1,7\}) \\
& \qquad + w_e(\{2,3\}) + w_e(\{2,4\}) + w_e(\{2,5\}) + w_e(\{2,6\}) + w_e(\{2,7\})) \\
& - \frac{1}{12}(w_e(\{3,4\}) + w_e(\{3,5\}) + w_e(\{3,6\}) + w_e(\{3,7\}) + w_e(\{4,5\}) + w_e(\{4,6\}) \\
& \qquad + w_e(\{4,7\}) + w_e(\{5,6\}) + w_e(\{5,7\}) + w_e(\{6,7\})) \\
& - \frac{1}{10}(w_e(\{1,2,3\}) + w_e(\{1,2,4\}) + w_e(\{1,2,5\}) + w_e(\{1,2,6\}) + w_e(\{1,2,7\})) \\
& + \frac{1}{12}(w_e(\{1,3,4\}) + w_e(\{1,3,5\}) + w_e(\{1,3,6\}) + w_e(\{1,3,7\}) + w_e(\{1,4,5\}) \\
& \qquad + w_e(\{1,4,6\}) + w_e(\{1,4,7\}) + w_e(\{1,5,6\}) + w_e(\{1,5,7\}) + w_e(\{1,6,7\}) \\
& \qquad + w_e(\{2,3,4\}) + w_e(\{2,3,5\}) + w_e(\{2,3,6\}) + w_e(\{2,3,7\}) + w_e(\{2,4,5\}) \\
& \qquad + w_e(\{2,4,6\}) + w_e(\{2,4,7\}) + w_e(\{2,5,6\}) + w_e(\{2,5,7\}) + w_e(\{2,6,7\})) \\
& - \frac{1}{12}(w_e(\{3,4,5\}) + w_e(\{3,4,6\}) + w_e(\{3,4,7\}) \\
& \qquad + w_e(\{3,5,6\}) + w_e(\{3,5,7\}) + w_e(\{3,6,7\}) \\
& \qquad + w_e(\{4,5,6\}) + w_e(\{4,5,7\}) + w_e(\{4,6,7\}) + w_e(\{5,6,7\})).
\end{aligned}
$$

Case $S = \{1\}$:

$$
\begin{aligned}
&\mathrm{Vol}_{G_e}(\partial\{1\}) \\
=&w_e(\{1\}) - \frac{1}{3}(w_e(\{2\}) + w_e(\{3\}) + w_e(\{4\}) + w_e(\{5\}) + w_e(\{6\}) + w_e(\{7\})) \\
+&\frac{1}{3}(w_e(\{1,2\}) + w_e(\{1,3\}) + w_e(\{1,4\}) + w_e(\{1,5\}) + w_e(\{1,6\}) + w_e(\{1,7\})) \\
-&\frac{2}{15}(w_e(\{2,3\}) + w_e(\{2,4\}) + w_e(\{2,5\}) + w_e(\{2,6\}) + w_e(\{2,7\}) + w_e(\{3,4\}) \\
&\quad + w_e(\{3,5\}) + w_e(\{3,6\}) + w_e(\{3,7\}) + w_e(\{4,5\}) + w_e(\{4,6\}) \\
&\quad + w_e(\{4,7\}) + w_e(\{5,6\}) + +w_e(\{5,7\}) + w_e(\{6,7\})) \\
+&\frac{2}{15}(w_e(\{1,2,3\}) + w_e(\{1,2,4\}) + w_e(\{1,2,5\}) \\
&\quad\quad + w_e(\{1,2,6\}) + w_e(\{1,2,7\}) + w_e(\{1,3,4\}) \\
&\quad + w_e(\{1,3,5\}) + w_e(\{1,3,6\}) + w_e(\{1,3,7\}) + w_e(\{1,4,5\}) + w_e(\{1,4,6\}) \\
&\quad + w_e(\{1,4,7\}) + w_e(\{1,5,6\})) + w_e(\{1,5,7\}) + +w_e(\{1,6,7\})
\end{aligned}
$$

$$
\begin{aligned}
&v_1 = 1,\ v_2, v_3 \in \{2,3,4,5,6,7\}, \\
&\qquad w_e(\{v_1, v_2\}) + w_e(\{v_1, v_3\}) \geq w_e(\{v_2\}) + w_e(\{v_3\}) \\
&v_1 = 1,\ v_2, v_3, v_4, v_5 \in \{2,3,4,5,6,7\}, \\
&\qquad w_e(\{v_1, v_2, v_3\}) + w_e(\{v_1, v_4, v_5\}) \geq w_e(\{v_2, v_3\}) + w_e(\{v_4, v_5\}) \\
&\qquad\qquad \Longrightarrow \mathrm{Vol}_{G_e}(\partial\{1\}) \geq w_e(\{1\}).
\end{aligned}
$$

$$
\begin{aligned}
&v_1 = 1,\ v_2 \in \{2,3,4,5,6,7\},\ w_e(\{v_1, v_2\}) \leq w_e(\{v_1\}) + w_e(\{v_2\}) \\
&v_1 = 1,\ v_2, v_3 \in \{2,3,4,5,6,7\},\ w_e(\{v_1, v_2, v_3\}) \leq w_e(\{v_1\}) + w_e(\{v_2, v_3\}) \\
&\qquad\qquad \Longrightarrow \quad \mathrm{Vol}_{G_e}(\partial\{1\}) \leq 5 w_e(\{1\}).
\end{aligned}
$$

Case $S = \{1, 2\}$:

$$
\begin{aligned}
&\mathrm{Vol}_{G_e}(\partial\{1,2\}) \\
=&\frac{1}{3}(w_e(\{1\}) + w_e(\{2\})) - \frac{7}{15}(w_e(\{3\}) + w_e(\{4\}) + w_e(\{5\}) + w_e(\{6\}) + w_e(\{7\})) \\
+&w_e(\{1,2\}) - \frac{1}{10}(w_e(\{3,4\}) + w_e(\{3,5\}) + w_e(\{3,6\}) + w_e(\{3,7\}) + w_e(\{4,5\}) \\
&\quad + w_e(\{4,6\}) + w_e(\{4,7\}) + w_e(\{5,6\}) + w_e(\{5,7\}) + w_e(\{6,7\})) \\
+&\frac{7}{15}(w_e(\{1,2,3\}) + w_e(\{1,2,4\}) + w_e(\{1,2,5\}) + w_e(\{1,2,6\} + w_e(\{1,2,7\})) \\
-&\frac{1}{30}(w_e(\{1,3,4\}) + w_e(\{1,3,5\}) + w_e(\{1,3,6\}) + w_e(\{1,3,7\}) + w_e(\{1,4,5\}) \\
&\quad + w_e(\{1,4,6\}) + w_e(\{1,4,7\}) + w_e(\{1,5,6\}) + w_e(\{1,5,7\}) + w_e(\{1,6,7\}) \\
&\quad + w_e(\{2,3,4\}) + w_e(\{2,3,5\}) + w_e(\{2,3,6\}) + w_e(\{2,3,7\}) + w_e(\{2,4,5\}) \\
&\quad + w_e(\{2,4,6\}) + w_e(\{2,4,7\}) + w_e(\{2,5,6\}) + w_e(\{2,5,7\}) + w_e(\{2,6,7\})) \\
+&\frac{1}{6}(w_e(\{3,4,5\}) + w_e(\{3,4,6\}) + w_e(\{3,4,7\}) \\
&\quad + w_e(\{3,5,6\}) + w_e(\{3,5,7\}) + w_e(\{3,6,7\}) \\
&\quad + w_e(\{4,5,6\}) + w_e(\{4,5,7\}) + w_e(\{4,6,7\}) + w_e(\{5,6,7\}))
\end{aligned}
$$

$$
\begin{aligned}
&v_1, v_2 \in \{1,2\},\ v_3, v_4, v_5, v_6, v_7 \in \{3,4,5,6,7\}, \\
&\qquad w_e(\{v_2, v_6, v_7\}) \le w_e(\{v_1\}) + w_e(\{v_3, v_4, v_5\}) \\
&v_1 = 1, v_2 = 2, v_3, v_4, v_5, v_6, v_7 \in \{3,4,5,6,7\}, \\
&\qquad w_e(\{v_6, v_7\}) \le w_e(\{v_1, v_2, v_3\}) + w_e(\{v_3, v_4, v_5\}) - w_e(\{v_3\}) \\
&v_1 = 1, v_2 = 2, v_3, v_4 \in \{3,4,5,6,7\}, \\
&\qquad w_e(\{v_3\}) + w_e(\{v_4\}) \le w_e(\{v_1, v_2, v_3\}) + w_e(\{v_1, v_2, v_4\}) \\
&\qquad\qquad\qquad\qquad \Longrightarrow \quad \mathrm{Vol}_{G_e}(\partial\{1,2\}) \ge w_e(\{1,2\}).
\end{aligned}
$$

$$
\begin{aligned}
&v_1 = 1,\ v_2 = 2,\ v_3 \in \{3,4,5,6\},\\
&\qquad w_e(\{v_3\}) \geq w_e(\{v_1,v_2,v_3\}) - w_e(\{v_1,v_2\})\\
&v_1 = 1,\ v_2 = 2,\ v_3,v_4,v_5,v_6,v_7 \in \{3,4,5,6\},\\
&\qquad w_e(\{v_3,v_4\}) \geq w_e(\{v_5,v_6,v_7\}) - w_e(\{v_1,v_2\})\\
&v_1,\ v_2 \in \{1,2\},\ v_3,v_4,v_5,v_6,v_7 \in \{3,4,5,6\},\\
&\qquad w_e(\{v_1,v_3,v_4\}) \geq w_e(\{v_5,v_6,v_7\}) + w_e(\{v_1\}) - w_e(\{v_1,v_2\})\\
&\qquad\qquad\qquad\qquad \Longrightarrow \quad \mathrm{Vol}_{G_e}(\partial\{1,2\}) \leq 5w_e(\{1,2\}).
\end{aligned}
$$

Case $S = \{1,2,3\}$:

$$
\begin{aligned}
&\mathrm{Vol}_{G_e}(\partial\{1,2,3\})\\
=&-\frac{2}{15}(w_e(\{1\}) + w_e(\{2\}) + w_e(\{3\}))\\
&-\frac{2}{5}(w_e(\{4\}) + w_e(\{5\}) + w_e(\{6\}) + w_e(\{7\}))\\
&+\frac{7}{15}(w_e(\{1,2\}) + w_e(\{1,3\}) + w_e(\{2,3\})\\
&-\frac{1}{6}(w_e(\{1,4\}) + w_e(\{1,5\}) + w_e(\{1,6\}) + w_e(\{1,7\}) + w_e(\{2,4\}) + w_e(\{2,5\})\\
&\quad + w_e(\{2,6\}) + w_e(\{2,7\}) + w_e(\{3,4\}) + w_e(\{3,5\}) + w_e(\{3,6\}) + w_e(\{3,7\}))\\
&+\frac{1}{10}(w_e(\{4,5\}) + w_e(\{4,6\}) + w_e(\{4,7\}) + w_e(\{5,6\}) + w_e(\{5,7\}) + w_e(\{6,7\}))\\
&+w_e(\{1,2,3\}) + \frac{2}{15}(w_e(\{1,2,4\}) + w_e(\{1,2,5\}) + w_e(\{1,2,6\}) + w_e(\{1,2,7\})\\
&\quad + w_e(\{1,3,4\}) + w_e(\{1,3,5\}) + w_e(\{1,3,6\}) + w_e(\{1,3,7\})\\
&\quad + w_e(\{2,3,4\}) + w_e(\{2,3,5\}) + w_e(\{2,3,6\}) + w_e(\{2,3,7\})\\
&-\frac{1}{30}(w_e(\{1,4,5\}) + w_e(\{1,4,6\}) + w_e(\{1,4,7\})\\
&\quad + w_e(\{1,5,6\}) + w_e(\{1,5,7\}) + w_e(\{1,6,7\})\\
&\quad + w_e(\{2,4,5\}) + w_e(\{2,4,6\}) + w_e(\{2,4,7\})\\
&\quad + w_e(\{2,5,6\}) + w_e(\{2,5,7\}) + w_e(\{2,6,7\})\\
&\quad + w_e(\{3,4,5\}) + w_e(\{3,4,6\}) + w_e(\{3,4,7\})\\
&\quad + w_e(\{3,5,6\}) + w_e(\{3,5,7\}) + w_e(\{3,6,7\}))\\
&+\frac{1}{2}(w_e(\{4,5,6\}) + w_e(\{4,5,7\}) + w_e(\{4,6,7\}) + w_e(\{5,6,7\}))
\end{aligned}
$$

$$
\begin{aligned}
&v_1, v_2, v_3 \in \{1,2,3\},\ w_e(\{v_1\}) + w_e(\{v_2\}) \leq w_e(\{v_1, v_3\}) + w_e(\{v_2, v_3\}),\\
&v_1, v_2, v_3 \in \{1,2,3\},\ v_4, v_5, v_6, v_7 \in \{4,5,6,7\},\\
&\qquad w_e(\{v_4\}) \leq w_e(\{v_1, v_2, v_4\}) + w_e(\{v_4, v_5, v_6\}) - w_e(\{v_3, v_7\}),\\
&v_1, v_2, v_3 \in \{1,2,3\},\ v_4, v_5, v_6, v_7 \in \{4,5,6,7\},\\
&\qquad w_e(\{v_3, v_7\}) \leq w_e(\{v_1, v_2\}) + w_e(\{v_4, v_5, v_6\}),\\
&v_1, v_2, v_3 \in \{1,2,3\},\ v_4, v_5, v_6, v_7 \in \{4,5,6,7\},\\
&\qquad w_e(\{v_1, v_4, v_5\}) \leq w_e(\{v_2, v_3\}) + w_e(\{v_6.v_7\}),\\
&\qquad\qquad\qquad\qquad \Longrightarrow \quad \mathrm{Vol}_{G_e}(\partial\{1,2,3\}) \geq w_e(\{1,2,3\}).
\end{aligned}
$$

$$
\begin{aligned}
&v_1, v_2, v_3 \in \{1,2,3\},\ v_4, v_5, v_6, v_7 \in \{4,5,6,7\},\\
&\qquad w_e(\{v_1, v_4, v_5\}) \geq w_e(\{v_2, v_3\}) + w_e(\{v_4, v_5\}) - w_e(\{v_1, v_2, v_3\}),\\
&v_1, v_2, v_3 \in \{1,2,3\},\ v_4, v_5, v_6, v_7 \in \{4,5,6,7\},\\
&\qquad w_e(\{v_1, v_4\}) \geq w_e(\{v_1\}) + w_e(\{v_5, v_6, v_7\}) - w_e(\{v_1, v_2, v_3\}),\\
&v_1, v_2 \in \{1,2,3\},\ v_3 \in \{4,5,6,7\},\ w_e(\{v_3\}) \geq w_e(\{v_1, v_2, v_3\}) - w_e(\{v_1, v_2\}),\\
&v_1, v_2, v_3 \in \{1,2,3\},\ w_e(\{v_1\}) \geq w_e(\{v_2, v_3\}) - w_e(\{v_1, v_2, v_3\}),\\
&\qquad\qquad\qquad\qquad \Longrightarrow \quad \mathrm{Vol}_{G_e}(\partial\{1,2,3\}) \leq 6 w_e(\{1,2,3\}).
\end{aligned}
$$

A.2.3 Proof of Theorem 3.1.3

Suppose that $\{\mathcal{T}^o_{v\tilde{v}}\}_{\{v\tilde{v}\in E^{(e)}\}}$ and β^o represent the optimal solution of the optimization problem (3.8). To prove that the values of β^o are equal to those listed in Table 3.1, we proceed as follows. The result of the optimization procedure over $\{\mathcal{T}_{v\tilde{v}}\}_{\{v\tilde{v}\in E^{(e)}\}}$ produces the weights (coefficients) of the linear mapping. The optimization problem (3.8) may be rewritten as

$$
\begin{aligned}
\min_{\{\mathcal{T}_{v\tilde{v}}\}_{\{v,\tilde{v}\}\in E_o}} \quad & \beta \\
\text{s.t.} \quad & w_e(S) \leq \sum_{v\in S, \tilde{v}\in e/S} \mathcal{T}_{v\tilde{v}}(w_e) \leq \beta w_e(S), \\
& \qquad\qquad \forall\ S \in 2^e \text{ and submodular } w_e(\cdot),
\end{aligned}
\tag{A.2}
$$

which is essentially a linear programming. However, as there are uncountably many choices for the submodular functions $w_e(\cdot)$, the above optimization

problem has uncountably many constraints. However, given a finite collection of inhomogeneous cost functions $\Omega = \{w_e^{(1)}(\cdot), w_e^{(2)}(\cdot), ...\}$ all of which are submodular, the linear program

$$\begin{aligned} \min_{\{\mathcal{T}_{v\tilde{v}}\}_{\{v,\tilde{v}\}\in E_o}} \quad & \beta \\ \text{s.t.} \quad & w_e^{(r)}(S) \leq \sum_{v\in S, \tilde{v}\in e/S} \mathcal{T}_{v\tilde{v}}(w_e^{(r)}) \leq \beta w_e^{(r)}(S), \\ & \qquad \text{for all } S \in 2^e \text{ and } w_e^{(r)}(\cdot) \in \Omega \end{aligned} \tag{A.3}$$

can be efficiently computed and yields an optimal β^{Ω} that provides a lower bound for β^o. Therefore, we just need to identify the sets Ω for different values of $|e|$ that meet the values of $\beta^{(e)}$ listed in Table 3.1.

The proof involves solving the linear program (A.3). We start by identifying some structural properties of the problem.

Proposition A.2.1. *Given $|e|$, the optimization problem (3.8) over $\{\mathcal{T}_{v\tilde{v}}\}_{\{v\tilde{v}\in E^{(e)}\}}$ involves $3[\frac{|e|}{2}] - 1$ variables, where $[a]$ denotes the largest integer not greater than a.*

Proof. The linear mapping $\mathcal{T}_{v\tilde{v}}$ may be written as

$$\mathcal{T}_{v\tilde{v}}(w_e) = \sum_{S\in 2^e} \phi(v\tilde{v}, S) w_e(S),$$

where $\phi(v\tilde{v}, S)$ represent the coefficients that we wish to optimize, and which depend on the edge $v\tilde{v}$ and the subset S. Although we have $\binom{|e|}{2} \times 2^{|e|}$ coefficients, the coefficients are not independent from each other. To see what kind of dependencies exist, define the set of all permutations of the vertices of e $\pi : e \to \{1, 2, ..., |e|\}$; clearly, there are $|e|!$ such permutations π. Also, define $\pi(S) = \{\pi(v) | v \in S\}$ for $S \subseteq e$. If a set function $w(\cdot)$ over all the subsets of $\{1, 2, ..., |e|\}$ satisfies the conditions

$$\begin{aligned} w(\emptyset) &= 0, \\ w(S) &= w(\bar{S}), \\ w(S_1) + w(S_2) &\geq w(S_1 \cap S_2) + w(S_1 \cup S_2), \end{aligned}$$

for $S, S_1, S_2 \subseteq \{1, 2, ..., |e|\}$, then one may construct $|e|!$ many inhomogeneous cost functions $w_e^{(\pi)}(\cdot)$ such that for all distinct π one has $w_e^{(\pi)}(\cdot) = w(\pi(\cdot))$.

As all the weights $w_e^{(\pi)}$ are submodular and appear in the constraints of the optimization problem (A.2), the coefficients $\phi(v\tilde{v}, S)$ will be invariant under the permutations π; thus, they will depend only on two parameters, $|\{v, \tilde{v}\} \cap S|$ and $|S|$. We replace ϕ with another function $\tilde{\phi}$ to capture this dependence

$$\phi(v\tilde{v}, S) = \phi(\pi(v)\pi(\tilde{v}),\ \pi(S)) = \tilde{\phi}(|\{v, \tilde{v}\} \cap S|, |S|).$$

Moreover, as $w_e(S)$ is symmetric, i.e., as $w_e(S) = w_e(\bar{S})$, we also have

$$\tilde{\phi}(|\{v, \tilde{v}\} \cap \bar{S}|, |\bar{S}|) = \tilde{\phi}(|\{v, \tilde{v}\} \cap S|, |S|).$$

Hence, for any given $|e|$, the set of the coefficients of the linear function may be written as

$$\Phi = \{\tilde{\phi}(r, s) | (r, s) \in \{0, 1, 2\} \times \{1, 2, 3, ..., |e| - 2, |e| - 1\} / \{(2, 1), (0, |e| - 1)\},$$
$$\text{s.t.} \quad \tilde{\phi}(0, s) = \tilde{\phi}(2, |e| - s)\ ,\ \tilde{\phi}(1, s) = \tilde{\phi}(1, |e| - s)\},$$

which concludes the proof. □

Using Proposition A.2.1, we can transform the optimization problem (A.3) into the following form:

$$\min_{\Phi} \quad \beta$$
$$\text{s.t. } w_e^{(r)}(S) \leq \sum_{v \in S, \tilde{v} \in e/S} \sum_{S' \subseteq e} \tilde{\phi}(|\{v, \tilde{v}\} \cap S'|, |S'|) w_e^{(r)}(S') \leq \beta w_e^{(r)}(S),$$
$$\forall S \in 2^e, \forall w_e^{(r)}(\cdot) \in \Omega.$$

For a given $|e|$, we list the sets $\Omega = \{w_e^{(1)}, w_e^{(2)}, ...\}$ in Table. A.2. The above linear program yields optimal values of β^{Ω} equal to those listed in Table 3.1. As already pointed out, the cases $|e| = 2, 3$ are simple to verify, and hence we concentrate on the sets Ω for $|e| \geq 4$. The case $|e| = 7$ is handled similarly but requires a large verification table that we omitted for succinctness.

A.2.4 Proof of Theorem 3.1.4

For two pairs of vertices $v\tilde{v}, u\tilde{u} \in E^{(e)}$, count the number of $S \in 2^e/\{\emptyset, e\}$'s with $|S| = k$ and $u\tilde{u} \in \partial S$ that satisfy the following conditions:

1) $v, \tilde{v} \in \{u, \tilde{u}\}$ and $v \in S, \tilde{v} \in \bar{S}$: The number is $\binom{|e|-2}{k-1}$.

2) $v, \tilde{v} \in \{u, \tilde{u}\}$ and $v, \tilde{v} \in \bar{S}$: The number is 0.

3) $v \in \{u, \tilde{u}\}, \tilde{v} \notin \{u, \tilde{u}\}$ and $v \in S, \tilde{v} \in \bar{S}$: The number is $\binom{|e|-3}{k-1}$.

4) $v \in \{u, \tilde{u}\}, \tilde{v} \notin \{u, \tilde{u}\}$ and $v, \tilde{v} \in \bar{S}$: The number is $\binom{|e|-3}{k-1}$.

5) $v \notin \{u, \tilde{u}\}, \tilde{v} \in \{u, \tilde{u}\}$ and $v \in S, \tilde{v} \in \bar{S}$: The number is $\binom{|e|-3}{k-2}$.

6) $v \notin \{u, \tilde{u}\}, \tilde{v} \in \{u, \tilde{u}\}$ and $v, \tilde{v} \in \bar{S}$: The number is $\binom{|e|-3}{k-1}$.

7) $v, \tilde{v} \notin \{u, \tilde{u}\}$ and $v \in S, \tilde{v} \in \bar{S}$: The number is $2\binom{|e|-4}{k-2}$.

8) $v, \tilde{v} \notin \{u, \tilde{u}\}$ and $v, \tilde{v} \in \bar{S}$: The number is $2\binom{|e|-4}{k-1}$.

Moreover, some identities in the following can be demonstrated:

$$\begin{aligned}\sum_{k=1}^{|e|-1} \binom{|e|-3}{k-1} \frac{1}{k(|e|-k)} &\overset{a)}{=} \sum_{k=2}^{|e|-1} \binom{|e|-3}{k-2} \frac{1}{k(|e|-k)} \\ &\overset{b)}{=} \sum_{k=1}^{|e|-2} \binom{|e|-3}{k-1} \frac{1}{(k+1)(|e|-k-1)},\end{aligned}$$

$$\sum_{k=1}^{|e|-1} \binom{|e|-4}{k-2} \frac{1}{k(|e|-k)} \overset{c)}{=} \sum_{k=1}^{|e|-2} \binom{|e|-4}{k-1} \frac{1}{(k+1)(|e|-k-1)},$$

where the equalities are by substitution: a) $k \to |e| - k$, b) $k \to k + 1$, c) $k \to |e| - (k + 1)$.

As we assume that $w_e(S) = \sum_{u\tilde{u} \in \partial S} w^{(e)}_{u\tilde{u}}$, the RHS of formula (3.7) can be decomposed into the weighted sum of $w^e_{u\tilde{u}}$. As w_e is symmetric and the

above identities can be used,

$$
\begin{aligned}
& w_{v\tilde{v}}^{*(e)} \\
&= \sum_{S\in 2^e/\{\emptyset,e\}} \left[\frac{w_e(S)}{|S|(|e|-|S|)} 1_{v\in S,\tilde{v}\in\bar{S}} - \frac{w_e(S)}{(|S|+1)(|e|-|S|-1)} 1_{v,\tilde{v}\in\bar{S}} \right] \\
&= \sum_{k=1}^{|e|-1} \sum_{S:|S|=k} \frac{w_e(S)}{k(|e|-k)} 1_{v\in S,\tilde{v}\in\bar{S}} - \sum_{k=1}^{|e|-2} \sum_{S:|S|=k} \frac{w_e(S)}{(k+1)(|e|-k-1)} 1_{v,\tilde{v}\in\bar{S}} \\
&= \sum_{k=1}^{|e|-1} \sum_{S:|S|=k} \frac{1}{k(|e|-k)} 1_{v\in S,\tilde{v}\in\bar{S}} \sum_{u\tilde{u}\in\partial S} w_{u\tilde{u}}^e \\
&\quad - \sum_{k=1}^{|e|-2} \sum_{S:|S|=k} \frac{1}{(k+1)(|e|-k-1)} 1_{v,\tilde{v}\in\bar{S}} \sum_{u\tilde{u}\in\partial S} w_{u\tilde{u}}^e \\
&= \sum_{u\tilde{u}\in\partial S} w_{u\tilde{u}}^{(e)} \left\{ \sum_{k=1}^{|e|-1} \binom{|e|-2}{k-1} \frac{1}{k(|e|-k)} 1_{v,\tilde{v}\in\{u,\tilde{u}\}} \right. \\
&+ \left[\sum_{k=1}^{|e|-1} \binom{|e|-3}{k-1} \frac{1}{k(|e|-k)} - \sum_{k=1}^{|e|-2} \binom{|e|-3}{k-1} \frac{1}{(k+1)(|e|-k-1)} \right] 1_{v\in\{u,\tilde{u}\},\tilde{v}\notin\{u,\tilde{u}\}} \\
&+ \left[\sum_{k=1}^{|e|-1} \binom{|e|-3}{k-2} \frac{1}{k(|e|-k)} - \sum_{k=1}^{|e|-2} \binom{|e|-3}{k-1} \frac{1}{(k+1)(|e|-k-1)} \right] 1_{v\notin\{u,\tilde{u}\},\tilde{v}\in\{u,\tilde{u}\}} \\
&\left. + 2 \left[\sum_{k=1}^{|e|-1} \binom{|e|-4}{k-2} \frac{1}{k(|e|-k)} - \sum_{k=1}^{|e|-2} \binom{|e|-4}{k-1} \frac{1}{(k+1)(|e|-k-1)} \right] 1_{v\notin\{u,\tilde{u}\},\tilde{v}\notin\{u,\tilde{u}\}} \right\} \\
&= \sum_{k=1}^{|e|-1} \binom{|e|-2}{k-1} \frac{1}{k(|e|-k)} w_{v\tilde{v}}^{(e)} = \frac{2^{|e|}-2}{|e|(|e|-1)} w_{v\tilde{v}}^{(e)}
\end{aligned}
$$

which concludes the proof.

A.2.5 Complexity analysis

Recall that the proposed algorithm consists of three computational steps: 1) Projecting each InH-hyperedge onto a subgraph; 2) Combining the subgraphs to create a graph; 3) Performing spectral clustering on the derived graph based on Algorithm 3.1. The complexity of the algorithms depends on the complexity of these three steps. If in the first step we solve the

optimization procedure (3.2) for all InH-hyperedges with at most $2^{|e|}$ constraints, the worst case complexity of the algorithm is $O(2^{c\zeta(E)}|E|)$, where c is a constant that depends on the LP-solver. The second step has complexity $O((\zeta(E))^2|E|)$, while the third step has complexity $O(n^2)$, given that one has to find the eigenvectors corresponding to the extremal eigenvalues. Other benchmark hypergraph clustering algorithms such as Clique Expansion, Star Expansion [11] and Zhou's normalized hypergraph cut [120] share two steps of our procedure and hence have the same complexity for the corresponding computations. In practice, we usually deal with hyperedges of small size (< 10) and hence $\zeta(E)$ may, for all purposes, be treated as a constant. Hence, the complexity overhead of our method is of the same order as that of the last two steps, and hence we retain the same order of computation as classical homogeneous clustering methods. Nevertheless, to reduce the complexity of InH-partition, one may use predetermined mappings of the form (3.6) and (3.7). In the applications discussed in what follows, we exclusively used this computationally efficient approach.

