Almost Linear Time Algorithms for Flows in Graphs

Victor S. Portella

Supervisor: Marcel K. de Carli Silva

February 21, 2017

This work is partially supported by São Paulo Research Foundation (FAPESP) grant nº 2015/24747-3

Abstract

We study the main, high-level ingredients of the nearly-linear time Laplacian solver of Spielman and Teng and their application to finding an approximately maximum flow in a graph in almost-linear time. We start with basic tools from linear algebra, such as properties of symmetric and positive semidefinite matrices, as well as the Moore-Penrose pseudoinverse. We then move to the Laplacian matrix of a graph and some of its applications, such as computing the number of spanning trees (the so-called Matrix Tree Theorem) and approximating the sparse cuts of a graph.

Next we describe the well-known Conjugate Gradient Method, an iterative algorithm to approximate a solution to a linear system, and use this method with preconditioning to construct an efficient Laplacian solver. In the end, we describe an algorithm to find an approximately maximum flow in undirected graphs in almost-linear time with the help of nearly-linear Laplacian solvers and the multiplicative weights update method.

While the main algorithms covered here are not the fastest known, they contain the majority of the ingredients and tools from the latter.

Contents

1 Introduction **1**

2 Preliminaries **2**
2.1 Basic Notation and Definitions 2
2.2 Data Structures 6
2.3 Spectral Decomposition of Symmetric Matrices 7
2.4 Moore-Penrose Pseudoinverse 14
2.5 The Spectrum of the Adjacency Matrix 18
2.6 Incidence Matrices 19

3 The Graph Laplacian **20**
3.1 Flows in Graphs 21
3.2 Electrical Flows 22
3.3 Counting Spanning Trees 25
3.4 Sparse Cuts 27

4 The Conjugate Gradient Method **30**
4.1 Improving Gradient Descent 30
4.2 The Gram-Schmidt Method and Krylov subspaces 32
4.3 The Conjugate Gradient Iteration 35
4.4 Error Analysis with Polynomials 37
4.5 Improving the Analysis with Chebyshev Polynomials 39

5 Fast Laplacian Solvers **43**
5.1 Preconditioning 43
5.2 A Fast Solver 45

6 Maximum Flow in Graphs using Electrical Flows **49**
6.1 Computing Approximately Electrical Flows 49
6.2 Multiplicative Weights Update Method 53
6.3 Calculating an Approximately Maximum Flow 58

References **63**

Chapter 1

Introduction

The idea to call an algorithm efficient if its running time is asymptotically bounded by a polynomial in the input size was first introduced by Edmonds [6] and Cobham [5] in 1965. Since then, this concept was broadly adopted by researchers and algorithm designers. Recently, the problem of processing data sets so large that the use of traditional tools is impractical is becoming more and more important. When it is necessary to handle such data sets, which are usually called *Big Data*, the use of algorithms with quadratic running time is already impractical. Moreover, when seeking efficiency, one may be willing to accept an approximate answer if such an answer can be obtained quickly (or at all). This is one of the motivations behind research in the development of nearly-linear time algorithms, that is, algorithms with input size m that run in time $O(m \log^c m)$ for some positive constant c. In this context we use the $\tilde{O}$ (read soft-O) notation, that is, we say that an algorithm runs in time $\tilde{O}(f(m))$ if it runs in time $O(f(m) \log^c m)$ for some positive constant c.

A major breakthrough in the area of nearly-linear time algorithms was Spielman and Teng's Laplacian solver [13]. This algorithm solves a linear system of the form $Lx = b$, where L is the Laplacian matrix of a graph G, in time $\tilde{O}(m \log(1/\varepsilon))$, where m is the number of edges of G and $\varepsilon > 0$ is the error tolerance. Graph-theoretic ideas such as sparsifiers and low-stretch spanning trees, together with ideas and tools from numerical linear algebra, like preconditioning and the Conjugate Gradient method, are used in the construction of this algorithm. Following their work, many researchers have developed more efficient and simpler solvers [4, 8, 9, 10], bringing the running time down to $O(m \log^3 n)$, where n is the number of vertices of the graph.

Nearly-linear time Laplacian solvers, with the aid of tools from numerical linear algebra and spectral graph theory, have been used as a subroutine in almost linear time algorithms for a host of combinatorial problems. This "Laplacian paradigm", as proposed by Teng [15], is motivating research in algorithms that joins linear algebra and graph theory. Moreover, many classical problems with known exact algorithms close to the best possible running time in the traditional model are being revisited with the goal of developing almost linear time approximation algorithms for them.

One of these revisited problems is that of finding a maximum flow in a graph with capacities on its edges. This is one of the oldest and most studied problems in combinatorial optimization, and many algorithms to other problems solve maximum flow problem instances as a subroutine. Although the maximum flow problem has efficient algorithms that find an exact solution, they have a natural running-time barrier in the general case of $\Omega(mn)$ time (see [7]), which may be prohibitively expensive for massive instances of the problem.

In this monograph, we study the basic properties of the Laplacian matrix of a graph, many of the core ideas used in Spielman and Teng's solver, and later, we describe its applications to the maximum flow problem. In Chapter 2, we recall fundamental properties of symmetric and positive semidefinite matrices. In Chapter 3, we define the Laplacian matrix of a graph and study many of its properties and applications. In Chapter 4, we describe the Conjugate Gradient method, a famous iterative algorithm for solving linear systems. We also look into an application of this method in Chapter 5, where we state Spielman and Teng's fundamental result, describe preconditioning, and construct a $\tilde{O}(m^{4/3} \log 1/\varepsilon)$ Laplacian solver. In Chapter 6, we present the algorithm from [3] that approximately solves the maximum flow problem in time $\tilde{O}(m^{3/2}\varepsilon^{-5/2})$ using electrical flows, Laplacian solvers, and the multiplicative weights update method.

Chapter 2

Preliminaries

2.1 Basic Notation and Definitions

This section contains basic definitions and notation that will be used throughout the remainder of the text. The reader may skip this section and refer back to it when the need arises.

The set of natural numbers is denoted by $\mathbb{N}$, the set of integer numbers by $\mathbb{Z}$, the set of rational numbers by $\mathbb{Q}$, the set of real numbers by $\mathbb{R}$, and the set of complex numbers by $\mathbb{C}$. Let $S \in \{\mathbb{Z}, \mathbb{Q}, \mathbb{R}\}$, and define $S_+ \coloneqq \{\, s \in S : s \geq 0\}$ and $S_{++} \coloneqq \{\, s \in S : s > 0\}$. Define $[n] \coloneqq \{1, \ldots, n\}$ for each $n \in \mathbb{N}$. Throughout this text we will use Minkowski's notation, that is, if S is a set and $f \colon S \to W$ is a function, we define $f(S) \coloneqq \{\, f(s) : s \in S\}$. For example, $[n] - 1 = \{0, 1, \ldots, n-1\}$.

The **Iverson bracket** of a predicate P is defined by

$$[P] = \begin{cases} 1 & \text{if } P \text{ is true,} \\ 0 & \text{otherwise.} \end{cases}$$

Moreover, when P is false, we consider that $[P]$ is *strongly zero*, that is, the whole expression multiplied by $[P]$ is zero, even if there are invalid operations in the expression following the Iverson bracket. One example of a case like that is the expression $[x \neq 0]1/x$ for $x \in \mathbb{R}$, which we take to mean 0 when $x = 0$. Throughout this text,

$$V \text{ denotes an arbitrary finite set,}$$

unless stated otherwise (one may think of it as being a set of vertices of a graph, which shall be defined later). A **partition** of a set V is a collection $\mathcal{S}$ of nonempty subsets of V such that $S \cap T = \emptyset$ for every distinct $S, T \in \mathcal{S}$, and

$$\bigcup_{S \in \mathcal{S}} S = V.$$

Define $\binom{V}{k} \coloneqq \{\, S \subseteq V : |S| = k\}$ for every $k \in \mathbb{N}$.

The set of all functions from a set X to a set Y is denoted by Y^X, and if $X = [n]$, we abbreviate this notation to Y^n. Let $f \colon X \to \mathbb{R}$ be a function. The **support** of f is $\operatorname{supp}(f) \coloneqq \{\, v \in V : f(v) \neq 0\}$. Let $S \subseteq X$. The restriction of f to S is denoted by $f\restriction_S$, and $x^* \in S$ is a **global minimizer** of f over S if $f(x^*) \leq f(x)$ for every $x \in S$. Define

$$\operatorname*{arg\,max}_{x \in S} f(x) \coloneqq \{\, x \in S : f(x) \geq f(y)\ \forall y \in S\},$$

and

$$\operatorname*{arg\,min}_{x \in S} f(x) \coloneqq \{\, x \in S : f(x) \leq f(y)\ \forall y \in S\}.$$

Although $\arg\min_{x \in S} f(x)$ is a set, if $\left|\arg\min_{x \in S} f(x)\right| = 1$, we may write $y = \arg\min_{x \in S} f(x)$ instead of $y \in \arg\min_{x \in S} f(x)$. Analogously for $\arg\max_{x \in S} f(x)$.

The big-O notation is an important tool to help us talk about the running time and space consumption of algorithms, as well as the asymptotic growth of some functions. Let $g\colon \mathbb{R} \to \mathbb{R}$ be a function. Define

$$O(g(x)) \coloneqq \{\, f\colon \mathbb{R} \to \mathbb{R} : \text{there are } x_0 \in \mathbb{R} \text{ and } M \in \mathbb{R}_+ \text{ such that } f(x) \leq Mg(x) \text{ for every } x \geq x_0 \}.$$

Similarly, define

$$\Omega(g(x)) \coloneqq \{\, f\colon \mathbb{R} \to \mathbb{R} : \text{there are } x_0 \in \mathbb{R} \text{ and } M \in \mathbb{R}_+ \text{ such that } f(x) \geq Mg(x) \text{ for every } x \geq x_0 \}.$$

An important definition for this text which is not so well known as the classic Big-O definitions is the soft-O notation, defined by

$$\tilde{O}(g(x)) \coloneqq \bigcup_{k=0}^{\infty} O(g(x) \log^k g(x)),$$

that is, the soft-O notation "hides" the logarithmic terms.

We assume that the reader is familiarized with the definition of a vector space, and we denote the **dimension** of a vector space V by $\dim(V)$. If V is a real vector space, the **span** of a finite set $S \subseteq V$ is the subspace

$$\operatorname{span}(S) \coloneqq \Big\{ \sum_{s \in S} c_s s \in V : c \in \mathbb{R}^S \Big\}.$$

An **inner product** on a vector space V over $\mathbb{R}$ is a function $\langle \cdot, \cdot \rangle\colon V \times V \to \mathbb{R}$ such that

(i) $\langle x, x \rangle \geq 0$ for every $x \in V$, where equality holds if and only if $x = 0$;

(ii) $\langle x, y \rangle = \langle y, x \rangle$ for every $x, y \in V$;

(iii) $\langle \alpha x + \beta y, z \rangle = \alpha \langle x, z \rangle + \beta \langle y, z \rangle$ for every $\alpha, \beta \in \mathbb{R}$ and every $x, y, z \in \mathbb{R}$.

Let $S \subseteq V$, where V is a vector space equipped with a inner product $\langle \cdot, \cdot \rangle$. Define the **orthogonal complement** of S by

$$S^{\perp} \coloneqq \{\, v \in V : \langle v, s \rangle = 0 \text{ for each } s \in S \}.$$

A **norm** on a vector space V over $\mathbb{R}$ is a function $\|\cdot\|\colon V \to \mathbb{R}_+$ such that

(i) $\|x\| \geq 0$ for every $x \in V$, where equality holds if and only if $x = 0$;

(ii) $\|\alpha x\| = |\alpha| \|x\|$ for every $x \in V$ and $\alpha \in \mathbb{R}$;

(iii) $\|x + y\| \leq \|x\| + \|y\|$ for every $x, y \in V$.

This last item is known as the **triangle inequality**. We note that every inner product $\langle \cdot, \cdot \rangle$ on a real vector spacer V induces a norm given by

$$\|x\| \coloneqq \langle x, x \rangle^{1/2}, \qquad \forall x \in V.$$

Let V and W be finite sets. The vector space of $V \times W$ matrices with real entries is denoted by $\mathbb{R}^{V \times W}$. We assume that the reader has a good knowledge about matrices. Let $A \in \mathbb{R}^{V \times W}$. The **rank** of A is the dimension of the vector space spanned by the columns of A, and is denoted by $\operatorname{rank}(A)$. A known result from basic linear algebra that we may use it that the dimension of the vector space spanned by the columns of A is also $\operatorname{rank}(A)$. The **transpose** of A is denoted by A^T. For each $v \in V$ and $w \in W$, the entry in line v and column w of A is denoted by $A_{v,w}$. Denote the identity matrix of appropriate size by I. Let $S \subseteq V$ and let $T \subseteq W$. Note that A is a function $A : V \times W \to \mathbb{R}$. A **submatrix** of A is a restriction of A as a function. We will denote by $A[S, T]$ the restriction $A : S \times T \to \mathbb{R}$. We abbreviate $A[S] \coloneqq A[S, S]$. Define $\overline{S} \coloneqq [m] \setminus S$. If $S = \{i\}$, we may write $\overline{i}$ instead of $\overline{\{i\}}$. Let $A \in \mathbb{R}^{n \times n}$ and let $i, j \in [n]$. The (ij)-**principal minor** of A is $A[\overline{i}, \overline{j}]$. For each $A \in \mathbb{R}^{V \times W}$, define the sets

$$\operatorname{Im}(A) \coloneqq \{\, x \in \mathbb{R}^V : x = Ay \text{ for some } y \in W \} \qquad \text{and} \qquad \operatorname{Null}(A) \coloneqq \{\, y \in \mathbb{R}^W : Ay = 0 \},$$

which we call, respectively, the **image** and the **null space** of A. Let $P \in \mathbb{R}^{V \times V}$. The matrix P is a **projection matrix** or a **projector** if $P^2 = P$. The matrix P is an **orthogonal projector** if it is a projector and $P = P^T$. Note that if P is a projector, then $Pv = v$ for every $v \in \operatorname{Im}(P)$. We say the the matrix P is a projector onto some subspace S of $\mathbb{R}^V$ if $S = \operatorname{Im}(P)$. Note that if P is an orthogonal projector, then $\operatorname{Null}(P) = \operatorname{Im}(P)^\perp$. Moreover, one can show that the orthogonal projector onto a subspace S is unique, which we denote by Proj_S.

A **vector** is a $V \times [1]$ matrix. We identify $\mathbb{R}^{V\times[1]}$ with $\mathbb{R}^V$, and we equip $\mathbb{R}^V$ with the **euclidean** (or **standard**) inner-product, which is defined by

$$\langle x, y\rangle = x^T y \qquad \forall x, y \in \mathbb{R}^V.$$

If $x \in \mathbb{R}^V$ and $v \in V$, the v-th entry of x is denoted by x_v. We note that functions of one variable can be seen as vectors, and that the notation of vectors and functions will be used interchangeably. We denote by $\mathbb{1}$ the vector of appropriate size with ones in all its coordinates. Let $i \in V$ and define $e_i \in \mathbb{R}^V$ by $(e_i)_j := [i = j]$ for each $j \in V$. The set where a vector e_i lies in will not be explicitly stated when it is clear from context. If $\langle\cdot,\cdot\rangle$ is an inner-product on a real vector space V, then a set $S \subseteq V$ is **orthogonal** (with respect to $\langle\cdot,\cdot\rangle$) if $\langle u, v\rangle = 0$ for every distinct $u, v \in S$. When the inner-product is not explicitly stated, we assume it to be the euclidean inner-product. A subset of a vector space is **orthonormal** if it is orthogonal and each of its elements has norm 1. The **Hadamard product** $x \odot y \in \mathbb{R}^V$ of two vectors $x, y \in \mathbb{R}^V$ is defined by $(x \odot y)_i := x_i y_i$ for every $i \in V$. Let $f \in \mathbb{R}^V$. Define $\operatorname{sgn}\colon \mathbb{R}^V \to \{\pm 1\}^V$ by $\operatorname{sgn}(f)_i := (-1)^{[f_i<0]}$ for each $i \in V$.

Theorem 2.1. If $A \in \mathbb{R}^{m\times n}$, then $\operatorname{Im}(A) = \operatorname{Null}(A^T)^\perp$.