A.3 Proof for Chapter 4

A.3.1 Proof for Equation (2.6)

Suppose that $y' \in \arg\max_{y\in\mathcal{B}}\langle y, x\rangle$. Then, $f(x) = \langle y', x\rangle$, and $f(x') \geq \langle y', x'\rangle$ for all $x' \in \mathbb{R}^N$. Therefore, $f(x') - f(x) \geq \langle y', x' - x\rangle$, and thus y' is a subgradient of f at x.

Suppose next that $y' \in \nabla f(x)$, and let $S \subseteq [N]$. If $S = [N]$, we have $f(x \pm 1_{[N]}) \geq f(x) \pm \langle y', 1_{[N]}\rangle$. As $f(x \pm 1_{[N]}) = f(x)$, so $y'([N]) = 0$. When $S \neq [[N]]$, we have

$$\begin{aligned} F(S) =& f(1_S) = \max_{y\in\mathcal{B}}\langle y, 1_S\rangle = \max_{y\in\mathcal{B}}\langle y, x + 1_S - x\rangle \geq \max_{y\in\mathcal{B}}\langle y, x + 1_S\rangle - \max_{y\in\mathcal{B}}\langle y, x\rangle \\ =& f(x + 1_S) - f(x) \overset{1)}{\geq} \langle y', x + 1_S - x\rangle = y'(S), \end{aligned}$$

where 1) follows from the definition of the subgradient. Hence, $y' \in \mathcal{B}$. As $y' \in \nabla f(x)$, we have $f(\mathbf{0}) - f(x) \geq \langle y', -x\rangle$, which implies $\langle y', x\rangle \geq f(x)$. Hence, $y' \in \arg\max_{y\in\mathcal{B}}\langle y, x\rangle$.

A.3.2 Proof for Theorem 4.1.4

We first prove Statement 1. Note that since

$$\nabla Q_p(x) = p\triangle_p(x),$$

$y \in \nabla Q_p(x)$ is equivalent to $y/p \in \triangle_p(x)$.

When $p > 1$, $\mathcal{S}_{p,\mu}$ is a differentiable and symmetric manifold. As $(\nabla \|x\|^p_{\ell_p,\mu})_v = p\mu_v\varphi_p(x_v)$, the tangent space of $\mathcal{S}_{p,\mu}$ at x is a vector space that can be described as follows

$$T_x(\mathcal{S}_{p,\mu}) = \left\{ \sum_{v\in[N]} c_v\chi_v, \text{ where } \{c_v\}_{v\in[N]} \text{ satisfies } \sum_{v\in[N]} c_v\mu_v\varphi_p(x_v) = 0 \right\},$$

where $\{\chi_v\}_{v\in[N]}$ is a canonical basis of $\mathbb{R}^N$. For a vector $y \in \nabla Q_p(x)$, its projection onto $T_x(\mathcal{S}_{p,\mu})$, i.e., $P_p(x)(y)$, vanishes if and only if $y \perp T_x(\mathcal{S}_{p,\mu})$. More precisely,

$$P_p(x)(y) = 0 \Leftrightarrow \text{there exists some } c \in \mathbb{R} \text{ such that } y_v = c\mu_v\varphi_p(x_v), \quad \forall v \in [N],$$

which implies that $y \in \nabla Q_p(x) \cap cU\varphi_p(x) \neq \emptyset$. Therefore, x is a critical point of $\tilde{Q}_p(x)$ if and only if x is an eigenvector of $\triangle_p$. The corresponding eigenvalue is $\lambda = \frac{\langle x, \triangle_p x\rangle}{\langle x, U\varphi_p(x)\rangle} = \frac{Q_p(x)}{\|x\|^p_{\ell_p,\mu}} = \tilde{Q}_p(x)$, i.e., the critical value of $\tilde{Q}_p$ at x.

When $p = 1$, $\mathcal{S}_{p,\mu}$ is a piecewise linear manifold, whose tangent space at $x \in \mathcal{S}_{p,\mu}$ is a cone. According to Theorem 4.2 in [66], for some vector $y \in \nabla Q_p(x)$, its projection onto the tangent space at x, i.e., $P_p(x)(y)$, vanishes if and only if there exists some $c \in \mathbb{R}$ and $\{c_u\}$, where $|c_u| \leq 1$, such that

$$y = c\left[\sum_{v:x_v\neq 0} \mu_v \text{sgn}(x_v)\chi_v + \sum_{u:x_u=0} \mu_u c_u \chi_u \right],$$

which implies $y \in cU\phi_p(x) \cap \nabla Q_p(x) \neq \emptyset$. Therefore, x is a critical point of $\tilde{Q}_p(x)$ if and only if x is an eigenvector of $\triangle_p$. The corresponding eigenvalue is $\lambda = \frac{\langle x, \triangle_p x\rangle}{\langle x, U\varphi_p(x)\rangle} = \frac{Q_p(x)}{\|x\|^p_{\ell_p,\mu}} = \tilde{Q}_p(x)$, i.e., the critical value of $\tilde{Q}_p$ at x.

Now we prove statements 2 and 3. For $p > 1$, $\|x\|_{\ell_p,\mu}^p$ is differentiable, so

$$\nabla R_p(x) = \frac{\|x\|_{\ell_p,\mu}^p \nabla Q_p(x) - pQ_p(x)U\varphi_p(x)}{\|x\|_{\ell_p,\mu}^{2p}} = \frac{p}{\|x\|_{\ell_p,\mu}^p}\left(\triangle_p(x) - R_p(x)U\varphi_p(x)\right). \tag{A.4}$$

Hence, $0 \in \nabla R_p(x)$ is equivalent to $0 \in \triangle_p(x) \cap R_p(x)U\varphi_p(x)$, i.e., $(x, R_p(x))$ is an eigenpair. However, for $p = 1$, we only have (see Proposition 2.3.14 [121])

$$\nabla R_p(x) \subseteq \frac{\|x\|_{\ell_p,\mu}^p \nabla Q_p(x) - pQ_p(x)U\varphi_p(x)}{\|x\|_{\ell_p,\mu}^{2p}}.$$

Therefore, $0 \in$ the set on the right-hand side does not necessarily imply that $0 \in \nabla R_p(x)$.

A.3.3 Proof for Lemma 4.2.2

The high level idea behind our proof is as follows: Given a hyperedge e, if for some nonempty $S \subset e$ we have $w_e(S) = 0$, then e can be split into two hyperedges $e_1 = S$ and $e_2 = e\backslash S$ with two modified submodular weights associated with e_1 and e_2. As the size of e is a constant, one can perform this procedure for all hyperedges e until all nonempty subsets S of e satisfy $w_e(S) > 0$.

Consider a hyperedge e with associated weight $w_e(S_1) = 0$ for some nonempty $S_1 \subset e$. Then, for any $S \subseteq e$, it must hold that

$$\begin{aligned}
2w_e(S) &\geq [w_e(S_1 \cup S) + w_e(S_1 \cap S) - w_e(S_1)] \\
&\quad + [w_e(S_1 \cup \bar{S}) + w_e(S_1 \cap \bar{S}) - w_e(S_1)] \\
&= [w_e(S_1 \cup S) + w_e(S_1 \cap S)] + [w_e(S\backslash S_1) + w_e(S_1\backslash S)] \\
&= [w_e(S \cap S_1) + w_e(S\backslash S_1)] + [w_e(S \cup S_1) + w_e(S_1\backslash S)] \\
&\geq 2w_e(S).
\end{aligned}$$

Hence, all inequalities must be strict equalities so that

$$\begin{aligned}
w_e(S) &= w_e(S_1 \cup S) + w_e(S_1 \cap S) = w_e(S_1\backslash S) + w_e(S\backslash S_1) \\
&= w_e(S \cap S_1) + w_e(S\backslash S_1) = w_e(S \cup S_1) + w_e(S_1\backslash S).
\end{aligned}$$

As a result, $w_e(S_1 \backslash S) = w_e(S \cap S_1)$ and $w_e(S \backslash S_1) = w_e(S \cup S_1)$. This implies that the hyperedge e can be partitioned into two hyperedges, $e_1 = S_1$ and $e_2 = e \backslash S_1$, with weights $(\vartheta_{e_i}, w_{e_i})_{i=1,2}$, such that

$$\vartheta_{e_i} = \max_{S \subset e_i} w_e(S), \quad \vartheta_{e_i} w_{e_i}(S) = \vartheta_e w_e(S) \quad \text{for all } S \subset e_i.$$

This partition ensures that w_{e_i} is a normalized, symmetric submodular function and that for any $S \subseteq e$, $\vartheta_e w_e(S) = \vartheta_{e_1} w_{e_1}(S \cap e_1) + \vartheta_{e_2} w_{e_2}(S \cap e_2)$. Therefore, for any subset S of $[N]$, the volume $\mathrm{vol}(\partial S)$ remains unchanged.

A.3.4 Proof for Lemma 4.2.4

Let x be an eigenvector associated with the eigenvalue 0. Then, $Q_p(x) = \langle x, \triangle_p x \rangle = 0$. Therefore, for each hyperedge e, we have $f_e(x) = 0$. Based on Lemma (4.2.2) of the main text, we may assume that the weights of G have been transformed so that for any $e \in E$ and any set $S \cap e \neq \{\emptyset, e\}$, one has $w_e(S) > 0$. Therefore, for any $v \in e$, x_v is a constant vector. As in the transformed G, for each pair of vertices $v, u \in [N]$, one can find a hyperedge path from v to u, so for all $v \in [N]$, x_v is a constant vector

A.3.5 Proof for discrete nodal domain theorems

The outline of the proof is similar to the one given by Tudisco and Hein [67] for graph $p-$Laplacians, with one significant change that involves careful handling of submodular hyperedges.

We start by introducing some useful notation. For a vector $x \in \mathbb{R}^N$ and a set $A \subset [N]$, define a vector $x|_A$ as

$$(x|_A)_v = \begin{cases} x_v & v \in A \\ 0 & v \notin A \end{cases}.$$

We also define the strong (weak) nodal space $\Xi(x)$ (respectively, $\xi(x)$) induced by x as the linear span of $x|_{A_1}, x|_{A_2}, \cdots, x|_{A_m}$, where $A_i, i = 1, \ldots, m$ are the strong (weak) nodal domains of x.

Lemma A.3.1. *A weak nodal space is a subspace of a strong nodal space. Hence, the number of weak nodal domains is upper bounded by the number*

of strong nodal domains for both $p = 1$ and $p > 1$ cases.

Proof. Suppose that the weak nodal domains of a vector x equal $A_1, A_2, ..., A_m$. Hence, its weak nodal space equals to $\xi(x) = \{y | y = \sum_{i\in[m]} \alpha_i x|_{A_i}, \alpha_i \in \mathbb{R}\}$. Let $Z = \{v \in [N] : x_v = 0\}$ and set $C_i = A_i \backslash Z$ for $i \in [m]$. The subgraph in G induced by the vertex set C_i may contain several connected components, in which case one may further partition C_i into disjoint sets $C_{i,1}, C_{i,2}, ..., C_{i,i_k}$, each of which corresponds to a connected component. It is easy to check that the strong nodal domains of x exactly consist of $\{C_{i,j}\}_{1\le i\le m, 1\le j\le i_k}$. Therefore, the strong nodal space equals $\Xi(x) = \{y | y = \sum_{i,j} \alpha_{ij} x|_{C_{ij}}, \alpha_{ij} \in \mathbb{R}\}$ and contains $\xi(x)$. □

Our subsequent analysis of nodal domains is primarily based on the following three lemmas.

Based on the deformation theorems for locally Lipschitzian even functions on $\mathcal{S}_{\mu,1}$ (Theorem 4.8 [66]) and $\mathcal{S}_{\mu,p}$ ($p > 1$, Theorem 3.1 [74]), one can guarantee that each critical value corresponds to at least one critical point, which is described in the first lemma.

Lemma A.3.2 (Lemma 2.2 [67])**.** *For $k \ge 1$ and $p \ge 1$, let $A^* \in \mathcal{F}_k(\mathcal{S}_{p,\mu})$ be a minimizing set, i.e., a set such that*

$$\lambda_k^{(p)} = \min_{A:\mathcal{F}_k(\mathcal{S}_{p,\mu})} \max_{x\in A} R_p(x) = \max_{x\in A^*} R_p(x).$$

Then A^* contains at least one critical point of $R_p(x)$ with respective to the critical value $\lambda_k^{(p)}$.

Lemma A.3.3 (Lemma 3.7 [67])**.** *Let $p > 1$, a, b, x, $y \in \mathbb{R}$, so that $x, y \ge 0$. Then*

$$|ax + by|^p \le (|a|^p x + |b|^p y)(x + y)^{p-1},$$

where the equality if and only if $xy = 0$ or $a = b$.

Lemma A.3.4. *Let $p \ge 1$ and let (x, λ) be an eigenpair of $\triangle_p$. Let $\Xi(x)$ ($\xi(x)$) be the strong (weak) nodal space induced by x. Then, for any vector $x' \in \Xi(x)$ ($\xi(x)$), it holds that $Q_p(x') \le \lambda \|x'\|_{\ell_{p,\mu}}^p$, and the inequality is tight for $p = 1$.*

Proof. Due to Lemma A.3.1, we only need to prove the claimed result for the strong nodal space. Suppose $A_1, A_2, ..., A_m$ are the strong nodal domains of x. Consider a vector in the strong nodal space of x, say $y = \sum_i \alpha_i x|_{A_i}$, where $\alpha_i \in \mathbb{R}$. The following observation is important when generalizing a result pertaining to graphs to the case of submodular hypergraphs. As we assume that the submodular hypergraph G is connected, we may without loss of generality assume that G is a hypergraph obtained from the transform described in Lemma 4.2.2. Then, based on the definition of nodal domains, each hyperedge e intersects at most two strong nodal domains with different signs. Hence, $x|_{A_i \cap e} \rightharpoonup x|_e$ for any $i \in [m], e \in E$ and $x|_{A_i \cap e} \rightharpoonup \operatorname{sgn}(\alpha_i) y|_e$ for any $i \in [m], \alpha_i \neq 0, e \in E$. From Lemma A.1.3, and for any $c \in \mathbb{R}, i \in [m]$, one has

$$\langle \nabla f_e(x), cx|_{A_i} \rangle = c \langle \nabla f_e(x|_e), x|_{A_i \cap e} \rangle = c f_e(x|_{A_i \cap e}) = c f_e(x|_{A_i}), \tag{A.5}$$

and

$$\begin{aligned} f_e(y) =& \langle f_e(y), y \rangle = \left\langle \nabla f_e(y), \sum_i \alpha_i x|_{A_i} \right\rangle = \sum_i \alpha_i \langle \nabla f_e(y), x|_{A_i \cap e} \rangle \\ =& \sum_i \alpha_i \langle \operatorname{sgn}(\alpha_i) \nabla f_e(\operatorname{sgn}(\alpha_i) y|_e), x|_{A_i \cap e} \rangle = \sum_i |\alpha_i| f_e(x|_{A_i \cap e}) \\ =& \sum_i |\alpha_i| f_e(x|_{A_i}). \end{aligned} \tag{A.6}$$

We partition the hyperedges into two sets according to how many nodal domains they intersect,

$$\begin{aligned} \mathcal{I}_1 &= \{e : |\{i | e \cap A_i \neq \emptyset\}| \leq 1\}, \\ \mathcal{I}_2 &= \{e : |\{i | e \cap A_i \neq \emptyset\}| = 2\}. \end{aligned}$$

Then, we have

$$\begin{aligned} Q_p(y) &= \sum_e \vartheta_e (f_e(y))^p \overset{1)}{=} \sum_e \vartheta_e \left(\sum_i |\alpha_i| f_e(x|_{A_i}) \right)^p \\ &= \sum_{e \in \mathcal{I}_1} \vartheta_e \sum_i |\alpha_i|^p (f_e(x|_{A_i}))^p + \sum_{e \in \mathcal{I}_2} \vartheta_e \left(\sum_i |\alpha_i| f_e(x|_{A_i}) \right)^p \\ &\overset{2)}{=} \sum_{e \in \mathcal{I}_1} \vartheta_e \sum_i |\alpha_i|^p f_e(x|_{A_i}) (f_e(x))^{p-1} + \sum_{e \in \mathcal{I}_2} \vartheta_e \left(\sum_i |\alpha_i| f_e(x|_{A_i}) \right)^p, \end{aligned}$$

where 1) follows from (A.6) and 2) is due to the fact that $f_e(x) = f_e(x|_{A_i})$ for those i such that $A_i \cap e \neq \emptyset$, and $f_e(x) = 0$ for those i such that $A_i \cap e = \emptyset$. Moreover, we have

$$\begin{aligned} \lambda \|y\|_{\ell_p,\mu}^p &= \sum_i |\alpha_i|^p \lambda \|x|_{A_i}\|_{\ell_p,\mu}^p \overset{1)}{=} \sum_i |\alpha_i|^p \langle x|_{A_i}, \triangle_p x \rangle \\ &= \sum_i |\alpha_i|^p \sum_e \vartheta_e \langle \nabla f_e(x), x|_{A_i} \rangle (f_e(x))^{p-1} \\ &\overset{2)}{=} \sum_i |\alpha_i|^p \sum_e \vartheta_e f_e(x|_{A_i}) (f_e(x))^{p-1}, \end{aligned}$$

where 1) is due to

$$\lambda \|x|_{A_i}\|_{\ell_p,\mu}^p = \langle x|_{A_i}, \lambda \varphi_p(x|_{A_i}) \rangle = \langle x|_{A_i}, \lambda \varphi_p(x) \rangle = \langle x|_{A_i}, \triangle_p x \rangle,$$

and 2) follows from (A.5). Therefore,

$$\begin{aligned} Q_p(y) - \lambda \|y\|_{\ell_p,\mu}^p &= \sum_{e \in \mathcal{I}_2} \vartheta_e \left[\left(\sum_i |\alpha_i| f_e(x|_{A_i}) \right)^p - \sum_i |\alpha_i|^p f_e(x|_{A_i}) (f_e(x))^{p-1} \right] \\ &= \sum_{e \in \mathcal{I}_2} \vartheta_e \tilde{f}_e(y), \end{aligned}$$

where

$$\begin{aligned} \tilde{f}_e(y) = \Big\{ &\left[|\alpha_{i_1}| f_e(x|_{A_{i_1}}) \; + |\alpha_{i_2}| f_e(x|_{A_{i_2}}) \right]^p - \\ &\left[|\alpha_{i_1}|^p f_e(x|_{A_{i_1}}) + |\alpha_{i_2}|^p f_e(x|_{A_{i_2}}) \right] \left[f_e(x|_{A_{i_1}}) + f_e(x|_{A_{i_2}}) \right]^{p-1} \Big\} \end{aligned}$$

and A_{i_1} and A_{i_2} are the two nodal domains intersecting e. Invoking Lemma A.3.3 proves the claimed result. □

Now, we are ready to prove Theorem 4.2.5. The proof of the strong nodal domain result for the graph $p-$Laplacian in [67] can be easily extended to our case via Lemma A.3.4, while the proof of the weak nodal domain result requires significant modifications.

Case 1: Strong nodal domains. Suppose that $\lambda_k^{(p)}$ has multiplicity r and associated eigenvector x. Let $\Xi(x)$ be the strong nodal space induced by x. If x supports m strong nodal domains, then $\gamma(\Xi(x) \cap \mathcal{S}_{p,\mu}) \leq m$. For any $x' \in \Xi(x) \cap \mathcal{S}_{p,\mu}$, we have $R_p(x') \leq R_p(x) = \lambda_k^{(p)}$ due to Lemma A.3.4. Therefore,

$$\lambda_m^{(p)} = \min_{A \in \mathcal{F}_m(\mathcal{S}_{p,\mu})} \max_{x' \in A} R_p(x') \leq \max_{x' \in \Xi(x) \cap \mathcal{S}_{p,\mu}} R_p(x') \leq \lambda_k^{(p)},$$

which implies $m \leq k+r-1$, where r is the mulplicity of λ_k. Given this upper bound of the number of strong nodal domains and Lemma A.3.1, one may naturally obtain bound for the number of weak nodal domains. However, for $p > 1$ case, one may derive a tighter bound.

Case 2: Weak nodal domains (for $p > 1$). Suppose that $\lambda_k^{(p)}$ has multiplicity r and associated eigenvector x. Suppose that $A_1, A_2, ..., A_m$ are the weak nodal domains of x. According to Lemma A.3.1, we know that m is upper bounded by the number of strong nodal domains which we know from Case 1 to be upper bounded by $k + r - 1$.

Let $\xi(x)$ be the weak nodal space induced by x. We use proof by contradiction and assume that $\dim(\xi(x)) > k$. Consider $\xi(x)'$ satisfying $\xi(x) = \text{Span}\{x\} \oplus \xi(x)'$.Then, we have $\gamma(\xi(x)' \cap \text{Span}\{x\}) \geq k$. Again, from Lemma A.3.4, it holds

$$\lambda_k^{(p)} \leq \min_{A \in \mathcal{F}_k(\mathcal{S}_{p,\mu})} \max_{x' \in A} \tilde{Q}_p(x') \leq \max_{x' \in \xi(x)' \cap \mathcal{S}_{p,\mu}} \tilde{Q}_p(x') \leq \tilde{Q}_p(x) = \lambda_k^{(p)},$$

which implies that $\xi(x)' \cap \mathcal{S}_{p,\mu}$ is a minimizing set in $\mathcal{F}_k(\mathcal{S}_{p,\mu})$. From Lemma A.3.2, it follows that there exists a critical point $y \in \xi(x)' \cap \mathcal{S}_{p,\mu}$ such that $\tilde{Q}_p(y) = \lambda_k^{(p)}$. Therefore, y is also an eigenvector of $\triangle_p$ with respect to the eigenvalue $\lambda_k^{(k)}$. Suppose that $y = \sum_i \alpha_i x|_{A_i}$. Later, we will show the contradiction by proving that $y \in \text{Span}\{x\}$, i.e., $\alpha_i = \alpha_j$ for all $i, j \in [m]$. For any two overlapping weak nodal domains, say A_1 and A_2 with $A_1 \cap A_2 \neq \emptyset$, consider the set of hyperedges that lie in $A_1 \cup A_2$, and denote this set by E^*. Without loss of generality, assume that A_1 is positive while A_2 is negative, as no hyperedge

can intersect two weak nodal domains with the same sign. Suppose that there exists a hyperedge $e \in E^*$ such that $e \cap (A_1 \backslash A_2)$ and $e \cap (A_2 \backslash A_1)$ are both nonempty. Then, both $f_e(x|_{A_1})$ and $f_e(x|_{A_2})$ are positive. According to the proof of Lemma A.3.4, as e intersects two strong nodal domains $A_1 \backslash A_2$ and $A_2 \backslash A_1$, in order to have $R_p(y) = \lambda_k^{(p)}$ one must also have $\tilde{f}_e(y) = 0$, which further implies $\alpha_1 = \alpha_2$. If there is no such hyperedge, then all hyperedges in E^* lie either in A_1 or A_2. Note that for all $u \in A_1 \cap A_2$, $\alpha_1 x_u = 0$, so that for $p > 1$, we have

$$\begin{aligned}
0 &= \lambda_k^{(p)} \mu_u \langle \mathbf{1}_u, \varphi_p(\alpha_1 x) \rangle = \langle \mathbf{1}_u, \triangle_p(\alpha_1 x) \rangle \\
&= \sum_{e: e \in E^*} \vartheta_e \langle \nabla f_e(\alpha_1 x), \mathbf{1}_u \rangle (f_e(\alpha_1 x))^{p-1} \\
&\overset{1)}{=} \sum_{e: e \in A_1} \vartheta_e \langle \nabla f_e(\alpha_1 x|_{A_1}), \mathbf{1}_u \rangle (f_e(\alpha_1 x|_{A_1}))^{p-1} \\
&+ \sum_{e: e \in A_2} \vartheta_e \langle \nabla f_e(\alpha_1 x|_{A_2}), \mathbf{1}_u \rangle (f_e(\alpha_1 x|_{A_2}))^{p-1},
\end{aligned}$$

where 1) is due to the fact that for all $e \subseteq A_1 \cap A_2$, one has $f_e(\alpha_1 x) = 0$. Similarly, as $y_u = 0$ and y is an eigenvector of $\triangle_p$, for $p > 1$, we have

$$\begin{aligned}
0 &= \lambda_k^{(p)} \mu_u \langle \mathbf{1}_u, \varphi_p(y) \rangle = \langle \mathbf{1}_u, \triangle_p(y) \rangle \\
&= \sum_{e: e \in E^*} \vartheta_e \langle \nabla f_e(y), \mathbf{1}_u \rangle (f_e(y))^{p-1} \\
&\overset{1)}{=} \sum_{e: e \subseteq A_1} \vartheta_e \langle \nabla f_e(\alpha_1 x|_{A_1}), \mathbf{1}_u \rangle (f_e(\alpha_1 x|_{A_1}))^{p-1} \\
&+ \sum_{e: e \subseteq A_2} \vartheta_e \langle \nabla f_e(\alpha_2 x|_{A_2}), \mathbf{1}_u \rangle (f_e(\alpha_2 x|_{A_2}))^{p-1},
\end{aligned}$$

where 1) once again is due to for all $e \subseteq A_1 \cap A_2$, one has $f_e(\alpha_1 x) = 0$. Subtracting the above two equations leads to

$$\begin{aligned}
0 = &\sum_{e: e \subseteq A_2} \vartheta_e \big[\langle \nabla f_e(\alpha_1 x|_{A_2}), \mathbf{1}_u \rangle (f_e(\alpha_1 x|_{A_2}))^{p-1} \\
&\qquad - \langle \nabla f_e(\alpha_2 x|_{A_2}), \mathbf{1}_u \rangle (f_e(\alpha_2 x|_{A_2}))^{p-1} \big] \\
= &(\varphi_p(\alpha_1) - \varphi_p(\alpha_2)) \sum_{e: e \subseteq A_2} \vartheta_e \langle f_e(x|_{A_2 \cap e}), \mathbf{1}_u \rangle (f_e(x|_{A_2}))^{p-1} \\
\overset{1)}{=} &(\varphi_p(\alpha_1) - \varphi_p(\alpha_2)) \sum_{e: e \subseteq A_2} \vartheta_e f_e(\mathbf{1}_u) (f_e(x|_{A_2}))^{p-1},
\end{aligned}$$

where 1) is due to $\mathbf{1}_u \rightharpoonup x|_{A_2 \cap e}$. Based of the definition of a weak nodal domain, there exists at least one hyperedge e intersecting both $A_1 \cap A_2$ and $A_2 \backslash A_1$. Therefore, for any $u \in A_1 \cap A_2$ such that $\sum_{e:e \subseteq A_2} \vartheta_e f_e(\mathbf{1}_u) f_e(x|_{A_2})^{p-1} > 0$, one has $\varphi_p(\alpha_1) = \varphi_p(\alpha_2)$ and consequently $\alpha_1 = \alpha_2$. Since the hypergraph is connected, it follows that $\alpha_1 = \alpha_2 = ... = \alpha_n = \alpha$, which implies that $y = \alpha x$. This is a contradiction and hence when $p > 1$, the number of weak nodal domains is $\leq k$.

Note that example 10 [68] shows that even for graphs, the number of weak nodal domains of an eigenvector of $\lambda_k^{(1)}$ can be greater than k.

A.3.6 Proof for Lemma 4.2.6

Consider a nonconstant eigenvector x and its corresponding eigenvalue λ. According to Lemma 4.2.4, if G is connected, then $\lambda \neq 0$. Moreover, when $p > 1$, $U\varphi_p(x)$ is a vector and not a set. Therefore, $\langle \mathbf{1}, U\varphi_p(x) \rangle \in \langle \mathbf{1}, \triangle_p(x) \rangle / \lambda = \sum_e \vartheta_e \langle \mathbf{1}, \nabla f_e(x) \rangle f_e(x)^{p-1} / \lambda = 0$. This implies that x contains both positive and negative components, which correspond to at least two weak (strong) nodal domains. Combining this result with that of Theorem 4.2.5 shows that the eigenvector corresponding to the eigenvalue λ_2 contains exactly two weak (strong) nodal domains.

For $p = 1$, we only have $\langle \mathbf{1}, U\varphi_p(x) \rangle \ni \langle \mathbf{1}, \triangle_p(x) \rangle / \lambda = 0$, which may allow that all components of x are either nonnegative or nonpositive. An example of a graph with a single weak (strong) nodal domain may be found in Example 11 of [68].

A.3.7 Proof for Lemma 4.2.7

According to the proof of Lemma 4.2.6, if $p > 1$, we have $\langle \mathbf{1}, U\varphi_p(x) \rangle = 0$ and thus $\mu_p^+(x) = \mu_p^-(x)$. Moreover, we have

$$\frac{\partial \|x - c\mathbf{1}\|_{\ell_p,\mu}^p}{\partial c}|_{c=0} = p \sum_{v \in [N]} \mu_v \operatorname{sgn}(x_v) |x_v|^{(p-1)} = \mu_p^+(x) - \mu_p^-(x) = 0. \quad \text{(A.7)}$$

Hence, $c \in \arg\min_{c \in \mathbb{R}} \|x - c\mathbf{1}\|_{\ell_p,\mu}^p$.