Proof. First, let us show that

$$\operatorname{Im}(A) \subseteq \operatorname{Null}(A^T)^\perp. \tag{2.1}$$

Let $x \in \operatorname{Im}(A)$. By definition of $\operatorname{Im}(A)$, there is $y \in \mathbb{R}^n$ such that $x = Ay$. Hence, for each $z \in \operatorname{Null}(A^T)$,

$$\langle x, z\rangle = \langle Ay, z\rangle = (Ay)^T z = y^T A^T z = y^T 0 = 0.$$

This ends the proof of (2.1). Let us now prove that

$$\operatorname{Null}(A^T)^\perp \subseteq \operatorname{Im}(A). \tag{2.2}$$

We have that (2.2) holds if and only if $\operatorname{Im}(A)^\perp \subseteq \operatorname{Null}(A^T)^{\perp\perp}$, and the later set is $\operatorname{Null}(A^T)$. Let $z \in \operatorname{Im}(A)^\perp$. For every $y \in \mathbb{R}^n$, since $Ay \in \operatorname{Im}(A)$, we have

$$0 = \langle z, Ay\rangle = z^T Ay = (A^T z)^T y = \langle A^T z, y\rangle.$$

Since this holds for every $y \in \mathbb{R}^n$, we conclude that $A^T z = 0$. Hence, $z \in \operatorname{Null}(A^T)$. ☐

Whenever it is possible (and convenient), we will use Householder's convention:

- greek letters for scalars, e.g. $\alpha, \beta \in \mathbb{R}$;
- lower case letters for vectors, e.g. $x, y \in \mathbb{R}^n$;
- upper case letters for matrices, e.g. $A, B \in \mathbb{R}^{n\times n}$.

The function $\operatorname{diag}\colon \mathbb{R}^{V\times V} \to \mathbb{R}^V$ extracts the diagonal of a matrix, and $\operatorname{Diag}\colon \mathbb{R}^V \to \mathbb{R}^{V\times V}$ is defined by $\operatorname{Diag}(x)_{i,j} := [i = j]x_i$ for every $x \in \mathbb{R}^V$ and $i, j \in V$. For each $S \subseteq V$, define $\mathbb{1}_S \in \{0,1\}^V$ by $(\mathbb{1}_S)_i := [i \in S]$ for every $i \in V$. A relation which is simple but of fundamental importance for the text is that, for any $A \in \mathbb{R}^{V\times V}$ and any $x \in \mathbb{R}^V$, we have

$$x^T Ax = \sum_{i\in V}\sum_{j\in V} x_i A_{i,j} x_j.$$

If $A = \operatorname{Diag}(y)$ for some $y \in \mathbb{R}^V$, we have

$$x^T \operatorname{Diag}(y)x = \sum_{i\in V} x_i^2 y_i.$$

Graphs are one of the objects of major interest throughout this text. A **(simple) graph** is an ordered triple $G = (V, E, \psi)$, where V and E are disjoint sets and $\psi\colon E \to \binom{V}{2}$ is a injective function. The elements of V and E are the **vertices** and the **edges** of G, respectively, and ψ is the **incidence function** of G. An edge $e \in E$ is **incident** to a vertex $i \in V$ if $i \in \psi(e)$. We say that $i, j \in V$ are **adjacent** if $\psi(e) = \{i, j\}$ for some $e \in E$. For a graph G, its vertex set is denoted by $V(G)$, its edge set by $E(G)$, and its incidence function by ψ_G. A **multigraph** is an ordered triple $G = (V, E, \psi)$ which is defined in a similar way to graphs, but the incidence function goes from E to $\binom{V}{2} \cup \binom{V}{1}$, and it does not need to be injective. Moreover, most of the definitions for graphs may be adapted for multigraphs, and any differences will be explicitly stated. If G is a multigraph, then $e \in E(G)$ is a **loop** if $|\psi_G(e)| = 1$. Often it will be more convenient to say that a graph G is an ordered pair (V, E) where $E \coloneqq \{\, \psi(e) : e \in E\}$ (which is a multiset in the case of a multigraph). Moreover, we may write $ij \in E$ instead of $\{i, j\} \in E$. A **digraph** is an ordered triple $D = (V, A, \psi)$ defined in a similar way to a graph, but the incidence function ψ goes from A to $V \times V$, and the elements of A are the **arcs** of D, which is denoted by $A(D)$. Often it will be more convenient to say that a digraph D is an ordered pair (V, A), where $A \coloneqq \{\, \psi(a) : a \in A(D)\}$ is a multiset. If $G = (V, E), \psi$ is a graph and $S \subseteq V$ is a set, then the **cut** (associated with S) is the set of edges

$$\delta(S) \coloneqq \{\, e \in E : \psi(e) = ij \text{ with } i \in S \text{ and } j \in V \setminus S\}.$$

Moreover, if $D = (V, A, \psi)$ is a digraph, $S \subseteq V$, and $\overline{S} \coloneqq V \setminus S$, then define the sets

$$\delta^{\text{in}}(S) \coloneqq \{\, a \in A : \psi(a) \in \overline{S} \times S\} \qquad \text{and} \qquad \delta^{\text{out}}(S) \coloneqq \{\, a \in A : \psi(a) \in S \times \overline{S}\}.$$

An **orientation** of a graph $G = (V, E, \psi)$ is a digraph $\vec{G} = (V, E, \psi')$ such that for every $e \in E$ we have that $\psi'(e) = (i, j)$ or $\psi'(e) = (j, i)$, where $\{i, j\} = \psi(e)$. A **weighted graph** is a quadruple $G = (V, E, \psi, w)$ where (V, E, ψ) is a graph and $w \in \mathbb{R}^E_{++}$ is a vector of weights on the edges of G. A **weighted digraph** is defined analogously.

Let $G = (V, E, \psi)$ be a graph. The **degree** (or **valency**) $\deg_G(i)$ of a vertex $i \in V$ is the number of edges incident to i. The **neighborhood** $N_G(i)$ of $i \in V$ is the subset of vertices of G which are adjacent to i. The subscript may be omitted when it is clear from context. We define $\Delta(G) \coloneqq \max_{i \in V} \deg_G(i)$. The graph G is **regular** if $\deg_G = \Delta(G)\mathbb{1}$. A **subgraph** $H = (V', E', \phi)$ of G is a graph such that $V' \subseteq V$, $E' \subseteq E$ and $\phi = \psi\restriction_{E'}$. The subgraph of G **induced** by $S \subseteq V$ is the graph $G[S] \coloneqq (S, E', \psi\restriction_{E'})$, where $E' \coloneqq \{\, e \in E : \psi(e) \subseteq S\}$. Let $S \subseteq V$ and define $G - S \coloneqq G[V \setminus S]$. We may write $G - i$ instead of $G - \{i\}$. Let $D = (V, E, \phi)$ be a digraph. A **walk** (from v_0 to v_k) in G (resp., in D) is a sequence $P = (v_0, f_1, v_1, f_2, \ldots, f_k, v_k)$ where $v_i \in V$ for each $i \in \{0\} \cup [k]$ and, for every $i \in [k]$, we have $\psi(f_i) = v_{i-1}v_i$ (resp., $\psi(f_i) = (v_{i-1}, v_i)$). The **endpoints** of P are v_0 and v_k. We may omit the edges in a walk when G (resp. D) is simple, thus writing $P = (v_0, v_1, \ldots, v_k)$. The **length** $|P|$ of a walk $P = (v_0, f_1, v_1, f_2, \ldots, f_k, v_k)$ is k. A **trail** is a walk with no repeated edges. A **path** is a walk with no repeated vertices. A walk is **closed** if its endpoints are equal. A **circuit** is a closed trail with no repeated vertices (besides its endpoints). Let $D = (V, A)$ be a digraph and let $v_0, v_k \in V$. A walk is **directed** if it is a walk in a digraph. If P is a walk (or a directed walk), we denote by $V(P)$ the set of vertices in P and by $E(P)$ the set of edges in P (or by $A(P)$ the set of arcs in P in the directed case). A graph G is **connected** if there is a walk in G with endpoints i and j for every $i, j \in V(G)$. A **component** of a graph G is a maximal connected subgraph of G. A graph (or digraph) is **acyclic** if it has no circuits (or directed circuits for digraphs). A **tree** is a connected acyclic graph, and a vertex of degree 1 in a tree is a **leaf**. The **distance** between two distinct vertices in a graph is the minimum length of a path between these two vertices. Let $G = (V, E, \psi)$ be a graph. A subset of vertices $S \subseteq V$ is **independent** if for every $i, j \in S$, there is no $e \in E$ such that $\psi(e) = ij$. A **bipartition** of a graph G is a partition of $V(G)$ into two independent sets. A graph is **bipartite** if it has a bipartition.

Lemma 2.2. For each $x \in \mathbb{R}$,

$$e^x \geq 1 + x.$$

Proof. Let us divide the proof in three cases. For $x \leq -1$ the statement is trivially true. Suppose that $x \geq 0$. By definition,

$$e^x = \sum_{i=0}^{\infty} \frac{x^i}{i!}.$$

Since $x \geq 0$, each term of this series is nonnegative. Hence,

$$e^x = \sum_{i=0}^{\infty} \frac{x^i}{i!} = 1 + x + \sum_{i=2}^{\infty} \frac{x^i}{i!} \geq 1 + x.$$

Suppose now that $-1 < x < 0$ and define $y \coloneqq -x$. Hence, $0 < y < 1$ and

$$e^x = e^{-y} = \sum_{i=0}^{\infty} \frac{(-y)^i}{i!}.$$

It is easy to see that the terms of this alternating serie decreases in modulus since $0 < y < 1$. Hence,

$$\sum_{i=2}^{\infty} \frac{(-y)^i}{i!} \geq 0 \implies e^{-y} = 1 - y + \sum_{i=2}^{\infty} \frac{(-y)^i}{i!} \geq 1 - y. \qquad \square$$

2.2 Data Structures

We describe here data structures to represent some of the objects used in the algorithms described in this text. The claims we use about the running time of many operations are valid using these data structures (but one may use other data structure with similar running times). We will make it explicit when a result depends on the data structures of this section. We will use arrays with indexes starting at 1.

To represent graphs and digraphs, we use **adjacency lists**. Let $G = (V, E)$ be a graph, and set $m \coloneqq |E|$ and $n \coloneqq |V|$. To store a graph $G = (V, E)$, for each $i \in V$ we maintain a linked list of the vertices $j \in V$ such that $ij \in E$. It takes time $O(m + n)$ to construct this data structure, and it takes time $O(m)$ to traverse all the edges in this data structure. Moreover, depth-first search and breadth-first search run in time $O(m + n)$ on graphs represented by adjacency lists. To store a digraph $D = (V, A)$, for each $i \in V$ we maintain a linked list of the vertices $j \in V$ such that $(i, j) \in A$.

Most of the matrices we manipulate in this text are sparse, that is, most of the entries of the matrix are zero. On these cases, storing the matrix in a 2-dimesional array will have many entries with zeros, which is inefficient for many reasons. We can improve this by exploiting the sparsity of the matrix by using a special data structure to store these matrices called **Compressed Sparse Row (CSR)**. To store a sparse matrix $A \in \mathbb{R}^{m \times n}$ with $k \in \mathbb{N}$ nonzero entries, this data structure uses three arrays VA, RA, and CA defined in the following way:

- VA has size k and, for each $i \in [k]$, $\mathit{VA}[i]$ stores the i-th nonzero entry of A in a left-to-right top-to-bottom order;
- RA has size $k + 1$ and is defined recursively as follows
  - $\mathit{RA}[1] \coloneqq 0$;
  - $\mathit{RA}[i] \coloneqq \mathit{RA}[i-1] + z_{i-1}$ for each $i \in [k+1] \setminus \{1\}$, where $z_i \in \mathbb{N}$ is the number of nonzero elements on the i-th row of A.
- CA has size k and, for each $i \in [k]$, $\mathit{CA}[i]$ stores the index of the column of the i-th nonzero entry of A in a left-to-right top-to-bottom order.

This way, for each $i \in [k]$ we will have that

$$\mathit{VA}[\mathit{RA}[i] + 1, \dots, \mathit{RA}[i+1]] \text{ are the nonzero elements of the } i\text{-th row of } A, \tag{2.3}$$

and the index of the column of the entry stored in $\mathit{VA}[i]$ is $\mathit{CA}[i]$. For example, for the matrix

$$A \coloneqq \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 6 & 0 \\ 4 & 0 & 0 & 0 \end{pmatrix},$$

its representation with the CSR data structure would be

$$VA = \begin{pmatrix}1 & 2 & 6 & 4\end{pmatrix}, \quad RA = \begin{pmatrix}0 & 0 & 2 & 3 & 4\end{pmatrix}, \quad CA = \begin{pmatrix}2 & 4 & 3 & 1\end{pmatrix}.$$

To left-multiply a vector (represented by an array) by a matrix represented by CSR, one just iterates through the rows of the matrix using (2.3), and since we have access to the indices of the columns of each element, we can easily do the inner product of this row with a vector. Hence, this data structure allow us to do left matrix-vector multiplication in time $O(\max\{k, m\})$. Moreover, if A is a triangular matrix, it is easy to see how to solve a system of the type $Ax = b$ in $O(\max\{k, m\})$ through backwards or forward substitution.

2.3 Spectral Decomposition of Symmetric Matrices

Let $A \in \mathbb{R}^{V \times V}$, and set $n := |V|$. The **eigenvalues** of A are the n roots of the polynomial

$$\lambda \mapsto \det(\lambda I - A).$$

A nonzero vector $v \in \mathbb{R}^V$ such that $Av = \lambda v$ for some eigenvalue λ of A is called an **eigenvector** of A associated to λ. The matrix A is **symmetric** if $A = A^T$, and we denote the set of all real symmetric $V \times V$ matrices by $\mathbb{S}^V$. It can be proved that if $A \in \mathbb{S}^V$, then all the eigenvalues of A are real. The function $\lambda^{\downarrow} \colon \mathbb{S}^V \to \mathbb{R}^{|V|}$ extracts all the eigenvalues of a matrix in non-increasing order. The function $\lambda^{\uparrow}$ is defined analogously with non-decreasing order. Define $\lambda_{\max} := \lambda_1^{\downarrow}$ and $\lambda_{\min} := \lambda_1^{\uparrow}$. An interesting property is that, for any matrices $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{n \times m}$, the eigenvalues of AB are the same of BA, except maybe for the multiplicity of the eigenvalue 0. We will now prove this result in the case of square matrices.

Proposition 2.3. Let $A \in \mathbb{R}^{n \times n}$ be a matrix. Then there exists a sequence $(A_k)_{k=0}^{\infty}$ of invertible matrices in $\mathbb{R}^{n \times n}$ distinct from A such that $\lim_{k \to \infty} A_k = A$.

Proof. Let $p(\lambda) = \det((1 - \lambda)A + \lambda I)$. Note that $p(\lambda)$ is a polynomial in λ of degree at most n, so $p(\lambda)$ has at most n real roots. Let r be the smallest positive real root of $p(\lambda)$ if one exists, otherwise define $r := 1$. Let

$$A_k := \left(1 - \frac{r}{k+2}\right)A + \frac{r}{k+2}I \qquad \forall k \in \mathbb{N}.$$

Note that $r/(k+2) \in (0, r)$ for all $k \in \mathbb{N}$. Therefore,

$$p\left(\frac{r}{k+2}\right) \neq 0 \qquad \forall k \in \mathbb{N}.$$

Hence, A_k is invertible of all $k \in \mathbb{N}$. Moreover,

$$\lim_{k \to \infty} A_k = A. \qquad \square$$

Lemma 2.4. If $A, B \in \mathbb{R}^{n \times n}$, then AB and BA have the same eigenvalues.

Proof. Let $(B_k)_{k=0}^{\infty}$ be a sequence of invertible matrices as in Proposition 2.3 that converges to B, let $k \in \mathbb{N}$, and let $t \in \mathbb{C}$. Then

$$\begin{aligned}\det(AB_k - tI) &= \det(AB_k - tB_k^{-1}B_k) = \det((A - tB_k^{-1})B_k) = \det(A - tB_k^{-1})\det(B_k) \\ &= \det(B_k(A - tB_k^{-1})) = \det(B_kA - tI).\end{aligned}$$

Hence, $\det(AB_k - tI) = \det(B_kA - tI)$. Since the determinant of a matrix is in fact a polynomial in the entries of A, it is then a continuous function. Hence, we find that

$$\det(AB - tI) = \lim_{k \to \infty} \det(AB_k - tI) = \lim_{k \to \infty} \det(B_kA - tI) = \det(BA - tI). \qquad \square$$

A matrix $Q \in \mathbb{R}^{V \times W}$ with $|V| = |W|$ is **orthogonal** if $Q^T Q = I$. Note that if a matrix Q is orthogonal, then its columns form a orthonormal set. Moreover, the **trace** $\operatorname{Tr} : \mathbb{R}^{V \times V} \to \mathbb{R}$ is defined by

$$\operatorname{Tr}(A) = \sum_{i \in V} A_{ii}, \qquad \forall A \in \mathbb{R}^{V \times V}.$$

Let $A \in \mathbb{R}^{V \times W}$ and $B \in \mathbb{R}^{W \times V}$. The fact that

$$\operatorname{Tr}(AB) = \sum_{i \in V} \sum_{j \in W} A_{i,j} B_{j,i} = \sum_{j \in W} \sum_{i \in V} B_{j,i} A_{i,j} = \operatorname{Tr}(BA)$$

may be used without mention. We state the following theorem without proof.

Theorem 2.5 (Spectral Decomposition)**.** If $A \in \mathbb{S}^n$, then there exists an orthogonal matrix $Q \in \mathbb{R}^{n \times n}$ such that $A = Q \operatorname{Diag}(\lambda^{\downarrow}(A)) Q^T$. In particular, there exists an orthonormal basis $\{q_1, \ldots, q_n\}$ of $\mathbb{R}^n$ such that $A = \sum_{i=1}^n \lambda_i^{\downarrow}(A) q_i q_i^T$.