If $p = 1$, $0 \in \langle \mathbf{1}, U\varphi_p(x) \rangle$, which implies $|\mu_1^+(x) - \mu_1^-(x)| \leq \mu^0(x)$. Further-

more, for any $c \geq 0$ we have

$$\|x - c\mathbf{1}\|_{\ell_1,\mu} = \sum_{v:x_v>c} \mu_v(x_v - c) + \sum_{v:0\leq x_v<c} \mu_v(c - x_v) + \sum_{v:x_v<0} \mu_v(c - x_v)$$
$$= \sum_{v:x_v>0} \mu_v x_v - \sum_{v:x_v<0} \mu_v x_v + 2 \sum_{v:0<x_v<c} \mu_v(c - x_v) + c(\mu^0(x) + \mu_1^-(x) - \mu_1^+(x))$$
$$\geq \sum_{v:x_v>0} \mu_v x_v - \sum_{v:x_v<0} \mu_v x_v = \|x\|_{\ell_1,\mu}.$$

Therefore, $0 \in \arg\min_{c\in\mathbb{R}} \|x - c\mathbf{1}\|_{\ell_1,\mu}$.

A.3.8 Proof of Theorem 4.3.1

Let us first prove the second part of the theorem. Suppose that $\{S_1^*, S_2^*, ..., S_k^*\} \in P_k$ is one k-way partition such that $h_k = \max_{i\in[k]} c(S_i^*)$. Let

$$\mathcal{A} = \mathrm{Span}(\mathbf{1}_{S_1^*}, \mathbf{1}_{S_2^*}, ..., \mathbf{1}_{S_k^*}).$$

Choose a vector $x \in \mathcal{A} \cap \mathcal{S}_{p,\mu}$ and suppose that it can be written as $x = \sum_{i\in[k]} \alpha_i \mathbf{1}_{S_i^*}$.

Lemma A.3.5. *If $x = \sum_{i\in[k]} \alpha_i \mathbf{1}_{S_i^*}$ and $x \in \mathcal{S}_{p,\mu}$, then*

$$\sum_{i\in[k]} |\alpha_i|^p vol(S_i^*) = 1.$$

Proof. As $S_i^* \cap S_j^* = \emptyset$, we have $1 = \|x\|_{\ell_p,\mu}^p = \sum_{i\in[k]} \|\alpha_i \mathbf{1}_{S_i^*}\|_{\ell_p,\mu}^p = \sum_{i\in[k]} |\alpha_i|^p \mathrm{vol}(S_i^*)$. □

First, consider the following chain of inequalities that leads to an upper bound for $\tilde{Q}_p(x)$:

$$\begin{aligned}
\tilde{Q}_p(x) &= \sum_e \vartheta_e (f_e(x))^p = \sum_e \vartheta_e \langle \nabla f_e(x), x\rangle^p = \sum_e \vartheta_e \left[\sum_{i\in[k]} \alpha_i \langle \nabla f_e(x), \mathbf{1}_{S_i^*}\rangle\right]^p \\
&\overset{1)}{\leq} \sum_e \vartheta_e \left(\min\{|e|, k\}\right)^{p-1} \sum_{i\in[k]} |\alpha_i|^p |\langle \nabla f_e(x), \mathbf{1}_{S_i^*}\rangle|^p \qquad \text{(A.8)} \\
&\overset{2)}{\leq} \left(\min\{\max|e|, k\}\right)^{p-1} \sum_{i\in[k]} |\alpha_i|^p \sum_e \vartheta_e (f_e(\mathbf{1}_{S_i^*}))^p \\
&\overset{3)}{\leq} \left(\min\{\max|e|, k\}\right)^{p-1} \sum_{i\in[k]} |\alpha_i|^p \mathrm{vol}(\partial S_i^*) \\
&\overset{4)}{\leq} \left(\min\{\max|e|, k\}\right)^{p-1} \frac{\sum_{i\in[k]} |\alpha_i|^p \mathrm{vol}(\partial S_i^*)}{\sum_{i\in[k]} |\alpha_i|^p \mathrm{vol}(S_i^*)} \\
&\leq \left(\min\{\max|e|, k\}\right)^{p-1} h_k.
\end{aligned}$$

Here, 1) follows from $|\{i \in [k] | \langle \nabla f_e(x), \mathbf{1}_{S_i^* \cap e}\rangle > 0\}| \leq \min\{|e|, k\}$ and Hölder's inequality; 2) follows from the definition of f_e; 3) is a consequence of the inequality $\sum_e \vartheta_e (f_e(\mathbf{1}_{S_i^*}))^p \leq \sum_e \vartheta_e w_e(S_i)^p \leq \sum_e \vartheta_e w_e(S_i) = \mathrm{vol}(\partial S_i^*)$; and 4) follows from Lemma A.3.5.

Before establishing the lower bound, we first prove the following lemma.

Lemma A.3.6. *For any vector $x \in \mathbb{R}^N_{\geq 0}\backslash\{\mathbf{0}\}$ and $p \geq 1$, there exists some $\theta \geq 0$ such that $\Theta(x, \theta) = \{u : x(u) > \theta\}$ satisfies*

$$R_p(x) \geq \left(\frac{1}{\tau}\right)^{p-1} \left(\frac{c(S)}{p}\right)^p,$$

where $\tau = \max_{v\in[N]} \frac{d_v}{\mu_v}$.

Proof. Let us consider the case $p > 1$ first. For a vector x, we use $(x)^p$ to denote the coordinatewise p-th power operation. Furthermore, let $q = \frac{p}{p-1}$.

For a vector $x' \in \mathbb{R}^N$, we write the Lovász extension $f_e(x')$ by only includ-

ing arguments that lie in e, i.e.,

$$f_e(x') = \sum_{k=1}^{|e|-1} w_e(S^{k,e})(x'_{i_k(e)} - x'_{i_{k+1}(e)}),$$

where $e = \{i_k(e)\}_{1\le k\le |e|}$, $x'_{i_1(e)} \ge x'_{i_2(e)} \ge \cdots \ge x'_{i_{|e|}(e)}$ and $S^{k,e} = \{i_j(e)\}_{1\le j\le k}$. Then,

$$\begin{aligned}
&Q_1(x^p) = \sum_e \vartheta_e f_e(x^p) = \sum_e \vartheta_e \sum_{k=1}^{|e|-1} w_e(S^{k,e})(x^p_{i_k(e)} - x^p_{i_{k+1}(e)}) \\
&\overset{1)}{\le} \sum_e \vartheta_e \sum_{k=1}^{|e|-1} w_e(S^{k,e}) p(x_{i_k(e)} - x_{i_{k+1}(e)}) \left(x_{i_k(e)}\right)^{p-1} \qquad \text{(A.9)} \\
&= p \sum_e \sum_{k=1}^{|e|-1} \vartheta_e^{1/p} w_e(S^{k,e})(x_{i_k(e)} - x_{i_{k+1}(e)}) \, \vartheta_e^{1/q} \left(x_{i_k(e)}\right)^{p/q} \\
&\overset{2)}{\le} p \left\{ \sum_e \sum_{k=1}^{|e|-1} \vartheta_e \left[w_e(S^{k,e})(x_{i_k(e)} - x_{i_{k+1}(e)}) \right]^p \right\}^{\frac{1}{p}} \left\{ \sum_e \vartheta_e \sum_{k=1}^{|e|-1} \left(x_{i_k(e)}\right)^p \right\}^{\frac{1}{q}} \\
&\le p (Q_p(x))^{\frac{1}{p}} \left(\sum_v d_v x_v^p \right)^{\frac{1}{q}} \\
&\le p \tau^{1-\frac{1}{p}} (Q_p(x))^{\frac{1}{p}} \|x\|^{p-1}_{\ell_p,\mu},
\end{aligned}$$

where 1) follows from the fact that $a \ge b \ge 0$ implies $a^p - b^p \le p(a-b)a^{p-1}$ and 2) is a consequence of Hölder's inequality. As when $p = 1$, we naturally have $Q_1(x) \le Q_1(x)$. For any $p \ge 1$, we have

$$\frac{Q_1(x^p)}{\|x\|^p_{\ell_p,\mu}} \le p\tau^{1-\frac{1}{p}} \frac{(Q_p(x))^{\frac{1}{p}}}{\|x\|_{\ell_p,\mu}}. \qquad \text{(A.10)}$$

Moreover, by representing Lovász extension by its integral form [28], we obtain

$$\begin{aligned}
Q_1(x^p) &= \sum_e \vartheta_e \int_0^{+\infty} w_e(\{v : x_v^p > \theta\} \cap e) \, d\theta \\
&= \int_0^{+\infty} \vartheta_e \sum_e w_e(\{v : x_v^p > \theta\} \cap e) d\theta.
\end{aligned}$$

Then,

$$\frac{Q_1(x^p)}{\|x\|^p_{\ell_p,\mu}} = \frac{\int_0^{+\infty} \vartheta_e \sum_e w_e(\{v : x_v^p > \theta\} \cap e) d\theta}{\int_0^{+\infty} \mu(\{v : x_v^p > \theta\}) d\theta} \geq \inf_{\theta \geq 0} \frac{\sum_e \vartheta_e w_e(\{v : x_v^p > \theta\} \cap e)}{\mu(\{v : x_v^p > \theta\})}$$
$$= \inf_{\theta \geq 0} \frac{\mathrm{vol}(\partial\{v : x_v^p > \theta\})}{\mathrm{vol}(\{v : x_v^p > \theta\})} = \inf_{\theta \geq 0} c(\{v : x_v^p > \theta\}).$$

Therefore, the minimizer θ^* induces a set $\Theta^* = \{v : x_v^p > \theta^*\} \subseteq A$, for which the following inequality holds

$$R_p(x) = \frac{Q_p(x)}{\|x\|^p_{\ell_p,\mu}} \geq \left(\frac{Q_1(x^p)}{\|x\|^p_{\ell_p,\mu}}\right)^p \frac{1}{p^p \tau^{p-1}} = \left(\frac{1}{\tau}\right)^{p-1} \left(\frac{c(\Theta^*)}{p}\right)^p.$$

This proves Lemma A.3.6.

Next, we turn our attention to the first inequality of Theorem 4.3.1. Suppose $\lambda_k^{(p)}$ has a corresponding eigenvector x that induces the strong nodal domains $A_1, A_2, ..., A_m$. According to Lemma A.3.4, we know that $\lambda_k^{(p)} \geq R_p(\mathbf{1}_{A_i})$. Moreover, due to Lemma A.3.6, for any $i \in [m]$, there exists a $B_i \subseteq A_i$ such that

$$R_p(\mathbf{1}_{A_i}) \geq \left(\frac{1}{\tau}\right)^{p-1} \left(\frac{c(B_i)}{p}\right)^p.$$

Therefore,

$$\lambda_k^{(p)} \geq \max_{i \in [m]} R_p(\mathbf{1}_{A_i}) \geq \max_{i \in [m]} \left(\frac{1}{\tau}\right)^{p-1} \left(\frac{c(B_i)}{p}\right)^p$$
$$\geq \min_{(B_1, B_2, ..., B_m) \in P_m} \max_{i \in [m]} \left(\frac{1}{\tau}\right)^{p-1} \left(\frac{c(B_i)}{p}\right)^p \geq \left(\frac{1}{\tau}\right)^{p-1} \left(\frac{h_m}{p}\right)^p.$$

□

Homogeneous Weights

We can use a similar approach to prove the previous result for homogeneous weights, i.e., weights such that $w_e(S) = 1$ for all $S \in 2^e \backslash \{\emptyset, e\}$. Only several steps have to be changed.

First, the inequality (A.8) may be tightened. Again, consider the partition $\{S_1^*, S_2^*, ..., S_k^*\} \in P_k$ such that $h_k = \max_{i \in [k]} c(S_i^*)$. For a given hyperedge e,

choose a pair of vertices $(u^*, v^*) \in \arg\max_{u,v\in e} |x_u - x_v|^p$. If both $u, v \in S_i^*$, then $f_e(x) = 0$. If not, assume that $u \in S_i^*$ and $v \in S_j^*$. Then,

$$\begin{aligned}(f_e(x))^p &= |x_{u^*} - x_{v^*}|^p \le 2^{p-1}(|x_{u^*}|^p + |x_{v^*}|^p) \\ &\le 2^{p-1}(|\alpha_i|^p f_e(1_{S_i^*})^p + |\alpha_j|^p f_e(1_{S_j^*})^p) = 2^{p-1}\sum_{i\in[k]} |\alpha_i|^p f_e(\mathbf{1}_{S_i^*})^p.\end{aligned}$$

Therefore, in the homogeneous case, we have

$$R_p(x) \le 2^{p-1} h_k.$$

Second, we will use the following lemma to prove the lower bound:

Lemma A.3.7 ([63])**.** *If $a, b \ge 0$, $p > 1$, then*

$$a^p - b^p \le \frac{p}{2^{1-\frac{1}{p}}}(a-b)\left(a^p + b^p\right)^{1-\frac{1}{p}}.$$

So the inequality (A.9) may be tightened as

$$f_e(x^p) = x_{i_0(e)}^p - x_{i_{|e|}(e)}^p \le \frac{p}{2^{1-\frac{1}{p}}}(x_{i_0(e)} - x_{i_{|e|}(e)})\left(x_{i_0(e)}^p + x_{i_{|e|}(e)}^p\right)^{1-\frac{1}{p}}.$$

With these two modifications, we can rewrite inequality (A.10) as

$$\frac{Q_1(x^p)}{\|x\|_{\ell_p,\mu}^p} \le \frac{p}{2^{1-\frac{1}{p}}}\tau^{1-\frac{1}{p}}\frac{(Q_p(x))^{\frac{1}{p}}}{\|x\|_{\ell_p,\mu}},$$

which leads to

$$\lambda_k^{(p)} \ge \left(\frac{2}{\tau}\right)^{p-1}\left(\frac{h_m}{p}\right)^p.$$

A.3.9 Proof of Theorem 4.4.1

First, we prove that $\lambda_2^{(p)} \ge \inf_x \mathcal{R}_p(x)$. Suppose that x' is a nonconstant eigenvector corresponding to $\lambda_2^{(p)}$. If $\lambda_2^{(p)} = 0$, then $\langle x', \triangle_p(x')\rangle = \langle x', \lambda_2^{(p)} U x'\rangle = 0$, which implies that $Q_p(x') = 0$. Moreover, as x' is nonconstant, $\min_{c\in\mathbb{R}} \|x' - c\mathbf{1}\|_{\ell_p,\mu}^p > 0$, and thus $\mathcal{R}_p(x') = 0 \le \lambda_2^{(p)}$. This proves the claim of the theorem for the case that $\lambda_2^{(p)} = 0$. Next, suppose that $\lambda_2^{(p)} \ne 0$. First, we observe that Lemma 4.2.7 implies $0 \in \nabla_c Z_{p,\mu}(x', c)|_{c=0}$. As $Z_{p,\mu}(x', c)$ is convex in

c, $c = 0$ is a minimizer of $Z_{p,\mu}(x', c)$, *i.e.*, $Z_{p,\mu}(x', 0) = Z_{p,\mu}(x')$. Moreover, $\lambda_2^{(p)} = R_p(x') = \frac{Q_p(x')}{Z_{p,\mu}(x',0)} = \frac{Q_p(x')}{Z_{p,\mu}(x')} = \mathcal{R}_p(x')$. Therefore, $\lambda_2^{(p)} \geq \inf_x \mathcal{R}_p(x)$.

Second, we prove that $\inf_x \mathcal{R}_p(x) \geq \lambda_2^{(p)}$. First, we focus on the case $p > 1$. For any $t_1 \in \mathbb{R}\backslash\{0\}$ and $t_2 \in \mathbb{R}$, it is easy to show that $\mathcal{R}_p(t_1 x + t_2 \mathbf{1}) = \mathcal{R}_p(x)$. Therefore, to characterize the infimum of $\mathcal{R}_p(x)$, it suffices to consider $x \in \mathcal{S}_{p,\mu} \cap \mathcal{A}$, where $\mathcal{A} = \{x \in \mathbb{R}^N | 0 \in \arg\min_c Z_{p,\mu}(x, c)\}$. For $p > 1$, $Z_{p,\mu}(x, c)$ is differentiably convex in c. By using formula (A.7) once again, we know that $\mathcal{A} = \{x \in \mathbb{R}^N | \mu_p^+(x) - \mu_p^-(x) = 0\}$. Furthermore, $\mathcal{A}$ is closed, since the functions μ_p^+, μ_p^- are continuous. By recalling that $\mathcal{S}_{p,\mu}$ is a compact space we know that there exists a point $x_* \in \mathcal{S}_{p,\mu} \cap \mathcal{A}$ such that $x_* \in \arg\inf_x \mathcal{R}_p(x)$.

Consider next the subspace $\mathcal{A}' = \{t_1 x_* + t_2 \mathbf{1} : t_1, t_2 \in \mathbb{R}\}$. As x_* being nonconstant reduces to $x_* \neq c\mathbf{1}$ for any scalar $c \in \mathbb{R}$, we have $\gamma(\mathcal{A} \cap \mathcal{S}_p) = 2$. According to the definition of $\lambda_2^{(p)}$ (4.3), it follows that

$$\begin{aligned}\lambda_2^{(p)} &\leq \max_{x \in \mathcal{A}' \cap \mathcal{S}_{p,\mu}} Q_p(x) = \max_{t_1, t_2 \in \mathbb{R}} Q_p\left(\frac{t_1 x_* + t_2 \mathbf{1}}{\|t_1 x_* + t_2 \mathbf{1}\|_{\ell_p,\mu}}\right) = \max_{t_1, t_2 \in \mathbb{R}} \frac{Q_p(t_1 x_*)}{\|t_1 x_* + t_2 \mathbf{1}\|_{\ell_p,\mu}^p} \\ &= \max_{t_1 \in \mathbb{R}} \frac{Q_p(t_1 x_*)}{Z_{p,\mu}(t_1 x_*)} = \mathcal{R}_p(x_*).\end{aligned}$$

For any $a, b \in \mathbb{R}$, we can write $Q_p(ax + b\mathbf{1}) = |a|^p Q_p(x)$ and $Z_{p,\mu}(ax + b\mathbf{1}) = |a|^p Z_{p,\mu}(x)$. Combining these expressions with $\lambda_2^{(p)} \geq \inf_x \mathcal{R}_p(x)$ shows that $\lambda_2^{(p)} = \inf_x \mathcal{R}_p(x)$. This settles the case $p > 1$.

Next, we turn our attention to proving that $\min_x \mathcal{R}_1(x) = h_2$ for $p = 1$. This result, combined with the inequality $h_2 \geq \lambda_2^{(1)}$ from Theorem 4.3.1 proves that $\inf_x \mathcal{R}_1(x) = h_2 = \lambda_2^{(1)}$.

Recall that the 2-way Cheeger constant can be written as

$$\min_{S \subset [N]} \frac{|\partial S|}{\min\{\mathrm{vol}(S), \mathrm{vol}([N]\backslash S)\}}.$$

This expression, along with the fact that $\inf_x \mathcal{R}_1(x) = h_2$ (which is a special case of Theorem 1 in [122]), allows one to reduce the proof to showing that the Lovász extensions of $\mathrm{vol}(\partial S)$ and $\min\{\mathrm{vol}(S), \mathrm{vol}([N]\backslash S)\}$ are equal to $Q_1(x)$ and $Z_{1,\mu}(x)$, respectively. The claim regarding Q_1 naturally follows from the Definition 4.1.1. We hence only need to show that the Lovász extension of $\min\{\mathrm{vol}(S), \mathrm{vol}([N]\backslash S)\}$ equals $Z_{1,\mu}(x)$.

For a given $x \in \mathbb{R}^N$, suppose that $x_{i_1} \geq x_{i_2} \geq \cdots \geq x_{i_N}$. Then, the Lovász

extension of $\min\{\mathrm{vol}(S), \mathrm{vol}([N]\backslash S)\}$ can be written as

$$\sum_{k=1}^{N} \min\{\sum_{j=1}^{k} \mu_{i_j}, \sum_{j=k+1}^{N} \mu_{i_j}\} (x_{i_j} - x_{i_{j+1}}). \tag{A.11}$$

Let k^* be equal to $\min\left\{k \in \{1,2,...,N\} : \sum_{j=1}^{k} \mu_{i_j} \geq \sum_{j=k+1}^{N} \mu_{i_j}\right\}$. In this case, (A.11) is equivalent to

$$\sum_{k=1}^{k^*-1} \mu_{i_k}(x_{i_k} - x_{i_{k^*}}) + \sum_{k=k^*+1}^{N} \mu_{i_k}(x_{i_{k^*}} - x_{i_k}) = \|x - x_{i_{k^*}}\mathbf{1}\|_{\ell_1,\mu} = Z_{1,\mu}(x),$$

which establishes the claimed result.

A.3.10 Proof for Theorem 4.4.3

For a vector $x \in \mathbb{R}^N$, define two vectors $x^+, x^- \in \mathbb{R}^N$ according to $(x^+)_v = \max\{x_v, 0\}$ and $(x^-)_v = \max\{-x_v, 0\}$. Hence, $x = x^+ - x^-$ and $x^+, -x^- \rightharpoonup x$. Then,

$$\begin{aligned} Q_p(x) &= \sum_e \vartheta_e f_e(x)^p = \sum_e \vartheta_e[\langle \nabla f_e(x), x^+\rangle + \langle \nabla f_e(x), -x^-\rangle]^p \\ &\overset{1)}{=} \sum_e \vartheta_e[f_e(x^+) + f_e(-x^-)]^p \\ &\overset{2)}{=} \sum_e \vartheta_e[f_e(x^+)^p + f_e(x^-)^p] = Q_p(x^+) + Q_p(x^-), \end{aligned}$$

where in 1) we used Lemma A.1.3, and in 2) we used the fact that $f_e(x) = f_e(-x)$ and $(a+b)^p \geq a^p + b^p$ for $a, b \geq 0$, $p \geq 1$. Moreover, as $Z_{p,\mu}(x) = \|x\|_{\ell_p,\mu}^p = \|x^+\|_{\ell_p,\mu}^p + \|x^-\|_{\ell_p,\mu}^p$, we have

$$\mathcal{R}_p(x) \geq \min\{R_p(x^+), R_p(x^-)\}.$$

By applying Lemma A.3.6 on x^+ and x^-, and by observing that $c(x^+), c(x^-) \geq c(x)$, we have

$$\begin{aligned}\mathcal{R}_p(x) &\geq \min\{R_p(x^+), R_p(x^-)\} \geq \left(\frac{1}{\tau}\right)^{p-1} \left(\frac{\min\{c(x^+), c(x^-)\}}{p}\right)^p \\ &\geq \left(\frac{1}{\tau}\right)^{p-1} \left(\frac{c(x)}{p}\right)^p,\end{aligned}$$

which concludes the proof.

A.3.11 Proof for Lemma 4.4.4

First, it can be easily shown that $Ux \perp \mathbf{1}$, since

$$\sum_{v\in[N]} \mu_v x_v = \sum_{v\in[N]} \mu_v (x'_v)^T g = (\sum_{v\in[N]} \mu_v x'_v)^T g = 0.$$

Next, we establish a lower bound for $\|x\|^2_{\ell_2,\mu}$. For this purpose, we find the following lemma useful.

Lemma A.3.8 (Lemma 7.7 [8])**.** *Let Y_1, Y_2, ..., Y_k be zero-mean normal random variables that are not necessarily independent, such that $\mathbb{E}[\sum_i Y_i^2] = 1$. Then,*

$$\mathbb{P}\left[\sum_i Y_i^2 \geq \frac{1}{2}\right] \geq \frac{1}{12}.$$

We start by observing that

$$\mathbb{E}[\|x\|^2_{\ell_2,\mu}] = \mathbb{E}[\|X^T g\|^2_{\ell_2,\mu}] = \sum_{v\in[N]} \mu_v \|x'_v\|_2^2 = 1.$$

From Lemma A.3.8, it follows that

$$\mathbb{P}\left[\|x\|^2_{\ell_2,\mu} \geq \frac{1}{2}\right] \geq \frac{1}{12}. \tag{A.12}$$

Next, we prove an upper bound for $Q_2(x)$. For any $e \in E$, $w \in \mathcal{E}(\mathcal{B}_e)$, we

have

$$\mathbb{E}\left[\left(\max_{y\in\mathcal{E}(\mathcal{B}_e)}\langle y,x'\rangle\right)^2\right]=\mathbb{E}\left[\left(\max_{y\in\mathcal{E}(\mathcal{B}_e)}\left\langle g,\frac{Xy}{\|Xy\|_2}\right\rangle\right)^2\|Xy\|_2^2\right]$$
$$\leq\mathbb{E}\left[\left(\max_{y\in\mathcal{E}(\mathcal{B}_e)}\left\langle g,\frac{Xy}{\|Xy\|_2}\right\rangle\right)^2\right]\max_{y'\in\mathcal{E}(\mathcal{B}_e)}\|Xy'\|_2^2. \tag{A.13}$$

Suppose that the hyperedge e contains the following vertices $e=\{v_1,v_2,\ldots,v_{|e|}\}$. Let $\mathbb{A}=\mathrm{Span}(x'_{v_1}-x'_{v_{|e|}},x'_{v_2}-x'_{v_{|e|}},...,x'_{v_{|e|-1}}-x'_{v_{|e|}})$ and let $\mathbb{S}^n$ stand for the unit ball in $\mathbb{R}^n$. Recall n is the dimension of the space to embed the vectors for SDP relaxation which is no less than $\zeta(E)$. Then, given that $\sum_{v\in e}y_v=0$ and $y_u=0$ for $u\notin e$, $\frac{Xy}{\|Xy\|_2}$ always lies in $\mathbb{A}\cap\mathbb{S}^n$. Therefore,

$$\mathbb{E}\left[\left(\max_{y\in\mathcal{E}(\mathcal{B}_e)}\left\langle g,\frac{Xy}{\|Xy\|_2}\right\rangle\right)^2\right]\leq\mathbb{E}\left[\left(\max_{x'\in\mathbb{A}\cap\mathbb{S}^n}\langle g,x'\rangle\right)^2\right]=\dim(\mathbb{A})=|e|-1. \tag{A.14}$$

Combining (A.13) with (A.14), we have

$$\mathbb{E}[(\max_{y\in\mathcal{E}(\mathcal{B}_e)}\langle y,x'\rangle)^2]\leq(|e|-1)\max_{y\in\mathcal{E}(\mathcal{B}_e)}\|Xy\|_2^2.$$

As $Q_2(x)=\sum_{e\in E}w_e(\max_{y\in\mathcal{E}(\mathcal{B}_e)}\langle y,x\rangle)^2$, using Markov's inequality, we have

$$\mathbb{P}\left(Q_2(x)\geq 13\,\zeta(E)\sum_{e\in E}\max_{w'\in\mathcal{E}(\mathcal{B}_e)}\|Yw'\|_2^2\right)\leq\frac{1}{13}. \tag{A.15}$$

In addition, applying the union bound to (A.15) and using (A.12), we have

$$\mathbb{P}\left(\mathcal{R}_2(x)\leq 26\ \mathrm{SDPopt}\right)\geq\frac{1}{13}, \tag{A.16}$$

which concludes the proof.

Note that the distortion term $O(\zeta(E))$ is introduced through the inequalities (A.13) and (A.14), which are tight for this case. This may be shown as follows. Suppose the solution of the SDP produces a collection of vectors $\{x'_{v_i}\}_{1\leq i\leq|e|}$ that have the same ℓ_2-norm, i.e. $\|x'_{v_i}\|_2=a$, and are orthogonal in $\mathbb{R}^n$. Let $\mathcal{B}_e$ denote the base polytope corresponding to a submodular

function satisfying $w_e(S) = \frac{2}{|e|}\min\{|S\cap e|, |e| - |S\cap e|\}$. Define a subset of $\mathcal{B}_e$, $\mathcal{B}_{e,\mathrm{s}}$, as follows:

$$\mathcal{B}_{e,\mathrm{s}} \triangleq \Big\{ y\in\mathbb{R}^N \big| |y(\{v_i\})| \le \frac{2}{|e|}, y(\{v_{i+|e|/2}\}) = -y(\{v_i\}),\ \text{for } 1\le i\le |e|/2,\\ y(\{v\}) = 0,\ \text{for } v\notin e\Big\}.$$

Then, choosing a y' in $B_{e,\mathrm{s}}$ such that $y'(\{v_i\}) = \frac{2}{|e|}\frac{\langle g, x'_{v_i} - x'_{v_{i+|e|/2}}\rangle}{|\langle g, x'_{v_i} - x'_{v_{i+|e|/2}}\rangle|}$ for $1\le i\le |e|/2$, we obtain

$$\begin{aligned}
&\mathbb{E}\left[\left(\max_{y\in\mathcal{E}(\mathcal{B}_e)}\langle g, Xy\rangle\right)^2\right]\\
&\ge \mathbb{E}\left[\langle g, Xy'\rangle^2\right]\\
&= \mathbb{E}\left[\left\langle g, \sum_{1\le i\le |e|/2}\frac{2}{|e|}\frac{\langle g, x'_{v_i} - x'_{v_{i+|e|/2}}\rangle}{|\langle g, x'_{v_i} - x'_{v_{i+|e|/2}}\rangle|}(x'_{v_i} - x'_{v_{i+|e|/2}})\right\rangle^2\right]\\
&\ge \frac{4}{|e|^2}\mathbb{E}\left[\sum_{1\le i\le |e|/2}|\langle g, x'_{v_i} - x'_{v_{i+|e|/2}}\rangle|\right]^2\\
&\ge \frac{4}{|e|^2}\left[\sum_{1\le i\le |e|/2}\mathbb{E}|\langle g, x'_{v_i} - x'_{v_{i+|e|/2}}\rangle|\right]^2\\
&= \frac{4}{|e|^2}\left(\frac{|e|}{2}\times\sqrt{2}a\sqrt{\frac{2}{\pi}}\right)^2 = \frac{|e|}{\pi}\frac{4}{|e|}a^2\\
&\ge \frac{|e|}{\pi}a^2\max_{y\in\mathcal{E}(\mathcal{B}_e)}\|y\|^2 \ge \frac{|e|}{\pi}\max_{y\in\mathcal{E}(\mathcal{B}_e)}\sum_{1\le i\le |e|}\|y(\{v_i\})x_{v_i}\|_2^2\\
&= \frac{1}{\pi}|e|\max_{y\in\mathcal{E}(\mathcal{B}_e)}\|Xy\|^2,
\end{aligned}$$

where the last equality is using the assumption that $\{x_{v_i}\}_{v_i\in e}$ are mutually orthogonal. Therefore, the Gaussian projection X causes distortion $\Theta(|e|)$.