Corollary 2.6. If $A \in \mathbb{S}^n$, then $\operatorname{Tr}(A) = \mathbb{1}^T \lambda^{\downarrow}(A)$ and $\det(A) = \prod_{i=1}^n \lambda_i^{\downarrow}(A)$.

Proof. By Theorem 2.5, we have $A = Q \operatorname{Diag}(\lambda^{\downarrow}(A)) Q^T$, for some orthogonal matrix $Q \in \mathbb{R}^{n \times n}$. Hence,

$$\operatorname{Tr}(A) = \operatorname{Tr}(Q \operatorname{Diag}(\lambda^{\downarrow}(A)) Q^T) = \operatorname{Tr}(\operatorname{Diag}(\lambda^{\downarrow}(A)) Q^T Q) = \operatorname{Tr}(\operatorname{Diag}(\lambda^{\downarrow}(A))) = \mathbb{1}^T \lambda^{\downarrow}(A)$$

and

$$\begin{aligned} \det(A) &= \det(Q \operatorname{Diag}(\lambda^{\downarrow}(A)) Q^T) = \det(Q) \det(\operatorname{Diag}(\lambda^{\downarrow}(A))) \det(Q^T) \\ &= \det(\operatorname{Diag}(\lambda^{\downarrow}(A))) \det(Q Q^T) = \det(\operatorname{Diag}(\lambda^{\downarrow}(A))) = \prod_{i=1}^n \lambda_i^{\downarrow}(A). \end{aligned}$$ □

Corollary 2.7. If $A \in \mathbb{S}^n$, then

$$\lambda_{\max}(A) = \max_{x \in \mathbb{R}^n \setminus \{0\}} \frac{x^T A x}{x^T x} \qquad \text{and} \qquad \lambda_{\min}(A) = \min_{x \in \mathbb{R}^n \setminus \{0\}} \frac{x^T A x}{x^T x}. \tag{2.4}$$

Proof. Let $Q \in \mathbb{R}^{n \times n}$ be an orthogonal matrix as in Theorem 2.5 and let $x \in \mathbb{R}^n \setminus \{0\}$. Then $x = Qc$ for $c := Q^T x$. Therefore,

$$x^T A x = x^T Q \operatorname{Diag}(\lambda^{\downarrow}(A)) Q^T x = c^T \operatorname{Diag}(\lambda^{\downarrow}(A)) c. \tag{2.5}$$

Moreover,

$$c^T \operatorname{Diag}(\lambda^{\downarrow}(A)) c = \sum_{i=1}^n \lambda_i^{\downarrow}(A) c_i^2 \leq \lambda_{\max}(A) \sum_{i=1}^n c_i^2 = \lambda_{\max}(A) c^T c = \lambda_{\max}(A) x^T Q^T Q x = \lambda_{\max}(A) x^T x \tag{2.6}$$

with equality if $c = e_1$, i.e. $x = Qe_1$. Analogously,

$$c^T \operatorname{Diag}(\lambda^{\downarrow}(A)) c \geq \lambda_{\min}(A) x^T x, \tag{2.7}$$

with equality if $c = e_n$, that is, if $x = Qe_n$. Hence,

$$\lambda_{\min}(A) x^T x \overset{(2.7)}{\leq} c^T \operatorname{Diag}(\lambda^{\downarrow}(A)) c \overset{(2.6)}{\leq} \lambda_{\max}(A) x^T x, \tag{2.8}$$

where the second inequality holds with equality for $x = Qe_1$, and the first inequality holds with equality for $x = Qe_n$. Therefore, (2.8) together with (2.5) yields

$$\lambda_{\min}(A) \leq \frac{x^T A x}{x^T x} \leq \lambda_{\max}(A), \tag{2.9}$$

where equality holds in the cases cited above. □

A generalization of the above corollary is the following theorem, which we state without proof.

Theorem 2.8. Let $A \in \mathbb{S}^n$. If $\{q_1, \cdots, q_n\}$ is an orthonormal set such that q_i is an eigenvector of A associated with $\lambda_i^{\uparrow}(A)$ for each $i \in [n]$, then for every $k \in [n]$,

$$\begin{aligned}\lambda_k^{\uparrow}(A) &= \min\left\{ \frac{v^T A v}{v^T v} : v \in \mathbb{R}^n \setminus \{0\}, v^T q_i = 0 \ \forall i \in [k-1] \right\} \\ &= \max\left\{ \frac{v^T A v}{v^T v} : v \in \mathbb{R}^n \setminus \{0\}, v^T q_i = 0 \ \forall i \in [n] \setminus [k] \right\}.\end{aligned}$$

The **operator norm** of $A \in \mathbb{R}^{m \times n}$ is

$$\|A\|_2 \coloneqq \max_{\|x\| \le 1} \|Ax\|. \tag{2.10}$$

Corollary 2.9. If $A \in \mathbb{S}^n$, then $\|A\|_2 = \max\{|\lambda_{\max}(A)|, |\lambda_{\min}(A)|\}$.

Proof. Note that the maximum in (2.10) is attained by a unit vector. Let $Q \in \mathbb{R}^{n \times n}$ be an orthogonal matrix as in Theorem 2.5. Then,

$$\begin{aligned}\|A\|_2^2 &= \max_{\|x\|=1} \|Ax\|^2 = \max_{\|x\|=1} \|Q \operatorname{Diag}(\lambda^{\downarrow}(A)) Q^T x\|^2 \\ &\le \max_{\|x\|=1} \|Q\|_2^2 \|\operatorname{Diag}(\lambda^{\downarrow}(A))\|_2^2 \|Q^T\|_2^2 \|x\|^2 = \|\operatorname{Diag}(\lambda^{\downarrow}(A))\|_2^2 \\ &= \max_{\|y\|=1} y^T \operatorname{Diag}(\lambda^{\downarrow}(A))^2 y = \max_{y \in \mathbb{R}^n \setminus \{0\}} \frac{y^T \operatorname{Diag}(\lambda^{\downarrow}(A))^2 y}{y^T y} \\ &= \lambda_{\max}(\operatorname{Diag}(\lambda^{\downarrow}(A))^2) = \max\{\lambda_{\min}(A)^2, \lambda_{\max}(A)^2\}.\end{aligned}$$

Therefore,

$$\|A\|_2 \le \max\{|\lambda_{\min}(A)|, |\lambda_{\max}(A)|\}.$$

It only remains to show that $\|A\|_2 \ge \max\{|\lambda_{\min}(A)|, |\lambda_{\max}(A)|\}$. Let $q_{\min}, q_{\max} \in \mathbb{R}^n$ be unit eigenvectors associated, respectively, with the eigenvalues $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$. Then,

$$\|Aq_{\min}\| = |\lambda_{\min}(A)| \qquad \text{and} \qquad \|Aq_{\max}\| = |\lambda_{\max}(A)|. \qquad \square$$

A matrix $A \in \mathbb{S}^n$ is **semidefinite** if $x^T A x \ge 0$ for every $x \in \mathbb{R}^n$ or if $x^T A x \le 0$ for every $x \in \mathbb{R}^n$. A matrix A is **indefinite** if it is not semidefinite. If $x^T A x \ge 0$ for every $x \in \mathbb{R}^n \setminus \{0\}$, then A is **positive semidefinite**. Similarly, if $x^T A x \le 0$ for every $x \in \mathbb{R}^n \setminus \{0\}$, then A is **negative semidefinite**. In the case where the inequalities are strict the matrix A is **positive definite** or **negative definite**, respectively. For every $A, B \in \mathbb{S}^n$, we write $A \succeq B$ or $A \succ B$ if $A - B$ is positive semidefinite or positive definite, respectively. Denote by $\mathbb{S}^n_+$ the set of positive semidefinite matrix in $\mathbb{S}^n$. Similarly, denote by $\mathbb{S}^n_{++}$ the set of the positive definite matrices on $\mathbb{S}^n$. We may use the next proposition without mentioning it.

Proposition 2.10. Let $X \in \mathbb{S}^n$ and let $L \in \mathbb{R}^{m \times n}$. If $X \succeq 0$, then $LXL^T \succeq 0$. Moreover, if $m = n$ and L is non-singular, then $LXL^T \succeq 0$ implies that $X \succeq 0$.