A.3.12 Proof of Theorem 4.4.5

By combining Theorem 4.4.1, Theorem 4.4.3, Lemma 4.4.4 and Theorem (4.3.1), we obtain

$$\begin{aligned} c(x) &\leq O(\sqrt{\tau})\mathcal{R}_2(x)^{1/2} \leq O(\sqrt{\zeta(E)\,\tau})\left(\inf_x \mathcal{R}_2(x)\right)^{1/2} \\ &= O(\sqrt{\zeta(E)\,\tau})\left(\lambda_2^{(2)}\right)^{1/2} \leq O(\sqrt{\zeta(E)\,\tau h_2}) \text{ w.h.p.} \end{aligned}$$

A.3.13 Proof of Theorem 4.4.6

First, according to Step 3, we have

$$Q_1(z^{k+1}) - \hat{\lambda}^k\langle z^{k+1}, g^k\rangle \leq Q_1(z^k) - \hat{\lambda}^k\langle z^k, g^k\rangle.$$

It is also straightforward to check that g^k satisfies

$$\langle \mathbf{1}, g^k\rangle = 0, \qquad \langle x^k, g^k\rangle = \|x^k\|_{\ell_1,\mu}.$$

Therefore,

$$\begin{aligned} Q_1(x^{k+1}) - \hat{\lambda}^k\langle x^{k+1}, g^k\rangle &= Q_1(z^{k+1}) - \hat{\lambda}^k\langle z^{k+1}, g^k\rangle \leq Q_1(z^k) - \hat{\lambda}^k\langle z^k, g^k\rangle \\ &= Q_1(x^k) - \hat{\lambda}^k\langle x^k, g^k\rangle = Q_1(x^k) - \hat{\lambda}^k\|x^k\|_{\ell_1,\mu} = 0, \end{aligned}$$

which implies

$$\mathcal{R}_1(x^{k+1}) \leq \hat{\lambda}^k\frac{\langle x^{k+1}, g^k\rangle}{Z_{1,\mu}(x^{k+1})} = \hat{\lambda}^k\frac{\langle x^{k+1}, g^k\rangle}{\|x^{k+1}\|_{\ell_1,\mu}} \leq \hat{\lambda}^k\frac{\|x^{k+1}\|_{\ell_1,\mu}\|g^k\|_{\ell_\infty,\mu^{-1}}}{\|x^{k+1}\|_{\ell_1,\mu}} \overset{1)}{\leq} \hat{\lambda}^k.$$

Here, 1) follows from Lemma 3.11 which implies $\|g^k\|_{\ell_\infty,\mu^{-1}} \leq 1$. This proves the claimed result.

A.3.14 Proof of Theorem 4.4.7

If the norm $\|z\|$ stands for $\|z\|_2$, the duality result holds since

$$\begin{aligned}
\min_{z:\|z\|_2\leq 1} Q_1(z) - \hat{\lambda}^k\langle z, g^k\rangle &= \min_{z}\max_{\lambda\geq 0}\max_{y_e\in\vartheta_e\mathcal{B}_e}\sum_e\langle y_e, z\rangle - \hat{\lambda}^k\langle z, g^k\rangle + \frac{\lambda}{2}(\|z\|_2^2 - 1)\\
&= \max_{y_e\in\vartheta_e\mathcal{B}_e}\max_{\lambda\geq 0}\min_{z}\sum_e\langle y_e, z\rangle - \hat{\lambda}^k\langle z, g^k\rangle + \frac{\lambda}{2}(\|z\|_2^2 - 1)\\
&= \max_{y_e\in\vartheta_e\mathcal{B}_e}\max_{\lambda\geq 0} -\frac{\|\sum_{e\in E} y_e - \hat{\lambda}^k g^k\|_2^2}{2\lambda} - \frac{\lambda}{2}\\
&= \max_{y_e\in\vartheta_e\mathcal{B}_e} -\|\sum_{e\in E} y_e - \hat{\lambda}^k g^k\|_2.
\end{aligned}$$

The relationships between the primal and dual variables read as $z = \frac{\hat{\lambda}^k g^k - \sum_{e\in E} y_e}{\lambda}$ and $\lambda = \|\sum_{e\in E} y_e - \hat{\lambda}^k g^k\|_2$.

If the norm $\|z\|$ stands for $\|z\|_\infty$, let $z' = (z + \mathbf{1})/2$. As $Q_1(z') = Q(z)/2$ and $\langle g^k, z'\rangle = \langle g^k, z\rangle/2$, we have

$$\min_{z:\|z\|_\infty\leq 1}\frac{1}{2}[Q_1(z) - \hat{\lambda}^k\langle z, g^k\rangle] \quad\Longleftrightarrow\quad \min_{z':z'\leq[0,1]^N} Q_1(z') - \hat{\lambda}^k\langle z', g^k\rangle.$$

The right-hand side essentially reduces to the following discrete optimization problem (Proposition 3.7 [28])

$$\min_{S\subseteq[N]}\sum_e \vartheta_e w_e(S) - \hat{\lambda}^k g^k(S),$$

where the primal and dual variables satisfy $z'_v = 1$, if $v \in S$, or 0 if $v \notin S$.

A.4 Proof for Chapter 5

A.4.1 Proof of Lemma 5.2.2

The first part of the proof follows along the same line as the corresponding proof of Ene et al. [83] which is based on a submodular auxiliary graph and the path-augmentation algorithm [123], described in what follows.

Let $G = (V, E)$ be a directed graph such that the vertex set V corresponds to the elements in $[N]$, and where the arc set may be written as $E = \cup_{r\in[R]} E_r$,

with E_r corresponding to a complete directed graph on the set of elements S_r incident to F_r. With each arc (u, v), we associate a capacity value based on a $y' \in \mathcal{B}$ according to $c(u, v) \triangleq \min\{f_r(S) - y'_r(S) : S \subseteq S_r, u \in S, v \notin S\}$.

Next, we consider a procedure termed path augmentations over G that sequentially transforms y' from $y' = y$ to a point in $\mathcal{B}$ that satisfies $Ay' = z$; the vector y' is kept within $\mathcal{B}$ during the whole procedure. Let the set of source and sink nodes of the graph be defined as $N \triangleq \{v \in [N] | (Ay')_v < z_v\}$ and $P \triangleq \{v \in [N] | (Ay')_v > z_v\}$, where z is as defined in the statement of the lemma. If $N = P = \emptyset$, we have $Ay' = z$. It can be shown that there always exists a directed path with positive capacity from N to P unless $N = P = \emptyset$ [83]. In each step, we find the shortest directed path, denoted by $\mathcal{Q}$, with positive capacity from N to P. For each arc (u, v) in $\mathcal{Q}$, if the arc belongs to E_r, we set $y'_{r,u} \leftarrow y'_{r,u} + \rho$, $y'_{r,v} \leftarrow y'_{r,v} - \rho$, where ρ denotes the smallest capacity of any arc in $\mathcal{Q}$. This procedure ensures that $y' \in \mathcal{B}$ and that the procedure terminates in a finite number of steps, with $N = P = \emptyset$ [123].

The second part of the proof differs from the derivations of Ene et al. [83]. Suppose that $\{y'^{(0)} = y, y'^{(1)}, ..., y'^{(t)}\}$ is a sequence such that $y'^{(i)}$ equals the vector y' after the i-th step of the above procedure. We also assume that $Ay'^{(t)} = z$, implying that the algorithm terminated at step t. Hence, the point $y'^{(t)}$ is the desired value of ξ. During path-augmentation, no element appears in more than two updated arcs. Hence,

$$\|y'^{(i)} - y'^{(i-1)}\|_{2,\theta} \leq \sqrt{2 \sum_v \max_{r \in [R]: v \in S_r} \theta_{r,v} \rho} = \sqrt{2 \|\theta\|_{1,\infty}} \rho.$$

As $\|Ay'^{(i)} - Ay'^{(i-1)}\|_1 = 2\rho$, we have

$$\|y'^{(i)} - y'^{(i-1)}\|_{2,\theta} \leq \sqrt{\frac{\|\theta\|_{1,\infty}}{2}} \|Ay'^{(i)} - Ay'^{(i-1)}\|_1.$$

An important observation is that during the path-augmentation procedure, for each component $v \in [N]$, the updated sequence $\{(Ay'^{(i)})_v\}_{i=1,2,..,t}$ converges monotonically to z_v. Hence, $\|Ay'^{(t)} - Ay'^{(0)}\|_1 = \sum_{i=1}^{t} \|Ay'^{(i)} -$

$Ay'^{(i-1)}\|_1$. By using the triangle inequality for the norm $\|\cdot\|_{2,\theta}$, we obtain

$$\begin{aligned}\sqrt{\frac{\|\theta\|_{1,\infty}}{2}}\|z-Ay\|_1 = \sqrt{\frac{\|\theta\|_{1,\infty}}{2}}\|Ay'^{(t)}-Ay'^{(0)}\|_1 &\geq \sum_{i=1}^{t}\|y'^{(i)}-y'^{(i-1)}\|_{2,\theta}\\ &\geq \|y'^{(t)}-y'^{(0)}\|_{2,\theta} = \|y'^{(t)}-y\|_{2,\theta}.\end{aligned}$$

Invoking the Cauchy-Schwarz inequality establishes $\|z-Ay\|_1 \leq \sqrt{\|w^{-1}\|_1}\|z-Ay\|_{2,w}$, which concludes the proof.

A.4.2 Proof for Lemma 5.2.3

The equivalence between problem (5.5) and problem (5.4) is easy to establish, as y is obtained from y' by simply removing its zero components. The second statement is proved as follows:

$$\begin{aligned}&\min_{y\in\mathcal{B}}\min_{a:Aa=0,a_{r,i}=0,\,\forall(r,i):i\notin S_r}\frac{1}{2}\|y-a\|^2_{2,I(\mu)}\\ =&\min_{y\in\mathcal{B}}\min_{a:a_{r,i}=0,\,\forall(r,i):i\notin S_r}\max_{\lambda\in\mathbb{R}^N}\frac{1}{2}\|y-a\|^2_{2,I(\mu)}-\langle\lambda,Aa\rangle\\ \overset{1)}{=}&\min_{y\in\mathcal{B}}\max_{\lambda\in\mathbb{R}^N}\min_{a\in\otimes_{r=1}^{R}R^N}\frac{1}{2}\sum_{r\in[R]}\sum_{i\in S_r}[\mu_i(y_{r,i}-a_{r,i})^2-2\lambda_i a_{r,i}]\\ =&\min_{y\in\mathcal{B}}\max_{\lambda\in\mathbb{R}^N}\frac{1}{2}\sum_{r\in[R]}\sum_{i\in S_r}[\mu_i^{-1}\lambda_i^2-2\lambda_i(\mu_i^{-1}\lambda_i+y_{r,i})]\\ =&\min_{y\in\mathcal{B}}\max_{\lambda\in\mathbb{R}^N}\frac{1}{2}\sum_{r\in[R]}\sum_{i\in S_r}(-\mu_i^{-1}\lambda_i^2-2\lambda_i y_{r,i})\\ \overset{2)}{=}&\min_{y\in\mathcal{B}}\max_{\lambda\in\mathbb{R}^N}-\frac{1}{2}\|\lambda\|_2^2-\langle\lambda,Ay\rangle\\ =&\min_{y\in\mathcal{B}}\|Ay\|_2^2,\end{aligned}$$

where 1) is obtained using the incidence relations $y_{r,i}=a_{r,i}=0$ if $i\notin S_r$ and 2) holds because $\mu_i=|\{r\in[R]|i\in S_r\}|$. The optimal y,a,λ satisfy $a_{r,i}=y_{r,i}+\mu_i^{-1}\lambda_i$ for all $i\in S_r$, $r\in[R]$ and $\lambda=-Ay$.

A.4.3 Proof of Lemma 5.2.5

First, consider a $y \in \mathcal{B}/\Xi$. We have $d_{I(\mu)}(y, \mathcal{Z}) = \|Ay + x^*\|_2$, since

$$\begin{aligned}
\frac{1}{2} d_{I(\mu)}(y, \mathcal{Z})^2 &= \min_{a \in \mathcal{Z}} \frac{1}{2} \|y - a\|_{2,I(\mu)}^2 \\
&= \min_{a: a_{r,i}=0,\, \forall (r,i): i \notin S_r} \max_{\lambda \in \mathbb{R}^N} \frac{1}{2} \|y - a\|_{2,I(\mu)}^2 - \langle \lambda, Aa + x^* \rangle \\
&\overset{1)}{=} \max_{\lambda \in \mathbb{R}^N} \min_{a \in \otimes_{r=1}^R \mathbb{R}^N} \frac{1}{2} \sum_{r \in [R]} \sum_{i \in S_r} [\mu_i (y_{r,i} - a_{r,i})^2 - 2\lambda_i a_{r,i}] - \langle \lambda, x^* \rangle \\
&= \max_{\lambda \in \mathbb{R}^N} \frac{1}{2} \sum_{r \in [R]} \sum_{i \in S_r} [-\mu_i^{-1} \lambda_i^2 - \lambda_i y_{r,i}] - \langle \lambda, x^* \rangle \\
&\overset{2)}{=} \max_{\lambda \in \mathbb{R}^N} -\frac{1}{2} \|\lambda\|_2^2 - \lambda^T (Ay + x^*) \\
&= \frac{1}{2} \|Ay + x^*\|_2^2,
\end{aligned}$$

where 1) is obtained using the incidence relations $y_{r,i} = a_{r,i} = 0$ if $i \notin S_r$ and 2) holds because $\mu_i = |\{r \in [R] | i \in S_r\}|$. Based on Lemma 5.2.2, we know that there exists a $\xi \in \mathcal{B}$ such that $A\xi = -x^*$ and

$$\|y - \xi\|_{2,I(\mu)} \leq \sqrt{\frac{N \|I(\mu)\|_{1,\infty}}{2}} \|Ay + x^*\|_2 = \sqrt{\frac{N \|\mu\|_1}{2}} \|Ay + x^*\|_2.$$

Therefore, $\kappa(y) = \frac{d_{I(\mu)}(y,\Xi)}{d_{I(\mu)}(y,\mathcal{Z})} \leq \sqrt{\frac{N\|\mu\|_1}{2}}$.

Next, consider a $y \in \mathcal{Z}/\Xi$. As $\mathcal{B}$ is compact, there exists a $y' \in \mathcal{B}$ that achieves $d_{I(\mu)}(y, \mathcal{B}) = \|y - y'\|_{2,I(\mu)}$. Based on Lemma 3.1, we also know that there exists a $\xi \in \mathcal{B}$ such that $A\xi = -x^*$ and

$$\|\xi - y'\|_{2,I(\mu)} \leq \sqrt{\frac{\|I(\mu)\|_{1,\infty}}{2}} \|Ay' + x^*\|_1 = \sqrt{\frac{\|\mu\|_1}{2}} \|Ay' + x^*\|_1.$$

Moreover, we have

$$\begin{aligned}
\|Ay' + x^*\|_1 =& \|Ay' - Ay\|_1 \leq \|y' - y\|_1 = \sum_{v \in [N]} \sum_{r: v \in S_r} |y'_{r,v} - y_{r,v}| \\
&\leq \sum_{v \in [N]} \left[\mu_v \sum_{r: v \in S_r} (y'_{r,v} - y_{r,v})^2 \right]^{\frac{1}{2}} \leq \sqrt{N} \|y' - y\|_{2,I(\mu)}.
\end{aligned}$$

As $\xi \in \Xi$, it holds that $d_{I(\mu)}(y, \Xi) \leq \|\xi - y\|_{2,I(\mu)} \leq \|y' - y\|_{2,I(\mu)} + \|y' - \xi\|_{2,I(\mu)}$.

In addition, as

$$\|y' - \xi\|_{2,I(\mu)} \leq \sqrt{\frac{\delta^s}{2}}\|Ay' + x^*\|_1 \leq \sqrt{\frac{N\|\mu\|_1}{2}}\|y' - y\|_{2,I(\mu)},$$

we know that $d_{I(\mu)}(y, \Xi) \leq (1 + \sqrt{\frac{N\|\mu\|_1}{2}})\|y' - y\|_{2,I(\mu)}$. Therefore,

$$\kappa(y) = \frac{d_{I(\mu)}(y, \Xi)}{d_{I(\mu)}(y, \mathcal{B})} \leq \left(1 + \sqrt{\frac{N\|\mu\|_1}{2}}\right),$$

which concludes the proof.

A.4.4 Proof of Lemma 5.2.10

Choose $z = Ay^*$ in Lemma 5.2.2. Then, there is a $\xi \in \mathcal{B}$ such that $\|Ay - Ay^*\|^2 \geq \frac{2}{N\|\theta^P\|_{1,\infty}}\|y - \xi\|^2_{2,\theta^P}$. Moreover as $A\xi = z = Ay^* = -x^*$, we also have $\xi \in \Xi$. Therefore, $\|y - \xi\|^2_{2,\theta^P} \geq \|y - y^*\|^2_{2,\theta^P}$, which concludes the proof.

A.4.5 Proof for Theorem 5.2.11

First, given a group of blocks C and $y \in \otimes_{r=1}^R \mathbb{R}^N$, we define $y_{[C]} \in \otimes_{r=1}^R \mathbb{R}^N$ as

$$(y_{[C]})_r = \begin{cases} y_r & \text{if } r \in C, \\ 0 & \text{if } r \notin C. \end{cases}$$

The following lemma holds.

Lemma A.4.1. *Let C be a group of blocks sampled according to a $\alpha-$proper distribution P. Then, for any $y \in \otimes_{r=1}^R \mathbb{R}^N$ and $y_{r,i} = 0$, whenever $i \notin S_r$, one has*

$$\mathbb{E}_{C\sim P}(\|y_{[C]}\|^2_{2,I(\mu^C)}) = \mathbb{E}_{C\sim P}(\|y_{[C]}\|^2_{2,\theta^P}).$$

Proof.

$$\mathbb{E}_{C\sim P}(\|y_{[C]}\|^2_{2,I(\mu^C)}) = \mathbb{E}_{C\sim P}(\sum_{r\in C}\|y_r\|^2_{2,\mu^C}) = \sum_{r\in[R]}\mathbb{E}_{C\sim P}\left[\|y_r\|^2_{2,\mu^C}1_{r\in C}\right]$$
$$= \sum_{r\in[R]}\mathbb{E}\left[1_{r\in C}\mathbb{E}_{C\sim P}\left[\|y_r\|^2_{2,\mu^C}|r\in C\right]\right] = \sum_{r\in[R]}\mathbb{E}\left[1_{r\in C}\|y_r\|^2_{2,\theta^P_r}\right]$$
$$=\mathbb{E}_{C\sim P}(\|y_{[C]}\|^2_{2,\theta^P}).$$

□

Next, we turn our attention to the proof of the theorem. For this purpose, suppose that $y^* = \arg\min_{y\in\Xi}\|y - y^{(k)}\|_{2,\theta^P}$. We start by establishing the following results.

Lemma A.4.2. *It can be shown that*

$$\langle\nabla g(y^{(k)}), y^* - y^{(k)}\rangle \overset{1)}{\leq} g(y^*) - g(y^{(k)}) - \frac{1}{N\|\theta^P\|_{1,\infty}}\|y^{(k)} - y^*\|^2_{2,\theta^P}$$
$$\overset{2)}{\leq} \frac{4}{N\|\theta^P\|_{1,\infty}+2}\left[g(y^*) - g(y^{(k)}) - \frac{1}{2}\|y^{(k)} - y^*\|^2_{2,\theta^P}\right]. \tag{A.17}$$

Proof. From Lemma 3.8 we can infer that

$$\|Ay^{(k)} - Ay^*\|_2^2 \geq \frac{2}{N\|\theta^P\|_{1,\infty}}\|y^{(k)} - y^*\|^2_{2,\theta^P} \Rightarrow$$
$$g(y^*) \geq g(y^{(k)}) + \langle\nabla g(y^{(k)}), y^* - y^{(k)}\rangle + \frac{1}{N\|\theta^P\|_{1,\infty}}\|y^{(k)} - y^*\|^2_{2,\theta^P}, \tag{A.18}$$
$$g(y^{(k)}) \geq g(y^*) + \langle\nabla g(y^*), y^{(k)} - y^*\rangle + \frac{1}{N\|\theta^P\|_{1,\infty}}\|y^{(k)} - y^*\|^2_{2,\theta^P}. \tag{A.19}$$

As $\langle\nabla g(y^*), y^{(k)} - y^*\rangle \geq 0$, (A.30) gives

$$g(y^*) - g(y^{(k)}) \leq -\frac{1}{N\|\theta^P\|_{1,\infty}}\|y^{(k)} - y^*\|^2_{2,\theta^P}. \tag{A.20}$$

The inequality (A.29) yields claim 1) in (A.28). Claim 2) in (A.28) follows from (A.31). □

The following lemma is a direct consequence of the optimality of $y_r^{(k+1)}$ for an oblique projection.

Lemma A.4.3.

$$\langle \nabla_r g(y^{(k)}), y_r^{(k+1)} - y_r^* \rangle \leq \langle y_r^{(k)} - y_r^{(k+1)}, y_r^{(k+1)} - y_r^* \rangle_{\theta_r^P}.$$

The following lemma follows from a simple manipulation of the Euclidean norm.

Lemma A.4.4.

$$\begin{aligned}
&\frac{1}{2}\|y_r^{(k+1)} - y_r^{(k)}\|_{2,\theta_r^P}^2 \\
=&\frac{1}{2}\|y_r^{(k+1)} - y_r^*\|_{2,\theta_r^P}^2 + \frac{1}{2}\|y_r^* - y_r^{(k)}\|_{2,\theta_r^P}^2 + \langle y_r^{(k+1)} - y_r^*, y_r^* - y_r^{(k)} \rangle_{\theta_r^P} \\
=&-\frac{1}{2}\|y_r^{(k+1)} - y_r^*\|_{2,\theta_r^P}^2 + \frac{1}{2}\|y_r^* - y_r^{(k)}\|_{2,\theta_r^P}^2 + \langle y_r^{(k+1)} - y_r^*, y_r^{(k+1)} - y_r^{(k)} \rangle_{\theta_r^P}.
\end{aligned}$$

Let us analyze next the amount by which the objective function decreases

in each iteration. The following expectation is with respect to $C_{i_k} \sim P$.

$$
\begin{aligned}
&\mathbb{E}\left[g(y^{(k+1)})\right] \qquad\qquad (A.21)\\
&\overset{1)}{\leq} g(y^{(k)}) + \mathbb{E}\left\{\sum_{r\in C_{i_k}}\left[\langle \nabla_r g(y^{(k)}), y_r^{(k+1)} - y_r^{(k)}\rangle + \frac{1}{2}\|y_r^{(k+1)} - y_r^{(k)}\|^2_{2,\mu_r^{C_{i_k}}}\right]\right\}\\
&\overset{2)}{=} g(y^{(k)}) + \mathbb{E}\left\{\sum_{r\in C_{i_k}}\left[\langle \nabla_r g(y^{(k)}), y_r^{(k+1)} - y_r^{(k)}\rangle + \frac{1}{2}\|y_r^{(k+1)} - y_r^{(k)}\|^2_{2,\theta_r^P}\right]\right\}\\
&= g(y^{(k)}) + \mathbb{E}\left\{\sum_{r\in C_{i_k}}\left[\langle \nabla_r g(y^{(k)}), y_r^{*} - y_r^{(k)}\rangle + \langle \nabla_r g(y^{(k)}), y_r^{(k+1)} - y_r^{*}\rangle \right.\right.\\
&\qquad\qquad \left.\left. + \frac{1}{2}\|y_r^{(k+1)} - y_r^{(k)}\|^2_{2,\theta_r^P}\right]\right\}\\
&\overset{3)}{\leq} g(y^{(k)}) + \mathbb{E}\left\{\sum_{r\in C_{i_k}}\left[\langle \nabla_r g(y^{(k)}), y_r^{*} - y_r^{(k)}\rangle - \frac{1}{2}\|y_r^{(k+1)} - y_r^{*}\|^2_{2,\theta_r^P} \right.\right.\\
&\qquad\qquad \left.\left. + \frac{1}{2}\|y_r^{*} - y_r^{(k)}\|^2_{2,\theta_r^P}\right]\right\}\\
&= g(y^{(k)}) + \alpha\langle \nabla g(y^{(k)}), y^* - y^{(k)}\rangle - \mathbb{E}\left[\frac{1}{2}\|y_{[C_{i_k}]}^{(k+1)} - y_{[C_{i_k}]}^{*}\|^2_{2,\theta^P}\right]\\
&\qquad\qquad + \mathbb{E}\left[\frac{1}{2}\|y_{[C_{i_k}]}^{(k)} - y_{[C_{i_k}]}^{*}\|^2_{2,\theta^P}\right]\\
&\overset{4)}{=} g(y^{(k)}) + \alpha\langle \nabla g(y^{(k)}), y^* - y^{(k)}\rangle - \mathbb{E}\left[\frac{1}{2}\|y^{(k+1)} - y^{*}\|^2_{2,\theta^P}\right] + \mathbb{E}\left[\frac{1}{2}\|y^{(k)} - y^{*}\|^2_{2,\theta^P}\right]\\
&\overset{5)}{\leq} g(y^*) - \mathbb{E}\left[\frac{1}{2}\|y^{(k+1)} - y^{*}\|^2_{2,\theta^P}\right]\\
&\qquad + \left[1 - \frac{4\alpha}{N\|\theta^P\|_{1,\infty} + 2}\right]\left\{g(y^{(k)}) - g(y^*) - \frac{1}{2}\|y^{(k)} - y^{*}\|^2_{2,\theta^P}\right\},
\end{aligned}
$$

(A.22)

where 1) follows from inequality (5.9), 2) holds due to Lemma A.4.1, 3) is a consequence of Lemma A.5.4 and Lemma A.5.5, 4) is due to $y_r^{(k+1)} = y_r^{(k)}$ for $r \notin C_{i_k}$, and 5) may be established from (A.28).

Equation (A.41) further establishes that

$$
\begin{aligned}
\mathbb{E}\left[g(y^{(k+1)}) - g(y^*) + \frac{1}{2}d^2_{\theta^P}(y^{k+1}, \xi)\right] \leq& \mathbb{E}\left[g(y^{(k+1)}) - g(y^*) + \frac{1}{2}\|y^{(k+1)} - y^{*}\|^2_{2,\theta^P}\right]\\
&\leq \left(1 - \frac{4\alpha}{N\|\theta^P\|_{1,\infty} + 2}\right)\mathbb{E}\left[g(y^{(k)}) - g(y^*) + \frac{1}{2}d^2_{\theta^P}(y^{k}, \xi)\right].
\end{aligned}
$$

The proof follows by repeating the derivations for all k.

A.4.6 Proof of Lemma 5.2.12

According to the definition of θ^P, we have

$$\begin{aligned}
\max_{r\in[R]:i\in S_r} \theta^P_{r,i} &= \max_{r\in[R]:i\in S_r} \mathbb{E}_{C\sim P}\left[\mu_i^C | r\in C\right] \\
&= \max_{r\in[R]:i\in S_r} \mathbb{E}_{C\sim P}\left[\sum_{r'\in[R]:i\in S_{r'}} 1_{r'\in C} | r\in C\right] \\
&= \max_{r\in[R]:i\in S_r} \sum_{r'\in[R]:i\in S_{r'}} \mathbb{P}_{C\sim P}\left[r'\in C | r\in C\right] \qquad \text{(A.23)} \\
&= \frac{1}{\alpha} \max_{r\in[R]:i\in S_r} \sum_{r'\in[R]:i\in S_{r'}} \mathbb{P}_{C\sim P}\left[r'\in C, r\in C\right] \\
&\geq \frac{1}{\alpha\mu_i} \sum_{r,r'\in[R]:i\in S_r,S_{r'}} \mathbb{P}_{C\sim P}\left[r'\in C, r\in C\right] \\
&= \frac{1}{\alpha\mu_i} \mathbb{E}_{C\sim P}\left[|\{(r,r')\in C\times C : i\in S_r, i\in S_{r'}\}|\right] \\
&= \frac{1}{\alpha\mu_i} \mathbb{E}_{C\sim P}\left[(\mu_i^C)^2\right] \\
&\geq \frac{1}{\alpha d_i}\left[\mathbb{E}_{C\sim P}(\mu_i^C)\right]^2 = \frac{1}{\alpha d_i}\left(\sum_C \sum_{r\in[R]:i\in S_r} 1_{r\in C}\mathbb{P}(C)\right)^2 \\
&= \frac{1}{\alpha\mu_i}\left(\sum_{r\in[R]:i\in S_r} \mathbb{P}_{C\sim P}[r\in C]\right)^2 \\
&= \frac{1}{\alpha\mu_i}\left(\alpha\mu_i\right)^2 = \alpha\mu_i.
\end{aligned}$$

From (A.23), we also have $\sum_{r'\in[R]:i\in S_{r'}} \mathbb{P}_{C\sim P}\left[r'\in C | r\in C\right] \geq \mathbb{P}_{C\sim P}\left[r\in C | r\in C\right] = 1$, which proves the claimed result.

A.4.7 Proof of Lemma 5.2.13

Similar to what was established for (A.23), one can show that

$$\theta^P_{r,i} = \sum_{r'\in[R]:i\in S_{r'}} \mathbb{P}_{C\sim P}\left[r'\in C | r\in C\right].$$

Consider next the right-hand side of this equation for $\alpha = \frac{K}{R}$. In this case, for some r and some $i \in S_r$, we have

$$\begin{aligned}\sum_{r' \in [R]: i \in S_{r'}} \mathbb{P}_{C \sim P}\left[r' \in C | r \in C\right] &= \mathbb{P}_{C \sim P}\left[r \in C | r \in C\right] \\ &\quad + \sum_{r' \in [R]: i \in S_{r'}, r' \neq r} \mathbb{P}_{C \sim P}\left[r' \in C | r \in C\right] \\ &= 1 + \frac{R}{K} \sum_{r': i \in S_{r'}, r' \neq r} \mathbb{P}_{C \sim P}\left[r' \in C, r \in C\right] \\ &= 1 + \frac{R}{K}(\mu_i - 1)\frac{\binom{R-2}{K-2}}{\binom{R}{K}} = 1 + \frac{K-1}{R-1}(\mu_i - 1).\end{aligned}$$

Therefore, $\theta^P_{r,i} = \frac{K-1}{R-1}\mu_i + \frac{R-K}{R-1}$ when P is a uniform distribution.

A.4.8 Proof of Theorem 5.2.14

We start by establishing a number of background results.

The following lemma is due to the optimality of $z_r^{(k+1)}$.

Lemma A.4.5.

$$\langle \nabla_r g(p^{(k)}), z_r^{(k+1)} - y_r^* \rangle \leq \frac{\lambda_k}{\alpha} \langle z_r^{(k)} - z_r^{(k+1)}, z_r^{(k+1)} - y_r^* \rangle_{\theta_r^P}.$$

Once again, one can easily establish the following result pertaining to the Euclidean norm.

Lemma A.4.6.

$$\begin{aligned}\frac{1}{2}\|z_r^{(k+1)} - z_r^{(k)}\|^2_{2,\theta_r^P} &= \frac{1}{2}\|z_r^{(k+1)} - y_r^*\|^2_{2,\theta_r^P} + \frac{1}{2}\|y_r^* - z_r^{(k)}\|^2_{2,\theta_r^P} \\ &\quad + \langle z_r^{(k+1)} - y_r^*, y_r^* - z_r^{(k)} \rangle_{\theta_r^P} \\ &= -\frac{1}{2}\|z_r^{(k+1)} - y_r^*\|^2_{2,\theta_r^P} + \frac{1}{2}\|y_r^* - z_r^{(k)}\|^2_{2,\theta_r^P} \\ &\quad + \langle z_r^{(k+1)} - y_r^*, z_r^{(k+1)} - z_r^{(k)} \rangle_{\theta_r^P}.\end{aligned}$$

The next result follows from the convexity property of the function g.

Lemma A.4.7.

$$\begin{aligned}\lambda_k\langle\nabla g(p^{(k)}), y^* - z^{(k)}\rangle &= \langle\nabla g(p^{(k)}), \lambda_k y^* - \lambda_k z^{(k)}\rangle \\ &= \langle\nabla g(p^{(k)}), \lambda_k y^* - (p^{(k)} - (1-\lambda_k)y^{(k)})\rangle \\ &= \lambda_k\langle\nabla g(p^{(k)}), y^* - p^{(k)}\rangle + (1-\lambda_k)\langle\nabla g(p^{(k)}), y^{(k)} - p^{(k)}\rangle \\ &\leq \lambda_k\left[g(y^*) - g(p^{(k)})\right] + (1-\lambda_k)\left[g(y^{(k)}) - g(p^{(k)})\right].\end{aligned}$$

We are now ready to analyze the decrease of the objective function in each iteration of Algorithm 5.2. The expectation in the following equations is performed with respect to $C_{i_k} \sim P$.

$$\begin{aligned}&\mathbb{E}\left[g(y^{(k+1)})\right] \\ &\overset{1)}{\leq} g(p^{(k)}) + \frac{\lambda_k}{\alpha}\mathbb{E}\left\{\sum_{r\in C_{i_k}}\left[\langle\nabla_r g(p^{(k)}), z_r^{(k+1)} - z_r^{(k)}\rangle + \frac{\lambda_k}{2\alpha}\|z_r^{(k+1)} - z_r^{(k)}\|_{2,\mu^C}^2\right]\right\} \\ &\overset{2)}{=} g(p^{(k)}) + \frac{\lambda_k}{\alpha}\mathbb{E}\left\{\sum_{r\in C_{i_k}}\left[\langle\nabla_r g(p^{(k)}), z_r^{(k+1)} - z_r^{(k)}\rangle + \frac{\lambda_k}{2\alpha}\|z_r^{(k+1)} - z_r^{(k)}\|_{2,\theta_r^P}^2\right]\right\} \\ &= g(p^{(k)}) + \frac{\lambda_k}{\alpha}\mathbb{E}\left\{\sum_{r\in C_{i_k}}\left[\langle\nabla_r g(p^{(k)}), y_r^* - z_r^{(k)}\rangle + \langle\nabla_r g(p^{(k)}), z_r^{(k+1)} - z_r^*\rangle\right.\right. \\ &\qquad\left.\left. + \frac{\lambda_k}{2\alpha}\|z_r^{(k+1)} - z_r^{(k)}\|_{2,\theta_r^P}^2\right]\right\} \\ &\overset{3)}{\leq} g(p^{(k)}) + \frac{\lambda_k}{\alpha}\mathbb{E}\left\{\sum_{r\in C_{i_k}}\left[\langle\nabla_r g(p^{(k)}), y_r^* - z_r^{(k)}\rangle - \frac{\lambda_k}{2\alpha}\|z_r^{(k+1)} - y_r^*\|_{2,\theta_r^P}^2\right.\right. \\ &\qquad\left.\left. + \frac{\lambda_k}{2\alpha}\|y_r^* - z_r^{(k)}\|_{2,\theta_r^P}^2\right]\right\} \\ &= g(p^{(k)}) + \lambda_k\langle\nabla g(p^{(k)}), y^* - z^{(k)}\rangle \\ &\qquad + \frac{\lambda_k^2}{2\alpha^2}\mathbb{E}\left[\|z_{[C_{i_k}]}^{(k)} - y_{[C_{i_k}]}^*\|_{2,\theta^P}^2 - \|z_{[C_{i_k}]}^{(k+1)} - y_{[C_{i_k}]}^*\|_{2,\theta^P}^2\right] \\ &\overset{4)}{=} g(p^{(k)}) + \lambda_k\langle\nabla g(p^{(k)}), y^* - z^{(k)}\rangle + \frac{\lambda_k^2}{2\alpha^2}\mathbb{E}\left[\|z^{(k)} - y^*\|_{2,\theta^P}^2 - \|z^{(k+1)} - y^*\|_{2,\theta^P}^2\right] \\ &\overset{5)}{=} g(y^*) + (1-\lambda_k)\left[g(y^{(k)}) - g(y^*)\right] \\ &\qquad + \frac{\lambda_k^2}{2\alpha^2}\left\{\|z^{(k)} - y^*\|_{2,\theta^P}^2 - \mathbb{E}\left[\|z^{(k+1)} - y^*\|_{2,\theta^P}^2\right]\right\}, \quad \text{(A.24)}\end{aligned}$$

where 1) follows from (5.9), 2) may be deduced from Lemma A.4.1, 3) is a

consequence of Lemma A.4.5 and Lemma A.4.6, 4) is due to the fact that $y_r^{(k+1)} = y_r^{(k)}$ for $r \notin C_{i_k}$, and 5) follows from Lemma A.4.7.

Based on the definition of $\{\lambda_k\}_{k\geq 0}$, we also have

$$\frac{1-\lambda_k}{\lambda_k^2} = \frac{1}{\lambda_{k-1}^2}, \quad 0 < \lambda_{k+1} \leq \lambda_k \leq \frac{2}{k+2/\lambda_0} = \frac{2}{k+2}. \tag{A.25}$$

Hence, combining the above expression with (A.24), for $k \in [1, \frac{2}{\alpha}\lceil\sqrt{N\|\theta^P\|_{1,\infty}}\rceil + 1]$, we have

$$\begin{aligned}
&\mathbb{E}\left[\frac{1-\lambda_k}{\lambda_k^2}\left[g(y^{(k)}) - g(y^*)\right] + \frac{1}{2\alpha^2}\|z^{(k)} - y^*\|_{2,\theta^P}^2\right] \\
&= \mathbb{E}\left[\frac{1}{\lambda_{k-1}^2}\left[g(y^{(k)}) - g(y^*)\right] + \frac{1}{2\alpha^2}\|z^{(k)} - y^*\|_{2,\theta^P}^2\right] \\
&\leq \mathbb{E}\left[\frac{1-\lambda_{k-1}}{\lambda_{k-1}^2}\left[g(y^{(k-1)}) - g(y^*)\right] + \frac{1}{2\alpha^2}\|z^{(k-1)} - y^*\|_{2,\theta^P}^2\right] \\
&\leq \cdots \leq \frac{(1-\lambda_0)}{\lambda_0^2}\left[g(y^{(0)}) - g(y^*)\right] + \frac{1}{2\alpha^2}\|z^{(0)} - y^*\|_{2,\theta^P}^2.
\end{aligned} \tag{A.26}$$

Lemma 5.2.10 implies the strong convexity property as

$$\begin{aligned}
\|Ay^{(k)} - Ay^*\|_2^2 &\geq \frac{2}{N\|\theta^P\|_{1,\infty}}\|y^{(k)} - y^*\|_{2,\theta^P}^2 \Rightarrow \\
g(y^{(k)}) - g(y^*) &\geq \langle \nabla g(y^*), y^{(k)} - y^*\rangle + \frac{1}{N\|\theta^P\|_{1,\infty}}\|y^{(k)} - y^*\|_{2,\theta^P}^2 \\
&\overset{1)}{\geq} \frac{1}{N\|\theta^P\|_{1,\infty}}\|y^{(k)} - y^*\|_{2,\theta^P}^2.
\end{aligned} \tag{A.27}$$

Here, 1) holds since y^* is an optimal solution of $\min_y g(y)$ and thus $\langle \nabla g(y^*), y^{(k)} - y^*\rangle \geq 0$.

Combining (A.25), (A.26) and (A.27), we obtain

$$\begin{aligned}
\mathbb{E}\left[g(y^{(k)}) - g(y^*)\right] &\leq \lambda_{k-1}^2\left[\frac{1-\lambda_0}{\lambda_0^2}(g(y^{(0)}) - g(y^*)) + \frac{1}{2\alpha^2}\|y^{(0)} - y^*\|_{2,\theta^P}^2\right] \\
&\leq \left(\frac{2}{k+1}\right)^2 \frac{1}{2\alpha^2}\|y^{(0)} - y^*\|_{2,\theta^P}^2 \\
&\leq \left(\frac{2}{k+1}\right)^2 \frac{N\|\theta^P\|_{1,\infty}}{2\alpha^2}(g(y^{(0)}) - g(y^*)).
\end{aligned}$$

Therefore, for $k = \left\lceil (1+c)\frac{\sqrt{2N\|\theta^P\|_{1,\infty}}}{\alpha} + c \right\rceil$, we have

$$\mathbb{E}\left[g(y^{\left(\left\lceil (1+c)\frac{\sqrt{2N\|\theta^P\|_{1,\infty}}}{\alpha} + c \right\rceil \right)}) - g(y^*) \right] \leq \frac{1}{1+c}(g(y^{(0)}) - g(y^*)).$$

For each value of $k = l \times \left\lceil (1+c)\frac{\sqrt{2N\|\theta^P\|_{1,\infty}}}{\alpha} + c \right\rceil$, $l \in \mathbb{Z}_{\geq 0}$, the values $z^{(k)}$, λ_k are reinitialized. Using a similar proof as above, we have

$$\mathbb{E}\left[g(y^{\left((l+1)\times \left\lceil (1+c)\frac{\sqrt{2N\|\theta^P\|_{1,\infty}}}{\alpha} + c \right\rceil \right)}) - g(y^*) \right]$$
$$\leq \frac{1}{1+c}\left[g(y^{\left(l\times \left\lceil (1+c)\frac{\sqrt{2N\|\theta^P\|_{1,\infty}}}{\alpha} + c \right\rceil \right)}) - g(y^*) \right].$$

Therefore,

$$\mathbb{E}\left[g(y^{\left(l\left\lceil (1+c)\frac{\sqrt{2N\|\theta^P\|_{1,\infty}}}{\alpha} + c \right\rceil \right)}) - g(y^*) \right] \leq \frac{1}{(1+c)^l}(g(y^{(0)}) - g(y^*)).$$

This concludes the proof.

A.4.9 Avoiding full-dimensional vector operations for parallel ACDM

Algorithm 5.2 can be implemented without full-dimensional vector operations. In each step, only those coordinates within the blocks in C are updated. Consequently, one only needs to replace $p^{(k)}$ and $y^{(k)}$ with $p^{(k)} = z^{(k)} + \lambda_k^2 u^{(k)}$ and $y^{(k)} = z^{(k)} + \lambda_{k-1}^2 u^{(k)}$, where $u^{(k)}$ is a new variable described in Algorithm A.1.

Algorithm A.1: Parallel ACDM (an efficient implementation)

Input: $\mathcal{B}$, α

0: Initialize $z^{(0)} \in \mathcal{B}$, $u^{(0)} \leftarrow 0 \in \mathbb{R}^N$, $k \leftarrow 0$.

1: Do the following steps iteratively until the dual gap $< \epsilon$:

2: If $k = l \left\lceil (1+c)\sqrt{\frac{2N\|\theta^P\|_{1,\infty}}{\alpha}} + c \right\rceil$ for some $l \in \mathbb{Z}$ and $c > 0$,
$z^{(k)} \leftarrow z^{(k)} + \lambda_{k-1}^2 u^{(k)}$, $u^{(k)} \leftarrow 0$, $\lambda_k \leftarrow 1$

3: Sample one set C_{i_k} according to a α-proper distribution P

4: For $r \in C_{i_k}$:

5: $\triangle z_r \leftarrow \arg\min_{\triangle z + z_r^{(k)} \in B_r} \|\triangle z + \frac{\alpha}{\lambda_k}(\theta_r^P)^{-1} \odot \nabla_r g(z^{(k)} + \lambda_k^2 u^{(k)})\|_{2,\theta_r^P}^2$

6: $z_r^{(k+1)} \leftarrow z_r^{(k)} + \triangle z_r$

7: $u_r^{(k+1)} \leftarrow u_r^{(k)} + \frac{\lambda_k - \alpha}{\alpha \lambda_k^2} \triangle z_r$

8: $\lambda_{k+1} \leftarrow \frac{\sqrt{\lambda_k^4 + 4\lambda_k^2} - \lambda_k^2}{2}$

9: $k \leftarrow k + 1$

10: Output $y^{(k)}$

A.5 Proof for Chapter 6

A.5.1 Proofs of the results pertaining to the dual formulation

Proof of Lemma 6.1.1

In all derivations that follow, we may exchange the order of minimization and maximization (i.e., $\min\max = \max\min$) due to Proposition 2.2 [124]. Plugging equation (6.2) into (6.1), we obtain

$$\begin{aligned}
&\min_x \sum_{r\in[R]} [f_r(x)]^2 + \|x-a\|_W^2 \\
=&\min_x \max_{\phi_r \geq 0, y_r \in \phi_r B_r} \sum_{r\in[R]} \left[\langle y_r, x\rangle - \frac{\phi_r^2}{4}\right] + \|x-a\|_W^2 \\
=&\max_{\phi_r \geq 0, y_r \in \phi_r B_r} \min_x \sum_{r\in[R]} \left[\langle y_r, x\rangle - \frac{\phi_r^2}{4}\right] + \|x-a\|_W^2 \\
=&\max_{\phi_r \geq 0, y_r \in \phi_r B_r} -\frac{1}{4}\|\sum_{r\in[R]} y_r - 2Wa\|_{W^{-1}}^2 - \frac{1}{4}\sum_r \phi_r^2 + \|a\|_W^2.
\end{aligned}$$

By eliminating some constants, one obtains the dual (6.3).

Next, we prove that the problem (6.4) is equivalent to (6.3) when removing λ_r.

First, (6.4) is equivalent to

$$
\begin{aligned}
&\min_{\phi_r \geq 0, y_r \in \phi_r B_r, \lambda_r} \max_{\lambda} \sum_{r\in[R]} \left[\|y_r - \frac{\lambda_r}{\sqrt{R}}\|_{W^{-1}}^2 + \phi_r^2 \right] + \left\langle \lambda, \sum_{r\in[R]} \lambda_r - 2Wa \right\rangle \\
&= \min_{\phi_r \geq 0, y_r \in \phi_r B_r} \max_{\lambda} \min_{\lambda_r} \sum_{r\in[R]} \left[\|y_r - \frac{\lambda_r}{\sqrt{R}}\|_{W^{-1}}^2 + \phi_r^2 \right] + \left\langle \lambda, \sum_{r\in[R]} \lambda_r - 2Wa \right\rangle \\
&= \min_{\phi_r \geq 0, y_r \in \phi_r B_r} \max_{\lambda} \sum_{r\in[R]} \left[\frac{1}{4}\|\lambda\|_W^2 + \phi_r^2 \right] + \left\langle \lambda, \sqrt{R} \sum_{r\in[R]} (y_r - \frac{1}{2} W\lambda_r) - 2Wa \right\rangle \\
&= \min_{\phi_r \geq 0, y_r \in \phi_r B_r} \max_{\lambda} -\frac{R}{4}\|\lambda\|_W^2 + \sqrt{R} \left\langle \lambda, \sum_{r\in[R]} y_r - 2Wa \right\rangle + \sum_{r\in[R]} \phi_r^2 \\
&= \min_{\phi_r \geq 0, y_r \in \phi_r B_r} \| \sum_{r\in[R]} y_r - 2Wa\|_{W^{-1}}^2 + \sum_{r\in[R]} \phi_r^2,
\end{aligned}
$$

which is equivalent to (6.3).

Proof of Lemma 6.1.2

We start by recalling the following lemma from [70] that characterizes the geometric structure of the product base polytope.

Lemma A.5.1 ([70])**.** *Assume that $W \in \mathbb{R}^{N\times N}$ is a positive diagonal matrix. Let $y \in \bigotimes_{r\in[R]} \phi_r' B_r$ and let s be in the base polytope of the submodular function $\sum_r \phi_r' F_r$. Then, there exists a point $y' \in \bigotimes_{r\in[R]} \phi_r' B_r$ such that $\sum_{r\in[R]} y_r' = s$ and $\|y' - y\|_{I(W)} \leq \sqrt{\frac{\sum_{i=1}^N W_{ii}}{2}} \| \sum_{r\in[R]} y_r - s\|_1$.*

Lemma A.5.1 cannot be used directly to prove Lemma 6.1.2, since y, y' are in different product base polytopes, $\bigotimes_{r\in[R]} \phi_r B_r$ and $\bigotimes_{r\in[R]} \phi_r' B_r$, respectively. However, the following lemma shows that one can transform y to lie in the same base polytopes that contains y'.

Lemma A.5.2. *For a given feasible point $(y, \phi) \in \bigotimes_{r\in[R]} C_r$, and a nonnegative vector $\phi' = (\phi_r) \in \bigotimes_{r\in[R]} \mathbb{R}_{\geq 0}$, one has*

$$
\| \sum_{r\in[R]} y_r - s\|_1 + \frac{\rho}{2}\|\phi' - \phi\| \geq \| \sum_{r\in[R]} \frac{\phi_r'}{\phi_r} y_r - s\|_1.
$$

Proof. For all r, let $\tilde{y}_r = y_r/\phi_r \in B_r$, and define a function that depends on

ϕ,

$$h(\phi) = \| \sum_{r\in[R]} y_r - s\|_1 = \| \sum_{r\in[R]} \phi_r \tilde{y}_r - s\|_1.$$

For all ϕ and r, $|\nabla_{\phi_r} h(\phi)| \leq \|\tilde{y}_r\|_1$. Therefore,

$$\begin{aligned} h(\phi') &= h(\phi) + \int_{t=0}^{1} \langle \nabla h|_{\phi+t(\phi'-\phi)}, t(\phi'-\phi)\rangle dt \\ &\geq h(\phi) - \frac{\max_{t\in[0,1]} \|\nabla h|_{\phi+t(\phi'-\phi)}\|}{2}\|\phi'-\phi\| \\ &\geq h(\phi) - \frac{\max_{\tilde{y}_r\in B_r,\forall r} \sqrt{\sum_{r\in[R]} \|\tilde{y}_r\|_1^2}}{2}\|\phi'-\phi\| = h(\phi) - \frac{\rho}{2}\|\phi'-\phi\|. \end{aligned}$$

□

Combining Lemma A.5.2 with Lemma A.5.1, we can establish the claim of Lemma 6.1.2.

First, let $\rho(W^{(1)}) = \max_{y\in\bigotimes_{r\in[R]} B_r} \sqrt{\sum_{r\in[R]} \|y_r\|_{W_r}^2}$. Suppose that $y' \in \bigotimes_{r\in[R]} \phi'_r B_r$ is such that $\sum_{r\in[R]} y'_r = s$ and it minimizes $\sum_{r\in[R]} \|\frac{\phi'_r}{\phi_r} y_r - y'_r\|_{W^{(1)}}^2$. As s lies in the base polytope of $\sum_{r\in[R]} \phi'_r F_r$, we know that such an y' exists. Moreover, we have

$$\begin{aligned} \|y-y'\|_{I(W^{(1)})} &\leq \sum_{r\in[R]} \|y'_r - \frac{\phi'_r}{\phi_r} y_r\|_{W^{(1)}} + \sum_{r\in[R]} \|y_r - \frac{\phi'_r}{\phi_r} y_r\|_{W^{(1)}} \\ \overset{1)}{\leq} & \sqrt{\frac{\sum_{i\in[N]} W^{(1)}_{ii}}{2}} \| \sum_{r\in[R]} \frac{\phi'_r}{\phi_r} y_r - s\|_1 + \rho(W^{(1)})\|\phi'-\phi\| \\ \overset{2)}{\leq} & \sqrt{\frac{\sum_{i\in[N]} W^{(1)}_{ii}}{2}} \left[\| \sum_{r\in[R]} y_r - s\|_1 + \frac{\rho}{2}\|\phi'-\phi\| \right] + \rho(W^{(1)})\|\phi'-\phi\| \\ = & \sqrt{\frac{\sum_{i\in[N]} W^{(1)}_{ii} \sum_{j\in[N]} 1/W^{(2)}_{jj}}{2}} \| \sum_{r\in[R]} y_r - s\|_{W^{(2)}} \\ & + \left[\sqrt{\frac{\sum_{i\in[N]} W^{(1)}_{ii}}{2}} \frac{\rho}{2} + \rho(W^{(1)}) \right] \|\phi'-\phi\| \\ \overset{3)}{\leq} & \sqrt{\frac{\sum_{i\in[N]} W^{(1)}_{ii} \sum_{j\in[N]} 1/W^{(2)}_{jj}}{2}} \| \sum_{r\in[R]} y_r - a\|_{W^{(2)}} + \frac{3}{2}\sqrt{\frac{\sum_{i\in[N]} W^{(1)}_{ii}}{2}} \rho\|\phi'-\phi\|, \end{aligned}$$

where 1) follows from Lemma A.5.1 and the definition of $\rho(W^{(1)})$, 2) is a consequence of Lemma A.5.2 and 3) holds because

$$\sum_{i\in[N]} W_{ii}^{(1)} \sum_{r\in[R]} \|y_r\|_1^2 \geq \sum_{r\in[R]} \|y_r\|_{W_r}^2.$$

A.5.2 Proof for the linear convergence rate of the RCD algorithm

Proof of Lemma 6.2.1

If (y^*, ϕ^*) is the optimal solution, then it must hold that $\sum_{r\in[R]} y_r^* = 2W(a - x^*)$ because of the duality between the primal and dual variables. Moreover, we also know that there must exist a nonempty collection $\mathcal{Y}$ of points $y' \in \bigotimes_{r\in[R]} \phi_r^* B_r$ such that $\sum_{r\in[R]} y_r' = \sum_{r\in[R]} y_r^*$. Using Lemma 6.1.2, and setting $\phi' = \phi^*, s = 2W(a - x^*),\ W^{(1)}, W^{(2)} = W^{-1}$, we can show that there exists some $y' \in \mathcal{Y}$ such that

$$\|y - y'\|_{I(W^{-1})}^2 + \|\phi - \phi^*\|^2 \leq \mu(W^{-1}, W^{-1}) \left[\| \sum_{r\in[R]} (y_r - y_r')\|_{W^{-1}}^2 + \|\phi - \phi^*\|^2 \right].$$

According to the definition of y^*, one has $\|y - y^*\|_{I(W^{-1})}^2 \leq \|y - y'\|_{I(W^{-1})}^2$ for $y' \in \mathcal{Y}$. This concludes the proof.

Proof of Theorem 6.2.2

First, suppose that $(y^*, \phi^*) = \arg\min_{(y',\phi')\in\Xi} \|y^{(k)} - y'\|_{I(W^{-1})}^2 + \|\phi^{(k)} - \phi'\|^2$. Throughout the proof, for simplicity, we use μ to denote $\mu(W^{-1}, W^{-1})$. We start by establishing the following results.

Lemma A.5.3. *It can be shown that the following inequalities hold:*

$$\begin{aligned}
&\langle \nabla g(y^{(k)}, \phi^{(k)}), (y^* - y^{(k)}, \phi^* - \phi^{(k)}) \rangle \\
&\overset{1)}{\leq} g(y^*, \phi^*) - g(y^{(k)}, \phi^{(k)}) - \frac{1}{\mu} \left(\|y^{(k)} - y^*\|^2_{I(W^{-1})} + \|\phi^{(k)} - \phi^*\|^2 \right) \\
&\overset{2)}{\leq} \frac{2}{\mu + 1} \left[g(y^*, \phi^*) - g(y^{(k)}, \phi^{(k)}) - \|y^{(k)} - y^*\|^2_{I(W^{-1})} - \|\phi^{(k)} - \phi^*\|^2 \right].
\end{aligned} \tag{A.28}$$

Proof. From Lemma 6.2.1, we can infer that

$$\begin{aligned}
&\| \sum_{r \in [R]} (y_r - y_r^*) \|^2_{W^{-1}} + \|\phi - \phi^*\|^2 \geq \frac{1}{\mu} \left[\|y - y^*\|^2_{I(W^{-1})} + \|\phi - \phi^*\|^2 \right] \Rightarrow \\
&g(y^*, \phi^*) \geq g(y^{(k)}, \phi^{(k)}) + \langle \nabla g(y^{(k)}, \phi^{(k)}), (y^* - y^{(k)}, \phi^* - \phi^{(k)}) \rangle \\
&\qquad\qquad + \frac{1}{\mu} \left[\|y - y^*\|^2_{I(W^{-1})} + \|\phi - \phi^*\|^2 \right], \qquad \text{(A.29)} \\
&g(y^{(k)}, \phi^{(k)}) \geq g(y^*, \phi^*) + \langle \nabla g(y^*, \phi^*), (y^{(k)} - y^*, \phi^{(k)} - \phi^*) \rangle \\
&\qquad\qquad + \frac{1}{\mu} \left[\|y - y^*\|^2_{I(W^{-1})} + \|\phi - \phi^*\|^2 \right]. \qquad \text{(A.30)}
\end{aligned}$$

As $\langle \nabla g(y^*, \phi^*), (y^{(k)} - y^*, \phi^{(k)} - \phi^*) \rangle \geq 0$, (A.30) gives

$$g(y^*, \phi^*) - g(y^{(k)}, \phi^{(k)}) \leq -\frac{1}{\mu} \left[\|y - y^*\|^2_{I(W^{-1})} + \|\phi^{(k)} - \phi^*\|^2 \right]. \tag{A.31}$$

Inequality (A.29) establishes Claim 1) in (A.28). Claim 2) in (A.28) follows from (A.31). □

The following lemma is a direct consequence of the optimality of $y_r^{(k+1)}$ as the projection Π_{C_r}.