Proof. If $X \succeq 0$, then for every $h \in \mathbb{R}^m$ we have that

$$h^T LXL^T h = (L^T h) X (L^T h) \ge 0.$$

Using what we just proved, if $m = n$ and L is non-singular, then LXL^T implies that $X = L^{-1} LXL^T (L^T)^{-1} \succeq 0$. $\square$

Let $A \in \mathbb{S}^n_+$. A matrix $A^{1/2} \in \mathbb{S}^n_+$ is a **square root** of A if $(A^{1/2})^2 = A$. The next proposition shows that such a matrix is unique, and it shows how to construct it from the spectral decomposition of the matrix.

Proposition 2.11. Let $A \in \mathbb{S}^n_+$. Then A has a unique square root matrix $A^{1/2} \in \mathbb{S}^n_+$. Moreover, if $A = Q \operatorname{Diag}(\lambda^{\downarrow}(A)) Q^T$, where $Q \in \mathbb{R}^{n \times n}$ is an orthogonal matrix, then $A^{1/2} = Q \operatorname{Diag}(\mu) Q^T$, where $\mu \in \mathbb{R}^n$ is defined by $\mu_i \coloneqq \lambda_i^{\downarrow}(A)^{1/2}$ for each $i \in [n]$, and $\operatorname{Im}(A) = \operatorname{Im}(A^{1/2})$.

Proof. Let $Q \in \mathbb{R}^{n\times n}$ be an orthogonal matrix such that $A = Q\operatorname{Diag}(\lambda^{\downarrow}(A))Q^T$, and define $q_i := Qe_i$ for each $i \in [n]$. Let $\Lambda := \{\, \lambda_i^{\downarrow}(A) : i \in [n]\}$, and let $\mathcal{Q}(\lambda) := \{\, q_i : i \in [n] \text{ and } Aq_i = \lambda q_i\}$ for each $\lambda \in \Lambda$. Define $P_\lambda = \sum_{q\in\mathcal{Q}(\lambda)} qq^T$ for each $\lambda \in \Lambda$. Note that P_λ is the orthogonal projector onto the subspace spanned by the eigenvectors of A associated with $\lambda \in \Lambda$. Hence,

$$A = Q\operatorname{Diag}(\lambda^{\downarrow}(A))Q^T = \sum_{i=1}^{n} \lambda_i^{\downarrow}(A) q_i q_i^T = \sum_{\lambda\in\Lambda} \lambda P_\lambda.$$

Moreover, define $\mu \in \mathbb{R}^n$ by $\mu_i := \lambda_i^{\downarrow}(A)^{1/2}$ for each $i \in [n]$. Then,

$$A^{1/2} = Q\operatorname{Diag}(\mu)Q^T = \sum_{i=1}^{n} \lambda_i^{\downarrow}(A)^{1/2} q_i q_i^T = \sum_{\lambda\in\Lambda} \lambda^{1/2} P_\lambda.$$

Let us show that

$$\text{if } B \in \mathbb{S}^n_+ \text{ is such that } B^2 = A, \text{ then } B = \textstyle\sum_{\lambda\in\Lambda} \lambda^{1/2} P_\lambda. \tag{2.11}$$

Let $B \in \mathbb{S}^n_+$ be such that $B^2 = A$, and let $x \in \mathbb{R}^n$ be an eigenvector of B associated with an eigenvalue $\mu \in \mathbb{R}_+$ of B. Note that $Ax = B^2x = \mu^2 x$. Hence, if $x \in \mathbb{R}^n$ is an eigenvector of B associated with the eigenvalue $\mu \in \mathbb{R}_+$, then x is an eigenvector of A associated with the eigenvalue μ^2. Hence, for each $i \in [n]$,

$$\operatorname{Null}(\lambda_i^{\downarrow}(B)I - B) \subseteq \operatorname{Null}(\lambda_i^{\downarrow}(B)^2 I - A). \tag{2.12}$$

Let us show that

$$\text{equality holds in (2.12) for every } i \in [n]. \tag{2.13}$$

For each $i \in [n]$, define $L_i := \operatorname{Null}(\lambda_i^{\downarrow}(B)I - B)$ and $R_i := \operatorname{Null}(\lambda_i^{\downarrow}(B)^2 I - A)$. Since A and B are symmetric, we have that if $i, j \in [n]$ are distinct, $x \in L_i$, and $y \in L_j$, then $x^T y = 0$. Analogously, we have that if $i, j \in [n]$ are distinct, $x \in R_i$, and $y \in R_j$, then $x^T y = 0$. Hence,

$$L_i \subseteq L_j^{\perp} \text{ and } R_i \subseteq R_j^{\perp}, \qquad \text{for every distinct } i, j \in [n]. \tag{2.14}$$

By Theorem 2.5, there is a basis $\{u_1, \dots, u_n\}$ of $\mathbb{R}^n$ such that, for each $i \in [n]$, there is $j \in [n]$ such that $u_i \in L_j$. Hence,

$$\text{if } x \in \mathbb{R}^n, \text{ then there are } l_1, \dots, l_n \in \mathbb{R}^n \text{ such that } l_i \in L_i \text{ for each } i \in [n], \text{ and } x = \textstyle\sum_{i=1}^n l_i. \tag{2.15}$$

Suppose there is $x \in R_j \setminus \{0\}$ such that $x \notin L_j$ for some $j \in [n]$. We may suppose that $x \in L_j^{\perp}$ since we may take the orthogonal projection of x onto $L_j^{\perp}$, and such a projection is not zero since $x \notin L_j$. By (2.15), there are $l_1, \dots, l_n \in \mathbb{R}^n$ such that $l_i \in L_i$ for each $i \in [n]$, and $x = \sum_{i=1}^n l_i$. Since $x \in Ł_j^{\perp}$, we have that

$$0 = l_j^T x = \sum_{i=1}^{n} l_j^T l_i = \|l_j\|^2 \implies l_j = 0.$$

Since $x \neq 0$, there is $k \in [n] \setminus \{j\}$ such that $l_k \neq 0$. Since $l_k \in L_k \subseteq R_k \subseteq R_j^{\perp}$ by (2.14), and since $x \in R_j$, we have that

$$0 = l_k^T x = \sum_{i=1}^{n} l_k^T l_i = \|l_k\|^2 \neq 0,$$

a contradiction. This ends the proof of (2.13). Hence, for each $i \in [n]$, we have $\lambda_i^{\downarrow}(B)^2 = \lambda_i^{\downarrow}(A)$ and since $\lambda^{\downarrow}(B) \geq 0$, we conclude that $\lambda_i^{\downarrow}(B) = \lambda_i^{\downarrow}(A)^{1/2}$ for each $i \in [n]$. Therefore, for each $\lambda \in \Lambda$, we have $Bq = \lambda^{1/2} q$ for every $q \in \mathcal{Q}(\lambda)$. Since $\{q_1, \dots, q_n\}$ is a basis of $\mathbb{R}^n$, this ends the proof of (2.11). □

Proposition 2.12. If $A \in \mathbb{S}^n_+$ and $x \in \mathbb{R}^n$, then $x \in \operatorname{Null}(A)$ if and only if $x^T A x = 0$.

Proof. If $x \in \operatorname{Null}(A)$, it is clear that $x^T A x = 0$. Suppose now that $x^T A x = 0$. Then,

$$0 = x^T A x = \|A^{1/2} x\|^2 \implies A^{1/2} x = 0 \implies Ax = 0 \implies x \in \operatorname{Null}(A).$$

□

Theorem 2.13. Let $X \in \mathbb{S}^n$. Then the following are equivalent:

(i) $X \succeq 0$;

(ii) $\lambda^{\downarrow}(X) \geq 0$;

(iii) There are $m \in \mathbb{N}$, a vector $\mu \in \mathbb{R}^m_+$, and $\{h_1, \ldots, h_m\} \subseteq \mathbb{R}^n$ such that $X = \sum_{i=1}^m \mu_i h_i h_i^T$;

(iv) There are $m \in \mathbb{N}$ and $B \in \mathbb{R}^{n \times m}$ such that $X = BB^T$;

(v) For each $S \in \mathbb{S}^n_+$, it holds that $\operatorname{Tr}(XS) \geq 0$.

Proof. [(i) $\Rightarrow$ (ii)]: Since $X \succeq 0$, we have $h^T X h \geq 0$ for every $h \in \mathbb{R}^n$. Hence, by Corollary 2.7, we have that $\lambda_{\min}(X) \geq 0$.

[(ii) $\Rightarrow$ (iii)]: Follows immediately from Theorem 2.5.