Lemma A.5.4. *One has*

$$\begin{aligned}
&\langle \nabla_r g((y^{(k)}, \phi^{(k)})), (y_r^{(k+1)} - y_r^*, \phi_r^{(k+1)} - \phi_r^*) \rangle \\
&\qquad \leq 2 \langle y_r^{(k)} - y_r^{(k+1)}, y_r^{(k+1)} - y_r^* \rangle_{W^{-1}} + 2 \langle \phi_r^{(k)} - \phi_r^{(k+1)}, \phi_r^{(k+1)} - \phi_r^* \rangle.
\end{aligned}$$

The following lemma follows from a simple manipulation of the Euclidean norm.

Lemma A.5.5. *It holds that*

$$\begin{aligned}
&\|y_r^{(k+1)} - y_r^{(k)}\|_{W^{-1}}^2 + (\phi_r^{(k+1)} - \phi_r^{(k)})^2 \\
&\quad = \|y_r^{(k+1)} - y_r^*\|_{W^{-1}}^2 + (\phi_r^{(k+1)} - \phi_r^*)^2 + \|y_r^{(k)} - y_r^*\|_{W^{-1}}^2 + (\phi_r^{(k)} - \phi_r^*)^2 \\
&\qquad + 2\langle y_r^{(k+1)} - y_r^*, y_r^* - y_r^{(k)}\rangle_{W^{-1}} + 2\langle \phi_r^{(k+1)} - \phi_r^*, \phi_r^* - \phi_r^{(k)}\rangle \\
&\quad = -\|y_r^{(k+1)} - y_r^*\|_{W^{-1}}^2 - (\phi_r^{(k+1)} - \phi_r^*)^2 + \|y_r^{(k)} - y_r^*\|_{W^{-1}}^2 + (\phi_r^{(k)} - \phi_r^*)^2 \\
&\qquad + 2\langle y_r^{(k+1)} - y_r^*, y_r^{(k+1)} - y_r^{(k)}\rangle_{W^{-1}} + 2\langle \phi_r^{(k+1)} - \phi_r^*, \phi_r^{(k+1)} - \phi_r^{(k)}\rangle.
\end{aligned}$$

Our next task is to determine how the objective function decreases in each iteration. The following expectation is with respect to uniformly sampled values of $r \in [R]$ in the k-th iteration:

$$\begin{aligned}
&\mathbb{E}\left[g(y^{(k+1)}, \phi^{(k+1)})\right] - g(y^{(k)}, \phi^{(k)}) \\
&= \mathbb{E}\left[\langle \nabla_r g(y^{(k)}, \phi^{(k)}), (y_r^{(k+1)} - y_r^{(k)}, \phi_r^{(k+1)} - \phi_r^{(k)})\rangle\right. \\
&\qquad \left. + \|y_r^{(k+1)} - y_r^{(k)}\|_{W^{-1}}^2 + (\phi_r^{(k+1)} - \phi_r^{(k)})^2\right] \\
&= \mathbb{E}\left[\langle \nabla_r g(y^{(k)}, \phi^{(k)}), (y_r^* - y_r^{(k)}, \phi_r^* - \phi_r^{(k)})\rangle\right. \\
&\qquad + \langle \nabla_r g(y^{(k)}, \phi^{(k)}), (y_r^{(k+1)} - y_r^*, \phi_r^{(k+1)} - \phi_r^*)\rangle \\
&\qquad \left. + \|y_r^{(k+1)} - y_r^{(k)}\|_{W^{-1}}^2 + (\phi_r^{(k+1)} - \phi_r^{(k)})^2\right] \\
&\overset{1)}{\leq} \mathbb{E}\left[\langle \nabla_r g(y^{(k)}, \phi^{(k)}), (y_r^* - y_r^{(k)}, \phi_r^* - \phi_r^{(k)})\rangle - \|y_r^{(k+1)} - y_r^*\|_{W^{-1}}^2 + \|y_r^* - y_r^{(k)}\|_{W^{-1}}^2\right. \\
&\qquad \left. - (\phi_r^{(k+1)} - \phi_r^*)^2 + (\phi_r^* - \phi_r^{(k)})^2\right] \\
&\overset{2)}{\leq} \frac{1}{R}\langle \nabla g(y^{(k)}, \phi^{(k)}), (y^* - y^{(k)}, \phi^* - \phi^{(k)})\rangle \\
&\qquad - \mathbb{E}\left[\|y^{(k+1)} - y^*\|_{I(W^{-1})}^2 + \|\phi^{(k+1)} - \phi^*\|^2\right] \\
&\qquad + \|y^{(k)} - y^*\|_{I(W^{-1})}^2 + \|\phi^{(k)} - \phi^*\|^2 \qquad \text{(A.32)} \\
&\overset{3)}{\leq} \frac{2}{(\mu+1)R}\left[g(y^*, \phi^*) - g(y^{(k)}, \phi^{(k)})\right] \\
&\qquad + \left(1 - \frac{2}{(\mu+1)R}\right)\left[\|y^{(k)} - y^*\|_{I(W^{-1})}^2 + \|\phi^{(k)} - \phi^*\|^2\right] \\
&\qquad - \mathbb{E}\left[\|y^{(k+1)} - y^*\|_{I(W^{-1})}^2 + \|\phi^{(k+1)} - \phi^*\|^2\right]. \qquad \text{(A.33)}
\end{aligned}$$

Here, 1) is a consequence of Lemma A.5.4 and Lemma A.5.5, 2) is due to $y_{r'}^{(k+1)} = y_{r'}^{(k)}$, $\phi_{r'}^{(k+1)} = \phi_{r'}^{(k)}$ for $r' \neq r$, and 3) may be established from (A.28).

Equation (A.41) further leads to

$$\begin{aligned}
&\mathbb{E}\left[g(y^{(k+1)},\phi^{k+1}) - g(y^*,\phi^*) + d^2((y^{(k+1)},\phi^{(k+1)}),\Xi)\right] \\
\leq\ & \mathbb{E}\left[g(y^{(k+1)},\phi^{k+1}) - g(y^*,\phi^*) + \|y^{(k+1)} - y^*\|^2_{I(W^{-1})} + \|\phi^{(k+1)} - \phi^*\|^2_{I(W^{-1})}\right] \\
\leq\ & \left[1 - \frac{2}{(\mu+1)R}\right]\mathbb{E}\left[g(y^{(k)},\phi^k) - g(y^*,\phi^*) + d^2((y^{(k)},r^{(k)}),\Xi)\right].
\end{aligned}$$

The claimed proof follows by iterating the above derivations for all values of k.

A.5.3 Convergence analysis of the AP algorithm

Proof of Lemma 6.2.3

First, for $r \in [R]$, we define a diagonal matrix $A_r \in \mathbb{R}^{N\times N}$: $(A_r)_{ii} = 1$, if $i \in S_r$, and 0 otherwise. Start with a Lagrangian dual of (6.8), and transform it according to

$$\begin{aligned}
&\min_{(y,\phi)\in\bigotimes_{r\in[R]} C_r}\ \min_{\Lambda:\lambda_{r,i}=0,\,\forall(i,r)\notin\mathbb{S}}\ \max_{\alpha\in\mathbb{R}^N}\quad \sum_{r\in[R]}\left[\|y_r-\lambda_r\|^2_{\Psi W^{-1}} + \phi_r^2\right] \\
&\qquad\qquad + \langle\alpha, \sum_{r\in[R]}\lambda_r - 2Wa\rangle \\
=\ &\min_{(y,\phi)\in\bigotimes_{r\in[R]} C_r}\ \max_{\alpha\in\mathbb{R}^N}\ \min_{\Lambda:\lambda_{r,i}=0,\,\forall(i,r)\notin\mathbb{S}}\quad \sum_{r\in[R]}\left[\|y_r-\lambda_r\|^2_{\Psi W^{-1}} + \phi_r^2\right] \\
&\qquad\qquad + \langle\alpha, \sum_{r\in[R]}\lambda_r - 2Wa\rangle \\
\overset{1)}{=}\ &\min_{(y,\phi)\in\bigotimes_{r\in[R]} C_r}\ \max_{\alpha\in\mathbb{R}^N}\quad \sum_{r\in[R]}\left[\frac{1}{4}\|A_r\Psi^{-1}W\alpha\|^2_{\Psi W^{-1}} + \phi_r^2\right] \\
&\qquad\qquad + \left\langle\alpha, \sum_{r\in[R]}\left(y_r - \frac{1}{2}A_r\Psi^{-1}W\alpha\right) - 2Wa\right\rangle \\
\overset{2)}{=}\ &\min_{(y,\phi)\in\bigotimes_{r\in[R]} C_r}\ \max_{\alpha\in\mathbb{R}^N}\quad -\frac{1}{4}\|\alpha\|^2_W + \left\langle\alpha, \sum_{r\in[R]} y_r - 2Wa\right\rangle + \sum_{r\in[R]}\phi_r^2 \\
=\ &\min_{(y,\phi)\in\bigotimes_{r\in[R]} C_r}\quad \|\sum_{r\in[R]} y_r - 2Wa\|^2_{W^{-1}} + \sum_{r\in[R]}\phi_r^2.
\end{aligned}$$

Here, 1) is due to $\lambda_r = y_r - \frac{1}{2}A_r\Psi^{-1}W\alpha$ while 2) is based on the fact that $\Psi = \sum_{r\in[R]} A_r$. This establishes the claimed result.

Proof of Lemma 6.2.6

Suppose that $(y,\phi) \in \mathcal{C}/\Xi$. Then,

$$[d_{\Psi W^{-1}}((y,\phi),\mathcal{Z})]^2 = \min_{\lambda_r,\forall r\in[R]} \sum_r \left[\|y_r - \lambda_r\|^2_{\Psi W^{-1}} + (\phi_r - \phi_r^*)^2\right]$$
$$\text{s.t. } \sum_{r\in[R]} \lambda_r = 2W(a - x^*),\ \lambda_{r,i} = 0, \forall r \in [R], i \notin S_r.$$

By eliminating λ_r, we arrive at

$$[d_{\Psi W^{-1}}((y,\phi),\mathcal{Z})]^2 = \|\sum_r y_r - 2W(a-x^*)\|^2_{W^{-1}} + \sum_r (\phi_r - \phi_r^*)^2.$$

Based on Lemma 6.1.2, we know that there exists a $(y',\phi') \in \Xi$ such that

$$\mu(\Psi W^{-1}, W^{-1}) \left[\|\sum_r (y_r - y_r')\|^2_{W^{-1}} + \sum_r (\phi_r - \phi_r')^2\right]$$
$$\geq \|y - y'\|^2_{\Psi W^{-1}} + \sum_r (\phi_r - \phi_r')^2.$$

As ϕ_r^* is the unique optimum, it follows that $\phi_r^* = \phi_r'$. Also, $\sum_r y_r' = 2W(a - x^*)$. Moreover, as

$$\|y - y'\|^2_{\Psi W^{-1}} + \sum_r (\phi_r - \phi_r')^2 \geq [d_{\Psi W^{-1}}((y,\phi),\Xi)]^2$$

according to the above definition, we have

$$\frac{[d_{\Psi W^{-1}}((y,\phi),\Xi)]^2}{[d_{\Psi W^{-1}}((y,\phi),\mathcal{Z})]^2} \leq \mu(\Psi W^{-1}, W^{-1}).$$

Next, suppose that $(y,\phi) \in \mathcal{Z}/\Xi$ and that

$$(y',\phi') = \arg\min_{(z,\psi)\in\mathcal{C}} \|y - z\|^2_{I(\Psi W^{-1})} + \|\phi - \psi\|^2,$$

$$(y'',\phi'') = \arg\min_{(z,\psi)\in\Xi} \|y' - z\|^2_{I(\Psi W^{-1})} + \|\phi' - \psi\|^2.$$

Again, due to the definition of the distance $d_{\Psi W^{-1}}((y,\phi),\Xi)$, we have

$$[d_{\Psi W^{-1}}((y,\phi),\Xi)]^2 \leq \|y-y''\|^2_{I(\Psi W^{-1})} + \|\phi-\phi''\|^2. \tag{A.34}$$

Moreover, because of the way we chose (y',ϕ') and due to the fact that $\mathcal{C}$ is convex, we have

$$\begin{aligned}\|y-y''\|^2_{I(\Psi W^{-1})} + \|\phi-\phi''\|^2 \leq \|y-y'\|^2_{I(\Psi W^{-1})} + \|\phi-\phi'\|^2 \\ + \|y'-y''\|^2_{I(\Psi W^{-1})} + \|\phi'-\phi''\|^2.\end{aligned} \tag{A.35}$$

Using Lemma 6.1.2, we obtain

$$\begin{aligned}&\|y'-y''\|^2_{I(\Psi W^{-1})} + \|\phi'-\phi''\|^2 \\ &\quad\leq \mu(\Psi W^{-1}, W^{-1})(\|\sum_r (y'_r - y''_r)\|^2_{W^{-1}} + \|\phi'-\phi''\|^2),\end{aligned} \tag{A.36}$$

and we also have

$$\begin{aligned}\|\sum_r (y'_r - y''_r)\|^2_{W^{-1}} &= \|\sum_r y'_r - 2W(a-x^*)\|^2_{W^{-1}} \\ =\|\sum_r (y'_r - y_r)\|^2_{W^{-1}} &\overset{1)}{\leq} \|(y'-y)\|^2_{I(\Psi W^{-1})},\end{aligned} \tag{A.37}$$

where 1) follows from the Cauchy-Schwarz inequality over the entries $y_{r,i}$, $i \in S_r$.

Finally, we set $\phi = \phi'' = \phi^*$ and combine (A.34)-(A.37) to obtain

$$\begin{aligned}[d_{\Psi W^{-1}}((y,\phi),\Xi)]^2 &\leq (1+\mu(\Psi W^{-1}, W^{-1}))(\|y-y'\|^2_{I(\Psi W^{-1})} + \|\phi-\phi'\|^2) \\ &= (1+\mu(\Psi W^{-1}, W^{-1}))[d_{\Psi W^{-1}}((y,\phi),\mathcal{C})]^2,\end{aligned}$$

which concludes the proof.

A.5.4 Proof of Corollary 6.2.7

First, we establish an upper bound on ρ.

Lemma A.5.6. *Suppose that* $D_{ii} = \sum_{r:r\in[R],i\in C_r} \max_{S\subseteq V}[F_r(S)]^2$. *Then*

$$\rho^2 \leq 4\sum_{i\in[N]} D_{ii}.$$

Proof. For each r, consider $y_r \in B_r$. Sort the entries of y_r in descending order. Without loss of generality, assume that the ordering reads as $y_{r,i_1} \geq y_{r,i_2} \geq \cdots \geq y_{r,i_N}$. As $F_r([N]) = \sum_{j=1}^{N} y_{r,i_j} \geq 0$, we have $y_{r,i_1} \geq 0$. If $y_{r,i_N} \geq 0$, then $\|y_r\|_1 = \sum_{k=1}^{N} y_{r,i_k} = F_r([N]) \leq \max_{S\subseteq[N]} F_r(S)$. If $y_{r,i_N} < 0$, there exists a k' such that $y_{r,i_{k'}} \geq 0$ and $y_{r,i_{k'+1}} < 0$. Given the definition of B_r, we have

$$\sum_{k=k'+1}^{N} |y_{r,i_k}| \leq \sum_{k=1}^{k'} |y_{r,i_k}| \leq F_r(\{i_1, i_2, ..., i_{k'}\}) \leq \max_{S\subseteq[N]} F_r(S),$$

and thus $\|y_r\|_1 \leq 2\max_{S\subseteq[N]} F_r(S)$. Moreover, as each variable in $[N]$ is incident to at least one submodular function, we have

$$\sum_{r\in[R]} \max_{S\subseteq[N]}[F_r(S)]^2 \leq \sum_{i\in[N]} \sum_{r:i\in S_r} \max_{S\subseteq[N]}[F_r(S)]^2 \leq \sum_{i\in[N]} D_{ii}.$$

Combining all of the above results, we obtain

$$\rho^2 = \sum_{r\in[R]} \max_{y_r\in B_r} \|y_r\|_1^2 \leq 4\sum_{r\in[R]} \max_{S\subseteq[N]}[F_r(S)]^2 \leq 4\sum_{i\in[N]} D_{ii}.$$

□

When $W = \beta D$, we have

$$\sum_{i\in[N]} W_{ii} \sum_{j\in[N]} 1/W_{jj} \leq N^2 \max_{i,j} \frac{W_{ii}}{W_{jj}} = N^2 \max_{i,j} \frac{D_{ii}}{D_{jj}},$$

and

$$\rho^2 \sum_{j\in[N]} 1/W_{jj} \overset{1)}{\leq} 4\sum_{i\in[N]} D_{ii} \sum_{j\in[N]} 1/W_{jj} \leq \frac{4}{\beta} N^2 \max_{i,j} \frac{D_{ii}}{D_{jj}},$$

where 1) follows from Lemma A.5.6. According to the definition of $\mu(W^{-1}, W^{-1})$

(see (6.5)),

$$\mu(W^{-1}, W^{-1}) \leq N^2 \max\{1, 9\beta^{-1}\} \max_{i,j} \frac{D_{ii}}{D_{jj}}.$$

Similarly, we have

$$\mu(\Psi W^{-1}, W^{-1}) \leq N^2 \max\{1, 9\beta^{-1}\} \max_{i,j} \frac{\Psi_{jj} D_{ii}}{D_{jj}}.$$

This concludes the proof.

A.5.5 Convergence analysis of the conic MNP algorithm

Preliminary notation and lemmas

Given an active set $S = \{q_1, q_2, ...\}$, and a collection of coefficients $\lambda = \{\lambda_1, \lambda_2, ...\}$, if $y = \sum_{q_i \in S} \lambda_i q_i$, we simply refer to (y, S, λ) as a *triple*.

Define the following functions that depend on S

$$\tilde{h}(S, \lambda) \triangleq h(\sum_{q_i \in S} \lambda_i q_i, \sum_{q_i \in S} \lambda_i),$$
$$\tilde{h}(S) \triangleq \min_{\lambda : \lambda_i \in \mathbb{R}, \forall i} \tilde{h}(S, \lambda),$$
$$\tilde{h}_+(S) \triangleq \min_{\lambda : \lambda_i \geq 0, \forall i} \tilde{h}(S, \lambda).$$

If the coefficients λ ($\lambda_i \in \mathbb{R}, \forall i$) minimize $\tilde{h}(S, \lambda)$, we call the corresponding triple (y, S, λ) a *good* triple. Given a triple (y, S, λ), we also define

$$\triangle(y, q) = -\langle y - a, q \rangle - \sum_{q_i \in S} \lambda_i,$$

$$\triangle(y) = \max_{q \in B} \triangle(y) = -\min_{q \in B} \langle y - a, q \rangle - \sum_{q_i \in S} \lambda_i,$$

and

$$\text{err}(y) = h(y, \sum_{q_i \in S} \lambda_i) - h^*.$$

The following lemma establishes the optimality of a good triple.

Lemma A.5.7. *Given an active set S, consider the good triple (y', S, λ') and an arbitrary triple (y, S, λ). Then,*

$$\langle y' - a, y\rangle_{\tilde{W}} + \langle \sum_{q_i \in S} \lambda_i', \sum_{q_i \in S} \lambda_i \rangle = \langle y' - a, y' - y\rangle_{\tilde{W}} + \langle \sum_{q_i \in S} \lambda_i', \sum_{q_i \in S} (\lambda_i' - \lambda_i) \rangle = 0.$$

Proof. Without loss of generality, assume that

$$\langle y' - a, y\rangle_{\tilde{W}} + \langle \sum_{q_i \in S} \lambda_i', \sum_{q_i \in S} \lambda_i \rangle < 0.$$

Then, for any $\epsilon > 0$, $(y' + \epsilon y, S, \lambda' + \epsilon\lambda)$ is also a triple. For ϵ sufficiently small, we have $\tilde{h}(S, \lambda' + \epsilon\lambda) < \tilde{h}(S, \lambda')$, which contradicts the optimality of (y', S, λ'). Hence,

$$\langle y' - a, y\rangle_{\tilde{W}} + \langle \sum_{q_i \in S} \lambda_i', \sum_{q_i \in S} \lambda_i \rangle = 0.$$

As $(y' - y, S, \lambda' - \lambda)$ is also a triple, repeating the above procedure we obtain the claimed equality. □

Lemma A.5.8. *For any $\hat{y} \in B$,*

$$\arg\min_{\phi \geq 0} h(\phi\hat{y}, \phi) \leq \frac{\|a\|_{\tilde{W}}}{2}.$$

Moreover, $\phi^* \leq \frac{\|a\|_{\tilde{W}}}{2}$.

Proof. Given $\hat{y}$, the optimal value of ϕ satisfies

$$\phi = \frac{\langle a, \hat{y}\rangle_{\tilde{W}}}{1 + \|\hat{y}\|_{\tilde{W}}^2} \leq \frac{\|a\|_{\tilde{W}}}{\|\hat{y}\|_{\tilde{W}} + \frac{1}{\|\hat{y}\|_{\tilde{W}}}} \leq \frac{\|a\|_{\tilde{W}}}{2}.$$

This establishes the claimed result. □

Lemma A.5.9. *If (y, S, λ) is a good triple, then $\triangle(y) \geq \frac{err(y)}{\|a\|_{\tilde{W}}}$, where we recall that $err(y) = h(y, \sum_{q_i \in S} \lambda_i) - h^*$.*

Proof. Recall that (y^*, ϕ^*) denotes the optimal solution. As $y^*/\phi^* \in B$, we

have

$$\begin{aligned}
\phi^* \triangle(y) &\geq -\langle y-a, y^*\rangle_{\tilde{W}} - \langle \phi^*, \sum_{q_i \in S} \lambda_i \rangle \\
&\overset{1)}{=} -\langle y-a, y^*\rangle_{\tilde{W}} - \langle \phi^*, \sum_{q_i \in S} \lambda_i \rangle + \langle y-a, y\rangle_{\tilde{W}} + (\sum_{q_i \in S} \lambda_i)^2 \\
&= -\langle y-a, y^*-a\rangle_{\tilde{W}} - \langle \phi^*, \sum_{q_i \in S} \lambda_i \rangle + \langle y-a, y-a\rangle_{\tilde{W}} + (\sum_{q_i \in S} \lambda_i)^2 \\
&\overset{2)}{\geq} \frac{1}{2}\left[\|y-a\|^2_{\tilde{W}} + (\sum_{q_i \in S} \lambda_i)^2 - \|y^*-a\|^2_{\tilde{W}} + (\phi^*)^2 \right] \\
&= \frac{1}{2}\mathrm{err}(y),
\end{aligned}$$

where 1) follows from Lemma A.5.7, while 2) is a consequence of the Cauchy-Schwarz inequality. By using the bound for ϕ^* described in Lemma A.5.8, we arrive at the desired conclusion. □

Proof of Theorem 6.3.1

We only need to prove the following three lemmas which immediately give rise to Theorem 6.3.1. Lemma A.5.10 corresponds to the first statement of Theorem 6.3.1. Combining Lemma A.5.11 and Lemma A.5.12, we can establish the second statement of Theorem 6.3.1. This follows as we may choose $\epsilon = \delta\|a\|_{\tilde{W}}$.

If Algorithm 6.3 terminates after less than $O(N\|a\|^2_{\tilde{W}} \max\{Q^2, 1\}/\epsilon)$ iterations, then the condition of Lemma A.5.11 is satisfied and thus $\mathrm{err}(y^{(k)}) \leq \epsilon = \delta\|a\|_{\tilde{W}}$. If Algorithm 6.3 does not terminate after $O(N\|a\|^2_{\tilde{W}} \max\{Q^2, 1\}/\epsilon)$ iterations, Lemma A.5.12 guarantees $\mathrm{err}(y^{(k)}) \leq \epsilon = \delta\|a\|_{\tilde{W}}$.

Lemma A.5.10. *At any point before Algorithm 6.3 terminates, one has*

$$h(y^{(k)}, \phi^{(k)}) \geq h(y^{(k+1)}, \phi^{(k+1)});$$

moreover, if $(y^{(k)}, \phi^{(k)})$ triggers a MAJOR loop, the claimed inequality is strict.

The following lemma characterizes the pair (y, ϕ) at the point when the MNP method terminates.

Lemma A.5.11. *In the MAJOR loop at iteration k, if $\langle y^{(k)} - a, q^{(k)} \rangle_{\tilde{W}} + \phi^{(k)} \geq -\delta$, then $h(y^{(k)}, \phi^{(k)}) \leq h^* + \|a\|_{\tilde{W}} \delta$.*

Lemma A.5.12. *If Algorithm 6.3 does not terminate, then for any $\epsilon > 0$, one can guarantee that after $O(N\|a\|_{\tilde{W}}^2 \max\{Q^2, 1\}/\epsilon)$ iterations, Algorithm 6.3 generates a pair (y, ϕ) that satisfies err(y) $\leq \epsilon$.*

The proofs of Lemma A.5.10 and Lemma A.5.11 are fairly straightforward, while the proof of Lemma A.5.12 is significantly more involved and postponed to the next section.

Proof of Lemma A.5.10. Suppose that $(y^{(k)}, \phi^{(k)})$ starts a MAJOR loop. As

$$\langle y^{(k)} - a, q^{(k)} \rangle_{\tilde{W}} + \phi^{(k)} < -\delta,$$

we know that there exists some small ε such that

$$h(y^{(k)} + \varepsilon q^{(k)}, \phi^{(k)} + \varepsilon) < h(y^{(k)}, \phi^{(k)}).$$

Consider next the relationship between $(z^{(k)}, \sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} \alpha_i)$ and $(y^{(k)}, \phi^{(k)})$. Because of Step 6, we know that

$$h(z^{(k+1)}, \sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} \alpha_i) = \tilde{h}(S^{(k)} \cup \{q^{(k)}\}) \leq h(y^{(k)} + \varepsilon q^{(k)}, \phi^{(k)} + \varepsilon) < h(y^{(k)}, \phi^{(k)}).$$

If $(y^{(k+1)}, \phi^{(k+1)})$ is generated in some MAJOR loop, then

$$(y^{(k+1)}, \phi^{(k+1)}) = (z^{(k+1)}, \sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} \alpha_i),$$

which naturally satisfies the claimed condition. If $(y^{(k+1)}, \phi^{(k+1)})$ is generated in some MINOR loop, then $(y^{(k+1)}, \phi^{(k+1)})$ lies strictly within the segment between $(z^{(k+1)}, \sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} \alpha_i)$ and $(y^{(k)}, \phi^{(k)})$ (because $\theta > 0$). Therefore, we also have $h(y^{(k+1)}, \phi^{(k+1)}) < h(y^{(k)}, \phi^{(k)})$. If $(y^{(k)}, \phi^{(k)})$ starts a MINOR loop, then we have

$$h(z^{(k+1)}, \sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} \alpha_i) = \tilde{h}(S^{(k)}) \leq h(y^{(k)}, \phi^{(k)}),$$

once again due to Step 6. As $(y^{(k+1)}, \phi^{(k+1)})$ still lies within the segment between

$(z^{(k+1)}, \sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} \alpha_i)$ and $(y^{(k)}, \phi^{(k)})$, we have $h(y^{(k+1)}, \phi^{(k+1)}) \leq h(y^{(k)}, \phi^{(k)})$.

Proof of Lemma A.5.11. Lemma A.5.11 is a corollary of Lemma A.5.9. To see why this is the case, observe that in a MAJOR loop, $(y^{(k)}, S^{(k)}, \lambda^{(k)})$ is always a good triple. Since

$$\triangle(y^{(k)}) = -(\langle y^{(k)} - a, q^{(k)} \rangle_{\tilde{W}} + \phi^{(k)}) \leq \delta,$$

we have $\text{err}(y) \leq \delta \|a\|_{\tilde{W}}$.

Proof of Lemma A.5.12

The outline of the proof is similar to that of the standard case described in [85], and some results therein can be directly reused. The key step is to show that in every MAJOR loop k with no more than one MINOR loop, the objective achieved by $y^{(k)}$ decreases sufficiently, as precisely described in Theorem A.5.13.

Theorem A.5.13. *For a MAJOR loop with no more than one MINOR loop, if the starting point is y, the starting point y' of the next MAJOR loop satisfies*

$$err(y') \leq err(y) \left(1 - \frac{err(y)}{\|a\|_{\tilde{W}}(Q^2+1)}\right).$$

Based on this theorem, it is easy to establish the result in Lemma A.5.12 by using the next lemma and the same approach as described in [85].

Lemma A.5.14 (Lemma 1 [85])**.** *In any consecutive $3N+1$ iteratons, there exists at least one MAJOR loop with not more than one MINOR loop.*

We now focus on the proof of Theorem A.5.13. The next geometric lemma is the counterpart of Lemma 2 [85] for the conic case.

Lemma A.5.15. *Given an active set S, consider a good triple (y', S, λ') and an arbitrary triple (y, S, λ). Then, for any $q \in lin(S)$ such that $\triangle(y, q) > 0$, we have*

$$\|y - a\|_{\tilde{W}}^2 + (\sum_{q_i \in S} \lambda_i)^2 - \|y' - a\|_{\tilde{W}}^2 - (\sum_{q_i \in S} \lambda_i')^2 \geq \frac{\triangle^2(y, q)}{\|q\|_{\tilde{W}}^2 + 1}.$$

Proof. First, we have

$$\begin{aligned}
&\|y-a\|_{\tilde{W}}^2+(\sum_{q_i\in S}\lambda_i)^2-\|y'-a\|_{\tilde{W}}^2-(\sum_{q_i\in S}\lambda_i')^2\\
=&\|y-y'\|_{\tilde{W}}^2+[\sum_{q_i\in S}(\lambda_i-\lambda_i')]^2+2\langle y-y',y'-a\rangle_{\tilde{W}}+2\langle\sum_{q_i\in S}(\lambda_i-\lambda_i'),\sum_{q_i\in S}\lambda_i'\rangle\\
\overset{1)}{=}&\|y-y'\|_{\tilde{W}}^2+[\sum_{q_i\in S}(\lambda_i-\lambda_i')]^2,
\end{aligned}$$

where 1) follows from Lemma A.5.7. Next, for any $\phi\geq 0$,

$$\begin{aligned}
&\|y-y'\|_{\tilde{W}}^2+[\sum_{q_i\in S}(\lambda_i-\lambda_i')]^2\\
\overset{1)}{\geq}&\frac{\left[\langle y-y',y-\phi q\rangle_{\tilde{W}}+\langle\sum_{q_i\in S}(\lambda_i-\lambda_i'),\sum_{q_i\in S}\lambda_i-\phi\rangle\right]^2}{\|y-\phi q\|_{\tilde{W}}^2+(\sum_{q_i\in S}\lambda_i-\phi)^2}\\
\overset{2)}{=}&\frac{\left[\langle y-a,y-\phi q\rangle_{\tilde{W}}+\langle\sum_{q_i\in S}\lambda_i,\sum_{q_i\in S}\lambda_i-\phi\rangle\right]^2}{\|y-\phi q\|_{\tilde{W}}^2+(\sum_{q_i\in S}\lambda_i-\phi)^2},
\end{aligned}\tag{A.38}$$

where 1) follows from the Cauchy-Schwarz inequality and 2) is due to Lemma A.5.7. Since $\triangle(y,q)>0$, letting $\phi\to\infty$ reduces equation (A.38) to $\frac{\triangle^2(y,q)}{\|q\|_{\tilde{W}}^2+1}$. □

Next, using Lemma A.5.15, we may characterize the decrease of the objective function for one MAJOR loop with no MINOR loop. As $(y^{(k+1)},S^{(k+1)},\lambda^{(k+1)})$ is a good triple and $y^{(k)}$ also lies in $\mathrm{lin}(S)$, we have the following result.

Lemma A.5.16. *Consider some MAJOR loop k without MINOR loops. Then*

$$err(y^{(k)})-err(y^{(k+1)})\geq\frac{\triangle^2(y^{(k)},q^{(k)})}{Q^2+1}=\frac{\triangle^2(y^{(k)})}{Q^2+1}.$$

Next, we characterize the decrease of the objective function for one MAJOR loop with one single MINOR loop.

Lemma A.5.17. *Consider some MAJOR loop k with only one MINOR loop. Then*

$$err(y^{(k)})-err(y^{(k+2)})\geq\frac{\triangle^2(y^{(k)})}{Q^2+1}.$$

Proof. Suppose that the active sets associated with $y^{(k)}, y^{(k+1)}, y^{(k+2)}$ are $S^{(k)}, S^{(k+1)}, S^{(k+2)}$, respectively. We know that within the MINOR loop, $(z^{(k)}, S^{(k)} \cup \{q^{(k)}\}, \alpha)$ is a good triple and $y^{(k+1)} = \theta y^{(k)} + (1-\theta)z^{(k)}$, for some $\theta \in [0,1]$. Let

$$A = \|y^{(k)} - a\|^2_{\tilde{W}} + (\sum_{q_i \in S^{(k)}} \lambda_i^{(k)})^2 - \|z^{(k)} - a\|^2_{\tilde{W}} - (\sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} \alpha_i)^2. \quad \text{(A.39)}$$

From Lemma A.5.15, we have

$$A \geq \frac{\triangle^2(y^{(k)}, q^{(k)})}{\|q^{(k)}\|^2_{\tilde{W}} + 1} \geq \frac{\triangle^2(y^{(k)})}{Q^2 + 1}. \quad \text{(A.40)}$$

Note that both $S^{(k)}$ and $S^{(k+1)}$ are subsets of $S^{(k)} \cup \{q^{(k)}\}$. As $(z^{(k)}, S^{(k)} \cup \{q^{(k)}\}, \alpha)$ is a good triple, using Lemma A.5.7, we obtain

$$\langle z^{(k)} - a, z^{(k)} - y^{(k)} \rangle_{\tilde{W}} + \langle \sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} \alpha_i, \sum_{q_i \in S^{(k)} \cup \{q^{(k)}\}} (\alpha_i - \lambda_i^{(k)}) \rangle = 0.$$

Furthermore, as $y^{(k+1)} = \theta y^{(k)} + (1-\theta)z^{(k)} = z^{(k)} - \theta(z^{(k)} - y^{(k)})$ and $\lambda^{(k+1)} = \alpha - \theta(\alpha - \lambda^{(k)})$, we have

$$\|y^{(k)} - a\|^2_{\tilde{W}} + (\sum_{i \in S^{(k)}} \lambda_i^{(k)})^2 - \|y^{(k+1)} - a\|^2_{\tilde{W}} - (\sum_{i \in S^{(k+1)}} \lambda_i^{(k+1)})^2 = (1-\theta^2)A.$$

Moreover, we have

$$\triangle(y^{(k+1)}, q^{(k)}) = \theta\triangle(y^{(k)}, q^{(k)}) + (1-\theta)\triangle(z^{(k)}, q^{(k)}) = \theta\triangle(y^{(k)}, q^{(k)}) = \theta\triangle(y^{(k)}), \quad \text{(A.41)}$$

which holds because $\triangle(y, q)$ is linear in y and Lemma A.5.7 implies $\triangle(z^{(k)}, q^{(k)}) = 0$.

Since according to Lemma A.5.10, $h(y^{(k)}, \phi^{(k)}) > h(y^{(k+1)}, \phi^{(k+1)})$, Lemma 4 in [85] also holds in our case, and thus $q^{(k)} \in S^{(k+1)}$. To obtain $y^{(k+2)}$ and $S^{(k+2)}$, one needs to remove active points with a zero coefficients from $S^{(k+1)}$, so that $y^{(k+2)}$ once again belongs to a good triple with corresponding $S^{(k+1)}$.

Based on A.5.15 and equation (A.41), we have the following result.

$$\|y^{(k+1)}-a\|_{\tilde{W}}^2+(\sum_{q_i\in S^{(k+1)}}\lambda_i^{(k+1)})^2-\|y^{(k+2)}-a\|_{\tilde{W}}^2-(\sum_{q_i\in S^{(k+2)}}\lambda_i^{(k+2)})^2 \tag{A.42}$$

$$\geq\frac{\triangle^2(y^{(k+1)},q^{(k)})}{Q^2+1}=\frac{\theta^2\triangle^2(y^{(k)})}{Q^2+1}. \tag{A.43}$$

Consequently, combining equations (A.39), (A.40) and (A.42), we arrive at

$$\mathrm{err}(y^{(k)})-\mathrm{err}(y^{(k+2)})\geq\frac{\triangle^2(y^{(k)})}{Q^2+1}.$$

□

And, combining Lemma A.5.16, Lemma A.5.17 and Lemma A.5.9 establishes Theorem A.5.13.

A.5.6 Convergence analysis of the conic Frank-Wolfe algorithm

Proof of Theorem 6.3.2

Using the same strategy as in the proof of Lemma A.5.8, we may prove the following lemma. It hinges on the optimality assumption for $\gamma_1^{(k)}\phi^{(k)}+\gamma_2^{(k)}$ in Step 3 of Algorithm 6.4.

Lemma A.5.18. *In Algorithm 6.4, for all k,* $\phi^{(k+1)}\leq\frac{\|a\|_{\tilde{W}}}{2}$.

Now, we prove Theorem 6.3.2. We write $y=\phi\hat{y}$, where $\hat{y}\in B$, so that

$$h(y,\phi)=h(\phi\hat{y},\phi)\geq h(y^{(k)},\phi^{(k)})+2\langle y^{(k)}-a,\phi\hat{y}-y^{(k)}\rangle_{\tilde{W}}+\phi^2-(\phi^{(k)})^2.$$

For both sides, minimize $(\hat{y},\phi)$ over $B\times[0,\frac{\|a\|_{\tilde{W}}}{2}]$, which contains (y^*,ϕ^*). Since $q^{(k)}=\arg\min_{q\in B}\langle y^{(k)}-a,q\rangle_{\tilde{W}}$, we know that the optimal solutions satisfy $\hat{y}=q^{(k)}$ and $\phi=\tilde{\phi}^{(k)}=\min\{\max\{0,-\langle y^{(k)}-a,q^{(k)}\rangle\},\frac{\|a\|_{\tilde{W}}}{2}\}$ of the RHS

$$h^*=h(y^*,\phi^*)\geq h(y^{(k)},\phi^{(k)})+2\langle y^{(k)}-a,\tilde{\phi}^{(k)}q^{(k)}-y^{(k)}\rangle_{\tilde{W}}+(\tilde{\phi}^{(k)})^2-(\phi^{(k)})^2. \tag{A.44}$$

Moreover, because of the optimality of $(\gamma_1^{(k)} \times \gamma_2^{(k)}) \in \mathbb{R}^2_{\geq 0}$, for arbitrary $\gamma \in [0,1]$ we have

$$\begin{aligned}
&h(\gamma_1^{(k)} y^{(k)} + \gamma_2^{(k)} q^{(k)}, \gamma_1^{(k)} \phi^{(k)} + \gamma_2^{(k)}) \\
&\leq h((1-\gamma) y^{(k)} + \gamma \tilde{\phi}^{(k)} q^{(k)}, (1-\gamma)\phi^{(k)} + \gamma \tilde{\phi}^{(k)}) \\
&= h(y^{(k)}, \phi^{(k)}) + 2\gamma \langle y^{(k)} - a, \tilde{\phi}^{(k)} q^{(k)} - y^{(k)} \rangle_{\tilde{W}} \\
&+ \gamma[(\tilde{\phi}^{(k)})^2 - (\phi^{(k)})^2] + \gamma^2 \|\tilde{\phi}^{(k)} q^{(k)} - y^{(k)}\|^2_{\tilde{W}} + (\gamma^2 - \gamma)(\tilde{\phi}^{(k)} - \phi^{(k)})^2 \\
&\overset{1)}{\leq} h(y^{(k)}, \phi^{(k)}) + \gamma(h^* - h(y^{(k)}, \phi^{(k)})) + \gamma^2 \|\tilde{\phi}^{(k)} q^{(k)} - y^{(k)}\|^2_{\tilde{W}} \\
&\overset{2)}{\leq} h(y^{(k)}, \phi^{(k)}) + \gamma(h^* - h(y^{(k)}, \phi^{(k)})) + \gamma^2 \|a\|^2_{\tilde{W}} Q^2,
\end{aligned}$$

where 1) follows from (A.44) and $\gamma^2 - \gamma \leq 0$, and 2) follows from

$$\|\tilde{\phi}^{(k)} q^{(k)} - y^{(k)}\|^2_{\tilde{W}} \leq 4 \frac{\|a\|^2_{\tilde{W}}}{4} \max_{q \in B} \|q\|^2_{\tilde{W}} = \|a\|^2_{\tilde{W}} Q^2.$$

The claimed result now follows by induction. First, let $\hat{y}^* = y^*/\phi^*$, where $\phi^* = \frac{\langle \hat{y}^*, a \rangle_{\tilde{W}}}{1 + \|\hat{y}^*\|^2_{\tilde{W}}}$. Then,

$$h(y^{(0)}, \phi^{(0)}) - h^* \leq 2\langle y^*, a \rangle - (y^*)^2 - (\phi^*)^2 = \frac{\langle \hat{y}^*, a \rangle^2_{\tilde{W}}}{1 + \|\hat{y}^*\|^2_{\tilde{W}}} \leq \|a\|^2_{\tilde{W}} Q^2.$$

Suppose that $h(y^{(k)}, r^{(k)}) - h^* \leq \frac{2\|a\|^2_{\tilde{W}} Q^2}{k+2}$. In this case, for all $\gamma \in [0,1]$, we have

$$h(y^{(t+1)}, \phi^{(t+1)}) - h^* \leq (1-\gamma)[h(y^{(k)}, \phi^{(k)}) - h^*] + \gamma^2 \|a\|^2_{\tilde{W}} Q^2.$$

By choosing $\gamma = \frac{1}{k+2}$, we obtain $h(y^{(k+1)}, \phi^{(k+1)}) - h^* \leq \frac{2\|a\|^2_{\tilde{W}} Q^2}{k+3}$, which concludes the proof.

A.5.7 Proofs for the partitioning properties of PageRank

Proof of Lemma 6.4.3

Based on the definitions of $\mathcal{S}_j^p$ and $j \in V_p$, we have $\langle \nabla f_r(x), 1_{\mathcal{S}_j^p} \rangle = F_r(\mathcal{S}_j^p)$. Consequently,

$$
\begin{aligned}
& 2p(\mathcal{S}_j^p) - \sum_{r\in[R]} w_r f_r(x) \langle \nabla f_r(x), 1_{\mathcal{S}_j^p} \rangle \\
=& 2p(\mathcal{S}_j^p) - \sum_{r\in[R]} w_r f_r(x) F_r(\mathcal{S}_j^p) \\
=& 2p(\mathcal{S}_j^p) - \sum_{r\in[R]} w_r F_r(S_j^p) \max_{(i,j)\in S_r^{\uparrow}\times S_r^{\downarrow}} (x_i - x_j) \\
=& \left(I_p(\mathrm{vol}(\mathcal{S}_j^p)) - \sum_{r\in[R]} w_r F_r(\mathcal{S}_j^p) \max_{i\in S_r^{\uparrow}} x_i \right) + \left(I_p(\mathrm{vol}(\mathcal{S}_j^p)) + \sum_{r\in[R]} w_r F_r(\mathcal{S}_j^p) \min_{i\in S_r^{\downarrow}} x_i \right) \\
\leq& I_p \left(\mathrm{vol}(\mathcal{S}_j^p) - \sum_{r\in[R]} w_r F_r(\mathcal{S}_j^p) \right) + I_p \left(\mathrm{vol}(\mathcal{S}_j^p) + \sum_{r\in[R]} w_r F_r(\mathcal{S}_j^p) \right) \\
=& I_p \left(\mathrm{vol}(\mathcal{S}_j^p) - \mathrm{vol}(\partial\mathcal{S}_j^p) \right) + I_p \left(\mathrm{vol}(\mathcal{S}_j^p) + \mathrm{vol}(\partial\mathcal{S}_j^p) \right). \qquad \text{(A.45)}
\end{aligned}
$$

Using equation (6.11), we have

$$
\begin{aligned}
p(\mathcal{S}_j^p) &= \frac{\alpha}{2-\alpha} p_0(\mathcal{S}_j^p) + \frac{2-2\alpha}{2-\alpha} \left\{ \frac{1}{2} \left[M(p) - p \right] (\mathcal{S}_j^p) + p(\mathcal{S}_j^p) \right\} \\
&\overset{1)}{=} \frac{\alpha}{2-\alpha} p_0(\mathcal{S}_j^p) + \frac{2-2\alpha}{2-\alpha} \left[-\frac{1}{2} \sum_{r\in[R]} f_r(x) \langle \nabla f_r(x), 1_{\mathcal{S}_j^p} \rangle + p(\mathcal{S}_j^p) \right] \\
&\overset{2)}{\leq} \frac{\alpha}{2-\alpha} p_0(\mathcal{S}_j^p) + \frac{1-\alpha}{2-\alpha} \left[I_p \left(\mathrm{vol}(\mathcal{S}_j^p) - \mathrm{vol}(\partial\mathcal{S}_j^p) \right) + I_p \left(\mathrm{vol}(\mathcal{S}_j^p) + \mathrm{vol}(\partial\mathcal{S}_j^p) \right) \right],
\end{aligned}
$$

where 1) is due to Lemma 6.4.1 and 2) is due to equation (A.45). This proves the first inequality. By using the concavity of $I_p(\cdot)$, we also have

$$
I_p(\mathrm{vol}(\mathcal{S}_j^p)) \leq p_0(\mathcal{S}_j^p) \leq I_{p_0}(\mathrm{vol}(\mathcal{S}_j^p)).
$$

Moreover, as I_p is piecewise linear, the proof follows.

Proof of Theorem 6.4.4

This result can be proved in a similar way as the corresponding case for graphs [106], by using induction.

Define $\bar{k} = \min\{k, m-k\}$, $d_{\min} = \min_{i:(p_0)_i>0} d_i$ and

$$I^{(t)}(k) = \frac{k}{m} + \frac{\alpha}{2-\alpha}t + \sqrt{\frac{\bar{k}}{d_{\min}}}\left(1 - \frac{\Phi_p^2}{8}\right)^t.$$

When $t = 0$, $I_p(k) \leq I_{p_0}(k)$ holds due to Lemma 6.4.3. As $I^{(t)}(m) = I^{(t)}(d_{\min}) = 1$, for $k \in [d_{\min}, m]$, we have

$$I_{p_0}(k) \leq 1 \leq I^{(0)}(k).$$

Since for $k \in [0, d_{\min}]$, $I_{p_0}(k)$ is linear, we also have $I_{p_0}(0) = 0 \leq I^{(t)}(0)$. Hence, for $k \in [0, d_{\min}]$, it also holds that $I_{p_0}(k) \leq I^{(0)}(k)$.

Next, suppose that for step t, $I_p(k) \leq I^{(t)}(k)$ holds. We then consider the case $t+1$. For $k = \mathrm{vol}(\mathcal{S}_j^p)$, Lemma 6.4.3 indicates that

$$\begin{aligned}
I_p(k) &\leq \frac{\alpha}{2-\alpha}I_{p_0}(k) + \frac{1-\alpha}{2-\alpha}[I_p(k - \bar{k}\Phi(\mathcal{S}_j^p)) + I_p(k + \bar{k}\Phi(\mathcal{S}_j^p))] \\
&\overset{1)}{\leq} \frac{\alpha}{2-\alpha}I_{p_0}(k) \\
&\quad + \frac{1-\alpha}{2-\alpha}\left[\frac{2k}{m} + \frac{2\alpha}{2-\alpha}t + \left(\sqrt{\frac{k - \bar{k}\Phi(\mathcal{S}_j^p)}{d_{\min}}} + \sqrt{\frac{k + \bar{k}\Phi(\mathcal{S}_j^p)}{d_{\min}}}\right)\left(1 - \frac{\Phi_p^2}{8}\right)^t\right] \\
&\overset{2)}{\leq} \frac{k}{m} + \frac{\alpha}{2-\alpha}(t+1) + \frac{1-\alpha}{2-\alpha}\left(\sqrt{\frac{k - \bar{k}\Phi(\mathcal{S}_j^p)}{d_{\min}}} + \sqrt{\frac{k + \bar{k}\Phi(\mathcal{S}_j^p)}{d_{\min}}}\right)\left(1 - \frac{\Phi_p^2}{8}\right)^t \\
&\overset{3)}{\leq} \frac{k}{m} + \frac{\alpha}{2-\alpha}(t+1) + \frac{1-\alpha}{2-\alpha}\left(\sqrt{\frac{\bar{k} - \bar{k}\Phi(\mathcal{S}_j^p)}{d_{\min}}} + \sqrt{\frac{\bar{k} + \bar{k}\Phi(\mathcal{S}_j^p)}{d_{\min}}}\right)\left(1 - \frac{\Phi_p^2}{8}\right)^t \\
&\overset{4)}{\leq} \frac{k}{m} + \frac{\alpha}{2-\alpha}(t+1) + \frac{2-2\alpha}{2-\alpha}\sqrt{\frac{\bar{k}}{d_{\min}}}\left(1 - \frac{\Phi_p^2}{8}\right)^{t+1} \leq I^{(t+1)}(k),
\end{aligned}$$

where 1) follows from Lemma 6.4.3; 2) is due to $I_{p_0}(k) \leq 1$; 3) can be verified by considering two cases separately, namely $k \leq \frac{m}{2}$ and $k \geq \frac{m}{2}$; and 4) follows from $\Phi(\mathcal{S}_j^p) \geq \Phi_p$ and the Taylor-series expansion of $\sqrt{x \pm \phi x}$.

At this point, we have shown that $I_p(k) \leq I^{(t)}(k)$ for all $k = \mathrm{vol}(\mathcal{S}_j^p)$. Since

$\{\text{vol}(\mathcal{S}_j^p)\}_{j\in V_p}$ covers all break points of I_p and since $I^{(t)}$ is concave, we have that for all $k \in [0, m]$, it holds that $I_p(k) \leq I^{(t)}(k)$. This concludes the proof.

Proof of Theorem 6.4.5

First, we prove that under the assumptions for the vertex-sampling probability P in the statement of the theorem, $pr(\alpha, 1_i)(S)$ can be lower bounded as follows.

Lemma A.5.19. *If a vertex $v \in S$ is sampled according to a distribution P such that*

$$\mathbb{E}_{i\sim P}[pr(\alpha, 1_i)(\bar{S})] \leq c\, pr(\alpha, \pi_S)(\bar{S}),$$

where c is a constant, then with probability at least $\frac{1}{2}$, one has

$$pr(\alpha, 1_i)(S) \geq 1 - \frac{c\,\Phi(S)}{4\alpha}.$$

Proof. Let $p = pr(\alpha, \pi_S)$. Then,

$$\begin{aligned}
\alpha p(\bar{S}) &\overset{1)}{=} \alpha\pi_S(\bar{S}) + (1-\alpha)\left[M(p) - p\right](\bar{S}) \\
&\overset{2)}{=} -\sum_{r\in[R]} w_r f_r(x)\langle \nabla f_r(x), 1_{\bar{S}}\rangle \\
&\overset{3)}{\leq} \sum_{r\in[R]} w_r f_r(x) F(S) \leq \sum_{r\in[R]} w_r F(S) \max_{i\in S_r^{\uparrow}} x_i \\
&\leq I_p(\sum_{r\in[R]} w_r F(S)) = I_p(\text{vol}(\partial S)) \\
&\overset{4)}{\leq} I_{\pi_S}(\text{vol}(\partial S)) = \Phi(S),
\end{aligned}$$

where 1) is a consequence of Equation (6.11); 2) follows from Lemma 6.4.1; 3) holds because for any $x \in \mathbb{R}^N$, $\langle \nabla f_r(x), 1_{\bar{S}}\rangle \geq -F(S)$; and 4) is a consequence of Lemma 6.4.3. Hence,

$$\mathbb{E}_{v\sim P}[pr(\alpha, 1_i)(\bar{S})] \leq \frac{c}{8}\, pr(\alpha, \pi_S)(\bar{S}) \leq \frac{c}{8}\,\frac{\Phi(S)}{\alpha}.$$

Moreover, sampling according to $v \sim P$ and using Markov's inequality, we

have

$$\mathbb{P}\left[pr(\alpha,1_i)(\bar{S}) \geq \frac{c}{4}\frac{\Phi(S)}{\alpha}\right] \leq \frac{\mathbb{E}_{v\sim P}[pr(\alpha,1_i)(\bar{S})]}{\frac{c}{4}\frac{\Phi(S)}{\alpha}} \leq \frac{1}{2},$$

which concludes the proof. □

As $\Phi(S) \leq \frac{\alpha}{c}$, we have $\mathbb{P}\left[pr(\alpha,1_i)(S) \geq \frac{3}{4}\right] \geq \frac{1}{2}$. By combining this lower bound on $pr(\alpha,1_i)(S)$ with the upper bound of Theorem 6.4.4, we have

$$\frac{3}{4} \leq pr(\alpha,1_i)(S) \leq I_{pr(\alpha,1_i)}(\mathrm{vol}(S)) \leq \frac{1}{2} + \frac{\alpha}{2-\alpha}t + \sqrt{\frac{\mathrm{vol}(S)}{d_i}}\left(1 - \frac{\Phi^2_{pr(\alpha,1_i)}}{8}\right)^t.$$

Next, we choose $t = \lceil \frac{8}{\Phi^2_{pr(\alpha,1_i)}} \ln 8\sqrt{\frac{\mathrm{vol}(S)}{d_i}} \rceil$. Then, the above inequality may be transformed into

$$\frac{3}{4} \leq \frac{1}{2} + \frac{\alpha}{2-\alpha}\lceil \frac{8}{\Phi^2_{pr(\alpha,1_i)}} \ln 8\sqrt{\frac{\mathrm{vol}(S)}{d_i}} \rceil + \frac{1}{8} \leq \frac{5}{8} + \frac{\alpha}{2-\alpha}\frac{8}{\Phi^2_{pr(\alpha,1_i)}} \ln 10\sqrt{\frac{\mathrm{vol}(S)}{d_i}}.$$

Therefore,

$$\Phi_{pr(\alpha,1_i)} \leq \sqrt{32\alpha \ln \frac{100\mathrm{vol}(S)}{d_i}}.$$

A.5.8 Analysis of the parameter choices for experimental verification

Let $x' = W^{-1/2}x$ and $a' = W^{-1/2}a$. We can transform the objective (6.14) into a standard QDSFM problem,

$$\beta\|x' - a'\|^2_W + \sum_{r\in[R]} \max_{i,j\in S_r} (x'_i - x'_j)^2.$$

From Theorem 6.2.2, to achieve an ϵ-optimal solution, one requires

$$O(R\mu(\beta^{-1}W^{-1}, \beta^{-1}W^{-1}) \log \frac{1}{\epsilon})$$

iterations in the RCD algorithm (Algorithm 6.1). According to the particular settings for the experiment (undirected unweighted hypergraphs), we have

$$\rho^2 = \max_{y_r \in B_r, \forall r} \sum_{r \in [R]} \|y_r\|_1^2 = \max_{y_r \in B_r, \forall r} \sum_{r \in [R]} 2 = 2R. \tag{A.46}$$

From the definition of μ (6.5), we may rewrite $\mu(\beta^{-1}W^{-1}, \beta^{-1}W^{-1})$ as

$$\begin{aligned}
\mu(\beta^{-1}W^{-1}, \beta^{-1}W^{-1}) &\overset{1)}{=} \max\left\{ \frac{N^2}{2}\left(\max_{i,j\in[N]} \frac{W_{ii}}{W_{jj}} + 1 \right), \frac{9}{4}\rho^2 N\beta^{-1} \max_{j\in[N]} \frac{1}{W_{jj}} \right\} \\
&\overset{2)}{=} \max\left\{ \frac{N^2}{2}\left(\max_{i,j\in[N]} \frac{W_{ii}}{W_{jj}} + 1 \right), \frac{9}{2}\beta^{-1} NR \max_{j\in[N]} \frac{1}{W_{jj}} \right\} \\
&\overset{3)}{=} \max\left\{ \frac{N^2}{2}\left(\max_{i,j\in[N]} \frac{W_{ii}}{W_{jj}} + 1 \right), \frac{9}{2}\beta^{-1} N^2 \max_{i,j\in[N]} \frac{W_{ii}}{W_{jj}} \right\},
\end{aligned} \tag{A.47}$$

where 1) holds because half of the values of W_{ii} are set to 1 and the other half to a value in $\{1, 0.1, 0.01, 0.001\}$, 2) follows from (A.46) and 3) is due to the particular setting $N = R$ and $\max_{i\in[N]} W_{ii} = 1$. Equation (A.47) may consequently be rewritten as

$$O(N^2 \max(1, 9/(2\beta)) \max_{i,j\in[N]} W_{ii}/W_{jj}),$$

which establishes the claimed statement.

A.6 Additional Tables

Table A.1: Species in the Florida Bay foodweb with biological classification and assigned clusters. Cluster labels and colors correspond to the clusters shown in Figure 3.1. For InH-partition, in the first-level clustering, the species Roots is the only singleton while in the second-level clustering, the species Kingfisher, Hawksbill Turtle and Manatee are singletons.