[(iii) $\Rightarrow$ (iv)]: Define $H \in \mathbb{R}^{n \times m}$ such that $He_i \coloneqq \mu_i^{1/2} h_i$ for every $i \in [m]$. Then,

$$\begin{aligned} X = \sum_{i=1}^m \mu_i h_i h_i^T &= \sum_{i=1}^m He_i \big(He_i\big)^T \\ &= H\Big(\sum_{i=1}^m e_i e_i^T\Big)H^T = HH^T. \end{aligned}$$

[(iv) $\Rightarrow$ (i)]: Note that for every $h \in \mathbb{R}^n$,

$$h^T X h = h^T B B^T h = \|B^T h\|^2 \geq 0.$$

At this point, we know that properties (i)–(iv) are equivalent.

[(iii) $\Rightarrow$ (v)]: For every $S \in \mathbb{S}^n_+$,

$$\operatorname{Tr}(XS) = \operatorname{Tr}\Big(\sum_{i=1}^m \mu_i h_i h_i^T S\Big) = \sum_{i=1}^m \mu_i \operatorname{Tr}(h_i h_i^T S) = \sum_{i=1}^m \mu_i \operatorname{Tr}(h_i^T S h_i) = \sum_{i=1}^m \mu_i h_i^T S h_i \geq 0.$$

[(v) $\Rightarrow$ (i)]: Let $y \in \mathbb{R}^n$. Since we already showed that properties (i) and (iv) are equivalent, then yy^T is positive semidefinite. Hence,

$$0 \leq \operatorname{Tr}(Xyy^T) = \operatorname{Tr}(y^T X y) = y^T X y.$$

□

Lemma 2.14. Let $M \in \mathbb{R}^{m \times n}$ be a block matrix such that

$$M = \begin{pmatrix} A & B \\ C & D \end{pmatrix},$$

where A, B, C and D are matrices of appropriate size, and A is invertible. Then, $\det(M) = \det(A)\det(D - CA^{-1}B)$.

Proof. Note that

$$M = \begin{pmatrix} I & 0 \\ CA^{-1} & I \end{pmatrix} \begin{pmatrix} A & 0 \\ 0 & D - CA^{-1}B \end{pmatrix} \begin{pmatrix} I & A^{-1}B \\ 0 & I \end{pmatrix}.$$

Taking the determinant on both sides of the equation yields $\det(M) = \det(A)\det(D - CA^{-1}B)$. □

Lemma 2.15 (Schur Complement Lemma)**.** Let $X \in \mathbb{S}^m$, let $U \in \mathbb{R}^{m \times n}$ and let $T \in \mathbb{S}^n_{++}$. Then

$$M \coloneqq \begin{pmatrix} T & U^T \\ U & X \end{pmatrix} \succeq 0 \iff X \succeq UT^{-1}U^T.$$

Moreover, we have that $M \succ 0$ if and only if $X \succ UT^{-1}U^T$.

Proof. Note that

$$\begin{pmatrix} I & 0 \\ UT^{-1} & I \end{pmatrix} \begin{pmatrix} T & 0 \\ 0 & X - UT^{-1}U^T \end{pmatrix} \begin{pmatrix} I & T^{-1}U^T \\ 0 & I \end{pmatrix} = M. \tag{2.16}$$

Let $k \coloneqq m + n$, let $L \in \mathbb{R}^{k \times k}$ be the lower triangular matrix on the left of (2.16) and let $D \in \mathbb{S}^k$ be the block diagonal matrix on the middle of (2.16). Note that

$$L^{-1} = \begin{pmatrix} I & 0 \\ -UT^{-1} & I \end{pmatrix}.$$

Hence, by Proposition 2.10, we have that $M = LDL^T \succeq 0$ if and only if $D \succeq 0$. Since $T \succ 0$, we can conclude that $D \succeq 0$ if and only if $X - UT^{-1}U^T \succeq 0$. In particular, we have that $D \succ 0$ if and only if $X - UT^{-1}U^T \succ 0$. □

Proposition 2.16. Let $A \in \mathbb{S}^n$ be a matrix, and let $r \coloneqq \operatorname{rank}(A)$. Then, there are $S \subseteq [n]$ with $|S| = r$, and $R \in \mathbb{R}^{S \times \overline{S}}$ such that $\operatorname{rank}(A[S]) = r$, and

$$A = \begin{pmatrix} I \\ R^T \end{pmatrix} A[S] \begin{pmatrix} I & R \end{pmatrix}.$$

Proof. Since $\operatorname{rank}(A) = r$, there is $S \subseteq [n]$ with $|S| = r$ such that $\{\, Ae_i : i \in S\}$ is linearly independent, and S is a maximal set with such property. Hence, $A[[n], S]$ has full rank, and for every $j \in \overline{S}$, we have that $\{\, Ae_i : i \in S\} \cup \{Ae_j\}$ is linearly dependent. Hence, there is $R \in \mathbb{R}^{S \times \overline{S}}$ such that, for each $j \in \overline{S}$,

$$A[[n], S]Re_j = Ae_j.$$

Hence, we have $A[[n], S]R = A[[n], \overline{S}],$. In particular, $A[S]R = A[S, \overline{S}]$ and $A[\overline{S}, S]R = A[\overline{S}]$. Since A is symmetric, we have that

$$A[\overline{S}, S] = A[S, \overline{S}]^T = R^T A[S]^T = R^T A[S].$$

Hence,

$$A[\overline{S}] = A[\overline{S}, S]R = R^T A[S]R.$$

Therefore,

$$A = \begin{pmatrix} A[S] & A[S, \overline{S}] \\ A[\overline{S}, S] & A[\overline{S}] \end{pmatrix} = \begin{pmatrix} A[S] & A[S]R \\ R^T A[S] & R^T A[S]R \end{pmatrix} = \begin{pmatrix} I \\ R^T \end{pmatrix} A[S] \begin{pmatrix} I & R \end{pmatrix}.$$ □

Theorem 2.17. If $X \in \mathbb{S}^n$, then

(i) $X \succ 0$ if and only if $\det(X[\{1, \dots, i\}]) > 0$ for each $i \in [n]$;

(ii) $X \succeq 0$ if and only if $\det(X[S]) \geq 0$ for each $S \subseteq [n]$.

Proof. Let $X \in \mathbb{S}^n$. Let us first show that

If $X \succeq 0$, then $X[S] \succeq 0$ for every $S \subseteq [n]$. In particular, if $X \succ 0$, then $X[S] \succ 0$ for every $S \subseteq [n]$. (2.17)

Suppose $X \succ 0$, let $y \in \mathbb{R}^S \setminus \{0\}$, and define $z \in \mathbb{R}^n$ by

$$z_i \coloneqq [i \in S]y_i, \qquad \forall i \in [n].$$

Hence,

$$y^T X[S]y = z^T Xz \geq 0,$$

where the above inequality is strict if $X \succ 0$. This ends the proof of (2.17). Let us now prove (i).

Suppose $X \succ 0$, and let $k \in [n]$. By (2.17), we know that $X[\{1, \dots, k\}] \succ 0$. Then, by Theorem 2.13, we have $\lambda^{\downarrow}(X[\{1, \dots, k\}]) > 0$. Hence, by Corollary 2.6 we have $\det(X[\{1, \dots, k\}]) = \prod_{i=1}^{n} \lambda_i^{\downarrow}(X[\{1, \dots, k\}]) > 0$. Suppose now that $\det(X[\{1, \dots, k\}]) > 0$ for each $k \in [n]$. If $n = 1$, the statement is trivial. Hence, suppose $n > 1$. Define $A \coloneqq X[\{1, \ldots, n-1\}]$, let $u \in \mathbb{R}^{n-1}$ be the restriction of Xe_i to $[n-1]$, and define $\alpha \coloneqq X_{nn}$. Then,

$$X = \begin{pmatrix} A & u \\ u^T & \alpha \end{pmatrix}.$$

Since $\det(A) = \det(X[\{1, \ldots, n-1\}) > 0$, we have that A is non-singular. By Lemma 2.14, we know that $\det(X) = \det(A)(\alpha - u^T A^{-1} u)$. Since $\det(X)$ and $\det(A)$ are positive, we have that $\alpha - u^T A^{-1} u > 0$. By Lemma 2.15, we have that $X \succ 0$ if and only if $\alpha - u^T A^{-1} u > 0$. Hence, $X \succ 0$. This ends the proof of (i). Let us now prove (ii).