Species	Biological Classification	Cluster (Ours)	Cluster (Benson's [39])
Roots	producers (no predator)	Singleton	Singleton
2μm Spherical cya	phytoplankton producers	Green	Singleton
Synedococcus	phytoplankton producers	Green	Singleton
Oscillatoria	phytoplankton producers	Green	Singleton
Small Diatoms ($< 20\mu$m)	phytoplankton producers	Green	Singleton
Big Diatoms ($> 20\mu$m)	phytoplankton producers	Green	Singleton
Dinoflagellates	phytoplankton producers	Green	Singleton
Other Phytoplankton	phytoplankton producers	Green	Singleton
Free Bacteria	producers	Green	Green
Water Flagellates	producers	Green	Green
Water Cilitaes	producers	Green	Green
Kingfisher	bird (no predator)	Singleton	Singleton
A. Hawksbill Turtle	reptiles (no predator)	Singleton	Singleton
Manatee	mammal (no predator)	Singleton	Singleton
Rays	fish	Blue	Singleton
Bonefish	fish	Blue	Singleton
Lizardfish	fish	Blue	Red
Catfish	fish	Blue	Blue
Eels	fish	Blue	Red
Brotalus	fish	Blue	Blue
Needlefish	fish	Blue	Yellow
Snook	fish	Blue	Singleton
Jacks	fish	Blue	Singleton
Pompano	fish	Blue	Singleton
Other Snapper	fish	Blue	Singleton
Gray Snapper	fish	Blue	Singleton
Grunt	fish	Blue	Singleton
Porgy	fish	Blue	Singleton
Scianids	fish	Blue	Singleton
Spotted Seatrout	fish	Blue	Singleton
Red Drum	fish	Blue	Singleton
Spadefish	fish	Blue	Singleton
Flatfish	fish	Blue	Blue
Filefish	fish	Blue	Singleton
Puffer	fish	Blue	Singleton
Other Pelagic fish	fish	Blue	Yellow
Small Herons & Egrets	bird	Blue	Singleton
Ibis	bird	Blue	Singleton
Roseate Spoonbill	bird	Blue	Singleton
Herbivorous Ducks	bird	Blue	Singleton
Omnivorous Ducks	bird	Blue	Singleton
Gruiformes	bird	Blue	Singleton
Small Shorebird	bird	Blue	Singleton
Gulls & Terns	bird	Blue	Singleton
Loggerhead Turtle	reptiles (no predator)	Blue	Singleton
Sharks	fish (no predator)	Purple	Singleton
Tarpon	fish	Purple	Singleton
Grouper	fish (no predator)	Purple	Singleton
Mackerel	fish (no predator)	Purple	Singleton
Barracuda	fish	Purple	Singleton
Loon	bird (no predator)	Purple	Singleton
Greeb	bird (no predator)	Purple	Singleton
Pelican	bird	Purple	Singleton
Comorant	bird	Purple	Singleton
Big Herons & Egrets	bird	Purple	Singleton
Predatory Ducks	bird (no predator)	Purple	Singleton
Raptors	bird (no predator)	Purple	Singleton
Crocodiles	reptiles (no predator)	Purple	Singleton
SingleDolphin	mammal (no predator)	Purple	Singleton

Table A.1: Continued.

Species	Biological Classification	Cluster Labels	Cluster(Benson's [39])
Benthic microalgea	algea producers	Yellow	Blue
Thalassia	seagrass producers	Yellow	Blue
Halodule	seagrass producers	Yellow	Blue
Syringodium	seagrass producers	Yellow	Blue
Drift Algae	algea producers	Yellow	Blue
Epiphytes	algea producers	Yellow	Blue
Acartia Tonsa	zooplankton invertebrates	Yellow	Green
Oithona nana	zooplankton invertebrates	Yellow	Green
Paracalanus	zooplankton invertebrates	Yellow	Green
Other Copepoda	zooplankton invertebrates	Yellow	Green
Meroplankton	zooplankton invertebrates	Yellow	Green
Other Zooplankton	zooplankton invertebrates	Yellow	Green
Benthic Flagellates	invertebrates	Yellow	Blue
Benthic Ciliates	invertebrates	Yellow	Blue
Meiofauna	invertebrates	Yellow	Blue
Sponges	macro-invertebrates	Yellow	Green
Bivalves	macro-invertebrates	Yellow	Blue
Detritivorous Gastropods	macro-invertebrates	Yellow	Blue
Epiphytic Gastropods	macro-invertebrates	Yellow	Singleton
Predatory Gastropods	macro-invertebrates	Yellow	Blue
Detritivorous Polychaetes	macro-invertebrates	Yellow	Blue
Predatory Polychaetes	macro-invertebrates	Yellow	Blue
Suspension Feeding Polych	macro-invertebrates	Yellow	Blue
Macrobenthos	macro-invertebrates	Yellow	Blue
Benthic Crustaceans	macro-invertebrates	Yellow	Blue
Detritivorous Amphipods	macro-invertebrates	Yellow	Blue
Herbivorous Amphipods	macro-invertebrates	Yellow	Blue
Isopods	macro-invertebrates	Yellow	Blue
Herbivorous Shrimp	macro-invertebrates	Yellow	Red
Predatory Shrimp	macro-invertebrates	Yellow	Blue
Pink Shrimp	macro-invertebrates	Yellow	Blue
Thor Floridanus	macro-invertebrates	Yellow	Singleton
Detritivorous Crabs	macro-invertebrates	Yellow	Red
Omnivorous Crabs	macro-invertebrates	Yellow	Blue
Green Turtle	reptiles	Yellow	Singleton
Coral	macro-invertebrates	Red	Singleton
Other Cnidaridae	macro-invertebrates	Red	Blue
Echinoderma	macro-invertebrates	Red	Blue
Lobster	macro-invertebrates	Red	Singleton
Predatory Crabs	macro-invertebrates	Red	Red
Callinectus sapidus	macro-invertebrates	Red	Red
Stone Crab	macro-invertebrates	Red	Singleton
Sardines	fish	Red	Yellow
Anchovy	fish	Red	Yellow
Bay Anchovy	fish	Red	Yellow
Toadfish	fish	Red	Blue
Halfbeaks	fish	Red	Yellow
Other Killifish	fish	Red	Singleton
Goldspotted killifish	fish	Red	Yellow
Rainwater killifish	fish	Red	Yellow
Sailfin Molly	fish	Red	Singleton
Silverside	fish	Red	Yellow
Other Horsefish	fish	Red	Singleton
Gulf Pipefish	fish	Red	Singleton
Dwarf Seahorse	fish	Red	Singleton
Mojarra	fish	Red	Singleton
Pinfish	fish	Red	Singleton
Parrotfish	fish	Red	Singleton
Mullet	fish	Red	Blue
Blennies	fish	Red	Blue
Code Goby	fish	Red	Red
Clown Goby	fish	Red	Red
Other Demersal Fish	fish	Red	Blue

Table A.2: Inhomogeous cost functions $w_e^{(r)}(S)$ for $|e| \in \{4, 5, 6\}$.

$|e| = 4$, $\beta^{(e)} = 3/2$

r	S						
	1	2	3	4	1, 2	1, 3	1, 4
1	0	1	1	1	1	1	1
2	0	0	1	1	0	1	1
3	1	1	1	1	1	1	1
4	1	1	1	1	2	2	2

$|e| = 5$, $\beta^{(e)} = 2$

r	S														
	1	2	3	4	5	1, 2	1, 3	1, 4	1, 5	2, 3	2, 4	2, 5	3, 4	3, 5	4, 5
1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	0	1	1	1	1	1	1	1	1	2	2	2	2	2	2
3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	1	1	1	1	1	1	2	2	2	2	2	2	1	1	1
5	1	1	1	1	1	0	2	2	2	2	2	2	1	1	1
6	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2

$|e| = 6$, $\beta^{(e)} = 4$

r	S														
	1	2	3	4	5	6	1, 2	1, 3	1, 4	1, 5	1, 6	2, 3	2, 4	2, 5	2, 6
1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	0	1	1	1	1	1	1	1	1	1	1	2	2	2	2
3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2
5	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2
6	1	1	1	1	1	1	0	2	2	2	2	2	2	2	2
7	1	1	1	1	1	1	1	1	2	2	2	1	2	2	2
8	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2
9	1	1	1	1	1	1	1	1	2	2	2	1	2	2	2

r	S											
	3, 4	3, 5	3, 6	4, 5	4, 6	5, 6	1, 2, 3	1, 2, 4	1, 2, 5	1, 2, 6	1, 3, 4	1, 3, 5
1	1	1	1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2	2	2	2	2
3	1	1	1	1	1	1	1	1	1	1	1	1
4	2	2	2	2	2	2	3	3	3	3	3	3
5	1	1	1	1	1	1	1	1	1	1	2	2
6	1	1	1	1	1	1	1	1	1	1	2	2
7	2	2	2	1	1	1	1	2	2	2	2	2
8	2	2	2	2	2	2	1	3	3	3	3	3
9	2	2	2	1	1	1	0	2	2	2	2	2

r	S			
	1, 3, 6	1, 4, 5	1, 4, 6	1, 5, 6
1	1	1	1	1
2	2	2	2	2
3	1	1	1	1
4	3	3	3	3
5	2	2	2	2
6	2	2	2	2
7	2	2	2	2
8	3	3	3	3
9	2	2	2	2

REFERENCES

[1] S. Wasserman and K. Faust, *Social network analysis: Methods and applications*. Cambridge University Press, 1994, vol. 8.

[2] D. Warde-Farley, S. L. Donaldson, O. Comes, K. Zuberi, R. Badrawi, P. Chao, M. Franz, C. Grouios, F. Kazi, C. T. Lopes et al., "The genemania prediction server: biological network integration for gene prioritization and predicting gene function," *Nucleic Acids Research*, vol. 38, no. 2, pp. W214–W220, 2010.

[3] E. M. Rogers and D. L. Kincaid, *Communication Networks: Toward a New Paradigm for Research.* New York Free Press, 1981.

[4] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 40–51, 2007.

[5] C. Berge, *Hypergraphs: Combinatorics of Finite Sets*. Elsevier, 1984, vol. 45.

[6] C. Chekuri and C. Xu, "Computing minimum cuts in hypergraphs," in *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*. Society for Industrial and Applied Mathematics, 2017, pp. 1085–1100.

[7] D. Zhou, J. Huang, and B. Schölkopf, "Learning with hypergraphs: Clustering, classification, and embedding," in *Advances in Neural Information Processing Systems*, 2007, pp. 1601–1608.

[8] A. Louis, "Hypergraph Markov operators, eigenvalues and approximation algorithms," in *Proceedings of the ACM Symposium on Theory of Computing*. ACM, 2015, pp. 713–722.

[9] M. Hein, S. Setzer, L. Jost, and S. S. Rangapuram, "The total variation on hypergraphs-learning on hypergraphs revisited," in *Advances in Neural Information Processing Systems*, 2013, pp. 2427–2435.

[10] H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A.-L. Barabási, "The large-scale organization of metabolic networks," *Nature*, vol. 407, no. 6804, pp. 651–654, 2000.

[11] S. Agarwal, J. Lim, L. Zelnik-Manor, P. Perona, D. Kriegman, and S. Belongie, "Beyond pairwise clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2. IEEE, 2005, pp. 838–845.

[12] F. R. Chung, "Four proofs for the cheeger inequality and graph partition algorithms," in *Proceedings of ICCM*, vol. 2, 2007, p. 378.

[13] X. Zhu, J. Lafferty, and Z. Ghahramani, "Combining active learning and semi-supervised learning using gaussian fields and harmonic functions," in *ICML 2003 Workshop on the Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining*, vol. 3, 2003.

[14] X. Zhu, Z. Ghahramani, and J. D. Lafferty, "Semi-supervised learning using gaussian fields and harmonic functions," in *Proceedings of the 20th International Conference on Machine learning*, 2003, pp. 912–919.

[15] C. Arora, S. Banerjee, P. Kalra, and S. Maheshwari, "Generic cuts: An efficient algorithm for optimal inference in higher order MRF-MAP," in *Proceedings of the European Conference on Computer Vision*. Springer, 2012, pp. 17–30.

[16] A. Fix, T. Joachims, S. M. Park, and R. Zabih, "Structured learning of sum-of-submodular higher order energy functions," in *Proceedings of the IEEE International Conference on Computer Vision*. IEEE, 2013, pp. 3104–3111.

[17] P. Kohli, P. H. Torr et al., "Robust higher order potentials for enforcing label consistency," *International Journal of Computer Vision*, vol. 82, no. 3, pp. 302–324, 2009.

[18] P. Stobbe and A. Krause, "Efficient minimization of decomposable submodular functions," in *Advances in Neural Information Processing Systems*, 2010, pp. 2208–2216.

[19] S. Jegelka, F. Bach, and S. Sra, "Reflection methods for user-friendly submodular optimization," in *Advances in Neural Information Processing Systems*, 2013, pp. 1313–1321.

[20] A. Ene and H. Nguyen, "Random coordinate descent methods for minimizing decomposable submodular functions," in *Proceedings of the International Conference on Machine Learning*, 2015, pp. 787–795.

[21] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank citation ranking: Bringing order to the web." Stanford InfoLab, Tech. Rep., 1999.

[22] D. Gleich and M. Mahoney, "Anti-differentiating approximation algorithms: A case study with min-cuts, spectral, and flow," in *International Conference on Machine Learning*, 2014, pp. 1018–1025.

[23] T.-H. H. Chan, A. Louis, Z. G. Tang, and C. Zhang, "Spectral properties of hypergraph laplacian and approximation algorithms," *Journal of the ACM (JACM)*, vol. 65, no. 3, p. 15, 2018.

[24] M. Frank and P. Wolfe, "An algorithm for quadratic programming," *Naval Research Logistics*, vol. 3, no. 1-2, pp. 95–110, 1956.

[25] P. Wolfe, "Finding the nearest point in a polytope," *Mathematical Programming*, vol. 11, no. 1, pp. 128–149, 1976.

[26] V. Kaibel, "On the expansion of graphs of 0/1-polytopes," in *The Sharpest Cut: The Impact of Manfred Padberg and His Work*. SIAM, 2004, pp. 199–216.

[27] L. Lovász, "Submodular functions and convexity," in *Mathematical Programming The State of the Art*. Springer, 1983, pp. 235–257.

[28] F. Bach et al., "Learning with submodular functions: A convex optimization perspective," *Foundations and Trends in Machine Learning*, vol. 6, no. 2-3, pp. 145–373, 2013.

[29] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review," *ACM Computing Surveys (CSUR)*, vol. 31, no. 3, pp. 264–323, 1999.

[30] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems (NIPS)*, 2002, pp. 849–856.

[31] S. R. Bulò and M. Pelillo, "A game-theoretic approach to hypergraph clustering," in *Advances in Neural Information Processing Systems (NIPS)*, 2009, pp. 1571–1579.

[32] M. Leordeanu and C. Sminchisescu, "Efficient hypergraph clustering," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2012, pp. 676–684.

[33] H. Liu, L. J. Latecki, and S. Yan, "Robust clustering as ensembles of affinity relations," in *Advances in Neural Information Processing Systems (NIPS)*, 2010, pp. 1414–1422.

[34] N. Bansal, A. Blum, and S. Chawla, "Correlation clustering," in *The 43rd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2002, pp. 238–247.

[35] N. Ailon, M. Charikar, and A. Newman, "Aggregating inconsistent information: ranking and clustering," *Journal of the ACM (JACM)*, vol. 55, no. 5, p. 23, 2008.

[36] P. Li, H. Dau, G. Puleo, and O. Milenkovic, "Motif clustering and overlapping clustering for social network analysis," in *IEEE Conference on Computer Communications (INFOCOM)*, 2017, pp. 109–117.

[37] J. Kunegis, S. Schmidt, A. Lommatzsch, J. Lerner, E. W. De Luca, and S. Albayrak, "Spectral analysis of signed graphs for clustering, prediction and visualization," in *SIAM International Conference on Data Mining (ICDM)*, 2010, pp. 559–570.

[38] A. V. Knyazev, "Signed Laplacian for spectral clustering revisited," 2017, arXiv preprint arXiv:1701.01394.

[39] A. R. Benson, D. F. Gleich, and J. Leskovec, "Higher-order organization of complex networks," *Science*, vol. 353, no. 6295, pp. 163–166, 2016.

[40] C. E. Tsourakakis, J. Pachocki, and M. Mitzenmacher, "Scalable motif-aware graph clustering," in *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2017, pp. 1451–1460.

[41] N. R. Devanur, S. Dughmi, R. Schwartz, A. Sharma, and M. Singh, "On the approximation of submodular functions," 2013, arXiv preprint arXiv:1304.4948.

[42] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon, "Network motifs: simple building blocks of complex networks," *Science*, vol. 298, no. 5594, pp. 824–827, 2002.

[43] "Florida bay trophic exchange matrix," http://vlado.fmf.uni-lj.si/pub/networks/data/bio/foodweb/Florida.paj.

[44] S. Allesina, A. Bodini, and C. Bondavalli, "Ecological subsystems via graph theory: the role of strongly connected components," *Oikos*, vol. 110, no. 1, pp. 164–176, 2005.

[45] P. Awasthi, A. Blum, O. Sheffet, and A. Vijayaraghavan, "Learning mixtures of ranking models," in *Advances in Neural Information Processing Systems (NIPS)*, 2014, pp. 2609–2617.

[46] C. Meek and M. Meila, "Recursive inversion models for permutations," in *Advances in Neural Information Processing Systems (NIPS)*, 2014, pp. 631–639.

[47] J. Huang, C. Guestrin et al., "Uncovering the riffled independence structure of ranked data," *Electronic Journal of Statistics*, vol. 6, pp. 199–230, 2012.

[48] J. Jiao, K. Venkat, Y. Han, and T. Weissman, "Maximum likelihood estimation of functionals of discrete distributions," *IEEE Transactions on Information Theory*, vol. 63, no. 10, pp. 6774–6798, 2017.

[49] Y. Bu, S. Zou, Y. Liang, and V. V. Veeravalli, "Estimation of KL divergence: Optimal minimax rate," *IEEE Transactions on Information Theory*, vol. 64, no. 4, pp. 2648–2674, 2018.

[50] W. Gao, S. Oh, and P. Viswanath, "Demystifying fixed k-nearest neighbor information estimators," in *IEEE International Symposium on Information Theory (ISIT)*, 2017, pp. 1267–1271.

[51] R. L. Plackett, "The analysis of permutations," *Applied Statistics*, pp. 193–202, 1975.

[52] I. C. Gormley and T. B. Murphy, "A latent space model for rank data," in *Statistical Network Analysis: Models, Issues, and New Directions*. Springer, 2007, pp. 90–102.

[53] T. Kamishima, "Nantonac collaborative filtering: recommendation based on order responses," in *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2003, pp. 583–588.

[54] R. Vidal, "Subspace clustering," *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 52–68, 2011.

[55] G. Chen and G. Lerman, "Spectral curvature clustering (SCC)," *International Journal of Computer Vision (IJCV)*, vol. 81, no. 3, pp. 317–330, 2009.

[56] J. P. Costeira and T. Kanade, "A multibody factorization method for independently moving objects," *International Journal of Computer Vision (IJCV)*, vol. 29, no. 3, pp. 159–179, 1998.

[57] R. Tron and R. Vidal, "A benchmark for the comparison of 3-d motion segmentation algorithms," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007, pp. 1–8.

[58] R. Vidal, Y. Ma, and S. Sastry, "Generalized principal component analysis (GPCA)," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 12, pp. 1945–1959, 2005.

[59] J. Yan and M. Pollefeys, "A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate," in *European Conference on Computer Vision (ECCV)*, 2006, pp. 94–106.

[60] Y. Ma, H. Derksen, W. Hong, and J. Wright, "Segmentation of multivariate mixed data via lossy data coding and compression," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 9, 2007.

[61] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 2790–2797.

[62] P. Purkait, T.-J. Chin, A. Sadri, and D. Suter, "Clustering with hypergraphs: the case for large hyperedges," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016.

[63] S. Amghibech, "Eigenvalues of the discrete p-Laplacian for graphs," *Ars Combinatoria*, vol. 67, pp. 283–302, 2003.

[64] T. Bühler and M. Hein, "Spectral clustering based on the graph p-Laplacian," in *Proceedings of the International Conference on Machine Learning*. ACM, 2009, pp. 81–88.

[65] A. Szlam and X. Bresson, "Total variation and cheeger cuts," in *Proceedings of the International Conference on Machine Learning*, 2010, pp. 1039–1046.

[66] K. C. Chang, "Spectrum of the 1-Laplacian and Cheeger's constant on graphs," *Journal of Graph Theory*, vol. 81, no. 2, pp. 167–207, 2016.

[67] F. Tudisco and M. Hein, "A nodal domain theorem and a higher-order Cheeger inequality for the graph p-Laplacian," *Journal of Spectral Theory*, vol. 8, no. 3, pp. 883–908, 2018.

[68] K. Chang, S. Shao, and D. Zhang, "Nodal domains of eigenvectors for 1-Laplacian on graphs," *Advances in Mathematics*, vol. 308, pp. 529–574, 2017.

[69] M. Hein and T. Bühler, "An inverse power method for nonlinear eigenproblems with applications in 1-spectral clustering and sparse pca," in *Advances in Neural Information Processing Systems*, 2010, pp. 847–855.

[70] P. Li and O. Milenkovic, "Revisiting decomposable submodular function minimization with incidence relations," in *Advances in Neural Information Processing Systems*, 2018, pp. 2237–2247.

[71] P. Li and O. Milenkovic, "Inhomogeneous hypergraph clustering with applications," in *Advances in Neural Information Processing Systems*, 2017, pp. 2305–2315.

[72] A. Asuncion and D. Newman, "UCI machine learning repository," 2007.

[73] S. H. Gould, *Variational Methods for Eigenvalue Problems*. University of Toronto Press, Toronto, 1966, vol. 22, no. 12.

[74] K.-C. Chang, "Variational methods for non-differentiable functionals and their applications to partial differential equations," *Journal of Mathematical Analysis and Applications*, vol. 80, no. 1, pp. 102–129, 1981.

[75] T. Bıyıkoglu, J. Leydold, and P. F. Stadler, "Laplacian eigenvectors of graphs," *Lecture Notes in Mathematics*, vol. 1915, 2007.

[76] E. B. Davies, J. Leydold, and P. F. Stadler, "Discrete nodal domain theorems," *Linear Algebra and its Applications*, vol. 336, no. 1-3, pp. 51–60, 2001.

[77] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems*, 2002, pp. 849–856.

[78] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[79] V. Kolmogorov, "Minimizing a sum of submodular functions," *Discrete Applied Mathematics*, vol. 160, no. 15, pp. 2246–2258, 2012.

[80] R. Nishihara, S. Jegelka, and M. I. Jordan, "On the convergence rate of decomposable submodular function minimization," in *Advances in Neural Information Processing Systems*, 2014, pp. 640–648.

[81] D. R. Karger, "Global min-cuts in RNC, and other ramifications of a simple min-cut algorithm," in *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, vol. 93, 1993, pp. 21–30.

[82] Y. T. Lee, A. Sidford, and S. C.-w. Wong, "A faster cutting plane method and its implications for combinatorial and convex optimization," in *Foundations of Computer Science (FOCS), 2015 IEEE 56th Annual Symposium on*. IEEE, 2015, pp. 1049–1065.

[83] A. Ene, H. Nguyen, and L. A. Végh, "Decomposable submodular function minimization: discrete and continuous," in *Advances in Neural Information Processing Systems*, 2017, pp. 2874–2884.

[84] W. Meyer, "Equitable coloring," *The American Mathematical Monthly*, vol. 80, no. 8, pp. 920–922, 1973.

[85] D. Chakrabarty, P. Jain, and P. Kothari, "Provable submodular minimization using Wolfe's algorithm," in *Advances in Neural Information Processing Systems*, 2014, pp. 802–809.

[86] H. Karimi, J. Nutini, and M. Schmidt, "Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2016, pp. 795–811.

[87] Y. Nesterov, "Efficiency of coordinate descent methods on huge-scale optimization problems," *SIAM Journal on Optimization*, vol. 22, no. 2, pp. 341–362, 2012.

[88] A. Hajnal and E. Szemerédi, "Proof of a conjecture of Erdös," *Combinatorial Theory and Its Applications*, vol. 2, pp. 601–623, 1970.

[89] O. Fercoq and P. Richtárik, "Accelerated, parallel, and proximal coordinate descent," *SIAM Journal on Optimization*, vol. 25, no. 4, pp. 1997–2023, 2015.

[90] H. A. Kierstead, A. V. Kostochka, M. Mydlarz, and E. Szemerédi, "A fast algorithm for equitable coloring," *Combinatorica*, vol. 30, no. 2, pp. 217–224, 2010.

[91] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, and K. Siddiqi, "Turbopixels: Fast superpixels using geometric flows," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 12, pp. 2290–2297, 2009.

[92] W. W. Zachary, "An information flow model for conflict and fission in small groups," *Journal of Anthropological Research*, vol. 33, no. 4, pp. 452–473, 1977.

[93] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," arXiv preprint arXiv:1609.02907, 2016.

[94] N. Yadati, M. Nimishakavi, P. Yadav, A. Louis, and P. Talukdar, "HyperGCN: Hypergraph convolutional networks for semi-supervised classification," arXiv preprint arXiv:1809.02589, 2018.

[95] A. Chambolle and J. Darbon, "On total variation minimization and surface evolution using parametric maximum flows," *International Journal of Computer Vision*, vol. 84, no. 3, p. 288, 2009.

[96] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Reviews of Modern Physics*, vol. 74, no. 1, p. 47, 2002.

[97] R. Johnson and T. Zhang, "On the effectiveness of laplacian normalization for graph semi-supervised learning," *Journal of Machine Learning Research*, vol. 8, no. Jul, pp. 1489–1517, 2007.

[98] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Advances in Neural Information Processing Systems*, 2004, pp. 321–328.

[99] C. Zhang, S. Hu, Z. G. Tang, and T. H. Chan, "Re-revisiting learning on hypergraphs: confidence interval and subgradient method," in *Proceedings of the International Conference on Machine Learning*, 2017, pp. 4026–4034.

[100] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *Journal of Mathematical Imaging and Vision*, vol. 40, no. 1, pp. 120–145, 2011.

[101] S. Fujishige and S. Isotani, "A submodular function minimization algorithm based on the minimum-norm base," *Pacific Journal of Optimization*, vol. 7, no. 1, pp. 3–17, 2011.

[102] P. Wolfe, "Convergence theory in nonlinear programming," *Integer and Nonlinear Programming*, pp. 1–36, 1970.

[103] F. Locatello, M. Tschannen, G. Rätsch, and M. Jaggi, "Greedy algorithms for cone constrained optimization with convergence guarantees," in *Advances in Neural Information Processing Systems*, 2017, pp. 773–784.

[104] Z. Harchaoui, A. Juditsky, and A. Nemirovski, "Conditional gradient algorithms for norm-regularized smooth convex optimization," *Mathematical Programming*, vol. 152, no. 1-2, pp. 75–112, 2015.

[105] T.-H. H. Chan, Z. G. Tang, X. Wu, and C. Zhang, "Diffusion operator and spectral analysis for directed hypergraph Laplacian," *Theoretical Computer Science*, 2019.

[106] R. Andersen, F. Chung, and K. Lang, "Local graph partitioning using PageRank vectors," in *Proceedings of the 47th Annual Symposium on Foundations of Computer Science*. IEEE Computer Society, 2006, pp. 475–486.

[107] P. Li, G. J. Puleo, and O. Milenkovic, "Motif and hypergraph correlation clustering," arXiv preprint arXiv:1811.02089, 2018.

[108] D. Zhou, S. Zhang, M. Y. Yildirim, S. Alcorn, H. Tong, H. Davulcu, and J. He, "A local algorithm for structure-preserving graph cut," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2017, pp. 655–664.

[109] H. Yin, A. R. Benson, J. Leskovec, and D. F. Gleich, "Local higher-order graph clustering," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2017, pp. 555–564.

[110] D. F. Gleich, L.-H. Lim, and Y. Yu, "Multilinear PageRank," *SIAM Journal on Matrix Analysis and Applications*, vol. 36, no. 4, pp. 1507–1541, 2015.

[111] P. Li and O. Milenkovic, "Submodular hypergraphs: p-Laplacians, Cheeger inequalities and spectral clustering," in *Proceedings of the International Conference on Machine learning*, 2018.

[112] Y. Yoshida, "Cheeger inequalities for submodular transformations," in *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*. Society for Industrial and Applied Mathematics, 2019, pp. 2582–2601.

[113] K. Fujii, T. Soma, and Y. Yoshida, "Polynomial-time algorithms for submodular Laplacian systems," arXiv preprint arXiv:1803.10923, 2018.

[114] M. Ikeda, A. Miyauchi, Y. Takai, and Y. Yoshida, "Finding Cheeger cuts in hypergraphs via heat equation," arXiv preprint arXiv:1809.04396, 2018.

[115] T. Soma and Y. Yoshida, "Spectral sparsification of hypergraphs," in *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2019, pp. 2570–2581.

[116] R. Andersen, F. Chung, and K. Lang, "Local partitioning for directed graphs using PageRank," *Internet Mathematics*, vol. 5, no. 1-2, pp. 3–22, 2008.

[117] L. Lovasz and M. Simonovits, "The mixing rate of Markov chains, an isoperimetric inequality, and computing the volume," in *Proceedings of the 31st Annual Symposium on Foundations of Computer Science*. IEEE Computer Society, 1990, p. 1.

[118] A. Gammerman, V. Vovk, and V. Vapnik, "Learning by transduction," in *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc., 1998, pp. 148–155.

[119] T. Joachims, "Transductive learning via spectral graph partitioning," in *Proceedings of the 20th International Conference on Machine Learning*, 2003, pp. 290–297.

[120] D. Zhou, J. Huang, and B. Schölkopf, "Learning with hypergraphs: Clustering, classification, and embedding," in *Advances in Neural Information Processing Systems*, 2007, pp. 1601–1608.

[121] F. H. Clarke, *Optimization and Nonsmooth Analysis*. Siam, 1990, vol. 5.

[122] T. Bühler, S. S. Rangapuram, S. Setzer, and M. Hein, "Constrained fractional set programs and their application in local clustering and community detection," in *Proceedings of the International Conference on Machine Learning*. JMLR. org, 2013, pp. I–624.

[123] S. Fujishige and X. Zhang, "New algorithms for the intersection problem of submodular systems," *Japan Journal of Industrial and Applied Mathematics*, vol. 9, no. 3, p. 369, 1992.

[124] I. Ekeland and R. Temam, *Convex Analysis and Variational Problems*. Siam, 1999, vol. 28.