Suppose that $X \succeq 0$ and let $S \subseteq [n]$. By Theorem 2.13, we have $\lambda^{\downarrow}(X[S]) \geq 0$. Hence, by Corollary 2.6 we have $\det(X[S]) = \prod_{i=1}^{n} \lambda_i^{\downarrow}(X[S]) \geq 0$. Suppose now that $\det(X[S]) \geq 0$ for each $S \subseteq [n]$. By Proposition 2.16, there are $S^* \subseteq [n]$ and $R \in \mathbb{R}^{S \times \overline{S}}$ such that $X[S^*]$ has full rank, and

$$X = \begin{pmatrix} I \\ R^T \end{pmatrix} X[S^*] \begin{pmatrix} I & R \end{pmatrix}.$$

Hence, if $X[S^*] \succ 0$, by Proposition 2.10 we conclude that $X \succeq 0$. Hence, it suffices to show that

$$\text{if } A \in \mathbb{S}^k \text{ is such that } \det(A[S]) \geq 0 \text{ for each } S \subseteq [k], \text{ and } A \text{ has full rank, then } A \succ 0. \tag{2.18}$$

Let $A \in \mathbb{S}^k$ be as in the above claim. Let us prove (2.18) by induction on k. If $k = 1$, then $A \in \mathbb{R}$. Hence, since $\det(A) \geq 0$, and since A has full rank, we conclude that $A > 0$. Suppose now that $k > 1$. Define

$$\begin{pmatrix} B & x \\ x^T & \alpha \end{pmatrix} \coloneqq X[S^*], \tag{2.19}$$

where $B \in \mathbb{S}^{k-1}$, $x \in \mathbb{R}^{k-1}$, and $\alpha \in \mathbb{R}$. Since A has full rank and $\det(A) \geq 0$, we know that

$$\det(A) > 0. \tag{2.20}$$

Let us show that

$$\alpha > 0. \tag{2.21}$$

First of all, note that $0 \leq \det(A[\{k\}]) = A_{k,k} = \alpha$. Moreover, for each $i \in [k-1]$, we have

$$0 \leq \det(A[\{i,k\}]) = \det \begin{pmatrix} B_{i,i} & x_i \\ x_i & \alpha \end{pmatrix} = \alpha B_{i,i} - x_i^2 \implies x_i^2 \leq \alpha B_{i,i}.$$

Hence, if $\alpha = 0$, then $x = 0$, a contradiction since A has full rank. This ends the proof of (2.21). Hence, by Lemma 2.14,

$$0 \overset{(2.20)}{<} \det(A) = \alpha \det(B - \tfrac{1}{\alpha} x x^T).$$

This together with (2.21) imply that $\det(B - \frac{1}{\alpha} x x^T)$ is positive. Hence, $B - \frac{1}{\alpha} x x^T$ has full rank. Moreover, by Lemma 2.14, for each $J \subseteq [k-1]$,

$$\det((B - \tfrac{1}{\alpha} x x^T)[J]) = \frac{1}{\alpha} \det(A[J \cup \{k\}]) \geq 0.$$

Hence, by the induction hypothesis, $B - \frac{1}{\alpha} x x^T \succ 0$, and by Lemma 2.15 we conclude that $X[S^*] \succ 0$, ending the proof of (2.18). □

One consequence of the above theorem is that

$$\text{if a matrix } A \in \mathbb{S}_+^n \text{ is such that } A_{i,i} = 0 \text{ for some } i \in [n], \text{ then } Ae_i = (e_i^T A)^T = 0. \tag{2.22}$$

To see that, suppose there is $i \in [n]$ such that $A_{i,i} = 0$ and let $j \in [n] \setminus \{i\}$. Then

$$0 \leq \det(A[\{i,j\}]) = \det \begin{pmatrix} 0 & A_{i,j} \\ A_{j,i} & A_{j,j} \end{pmatrix} = -A_{i,j}^2 \implies A_{i,j} = 0.$$

We may use the above remark without referencing it.

Theorem 2.18. Let $A \in \mathbb{S}^n_+$. Define

$$L_A := \begin{cases} 1 & \text{if } n = 1, \\ \begin{pmatrix} [\alpha > 0] & 0^T \\ [\alpha > 0]\frac{1}{\alpha}u & L_{\tilde{B}} \end{pmatrix} & \text{if } A = \begin{pmatrix} \alpha & u^T \\ u & B \end{pmatrix} \text{ and } \tilde{B} := B - [\alpha > 0]\frac{1}{\alpha}uu^T. \end{cases}$$

and define

$$D_A := \begin{cases} A_{1,1} & \text{if } A \text{ is } 1 \times 1, \\ \begin{pmatrix} \alpha & 0^T \\ 0 & D_{\tilde{B}} \end{pmatrix} & \text{if } A = \begin{pmatrix} \alpha & u^T \\ u & B \end{pmatrix} \text{ and } \tilde{B} := B - [\alpha > 0]\frac{1}{\alpha}uu^T. \end{cases}$$

Then $A = L_A D_A L_A^T$.

Proof. Let us prove the above statement by induction on n. If $n = 1$, we have that $L_A D_A L_A^T = A_{1,1} = A$. Suppose that $n > 1$ and that

$$A = \begin{pmatrix} \alpha & u^T \\ u & B \end{pmatrix},$$

where $u \in \mathbb{R}^{[n]\setminus\{1\}}$ and $B \in \mathbb{S}^{[n]\setminus\{1\}}$. Define $\tilde{B} := B - [\alpha > 0]\frac{1}{\alpha}uu^T$. We have

$$\begin{aligned} L_A D_A L_A^T &= \begin{pmatrix} [\alpha > 0] & 0^T \\ [\alpha > 0]\frac{1}{\alpha}u & L_{\tilde{B}} \end{pmatrix} \begin{pmatrix} \alpha & 0^T \\ 0 & D_{\tilde{B}} \end{pmatrix} \begin{pmatrix} [\alpha > 0] & [\alpha > 0]\frac{1}{\alpha}u^T \\ 0 & L_{\tilde{B}} \end{pmatrix} \\ &= \begin{pmatrix} [\alpha > 0]\alpha & 0^T \\ [\alpha > 0]u & L_{\tilde{B}} D_{\tilde{B}} \end{pmatrix} \begin{pmatrix} [\alpha > 0] & [\alpha > 0]\frac{1}{\alpha}u^T \\ 0 & L_{\tilde{B}} \end{pmatrix} \\ &= \begin{pmatrix} [\alpha > 0]\alpha & [\alpha > 0]u^T \\ [\alpha > 0]u & [\alpha > 0]\frac{1}{\alpha}uu^T + L_{\tilde{B}} D_{\tilde{B}} L_{\tilde{B}}^T \end{pmatrix}. \end{aligned}$$

By the induction hypothesis, we have that $L_{\tilde{B}} D_{\tilde{B}} L_{\tilde{B}}^T = \tilde{B}$. Thus,

$$[\alpha > 0]\tfrac{1}{\alpha}uu^T + L_{\tilde{B}} D_{\tilde{B}} L_{\tilde{B}}^T = B.$$

Moreover, since $A \succeq 0$ we have that if $\alpha = 0$, then $u = 0$. Hence $u = [\alpha > 0]u$. Therefore, the result follows. ☐

Corollary 2.19 (Cholesky Decomposition)**.** If $A \in \mathbb{S}^n_+$, then there is $\tilde{L} \in \mathbb{R}^{n \times n}$ lower triangular such that $A = \tilde{L}\tilde{L}^T$.

Proof. By Theorem 2.18 there are $D \in \mathbb{S}^n_+$ diagonal and $L \in \mathbb{R}^{n \times n}$ lower triangular such that $A = LDL^T$. Note that if $\tilde{L} := LD^{1/2}$, then $A = \tilde{L}\tilde{L}^T$. Moreover, since L is lower triangular and $D^{1/2}$ is diagonal, we have that $\tilde{L}$ is lower triangular, and the theorem follows. ☐

2.4 Moore-Penrose Pseudoinverse

A (Moore-Penrose) **pseudoinverse** of $A \in \mathbb{R}^{m \times n}$ is a matrix $A^\dagger \in \mathbb{R}^{n \times m}$ such that:

(i) $AA^\dagger A = A$,

(ii) $A^\dagger A A^\dagger = A^\dagger$,

(iii) $AA^\dagger$ and $A^\dagger A$ are symmetric.

Note that if A is non-singular, then A^{-1} is a pseudoinverse of A. Moreover, one may verify that $(A^T)^\dagger = (A^\dagger)^T$. Hence, we may write $A^{T\dagger}$ instead of $(A^T)^\dagger$ without worrying about the order of the symbols.

Proposition 2.20. Every matrix $A \in \mathbb{R}^{m \times n}$ has at most one pseudoinverse.

Proof. Let $B, C \in \mathbb{R}^{n\times m}$ be pseudoinverses of A. Then

$$AB = (AB)^T = B^T A^T = B^T (ACA)^T = B^T A^T C^T A^T = (AB)^T (AC)^T = ABAC = AC.$$

Analogously, $BA = CA$. Therefore,

$$B = BAB = CAB = CAC = C. \qquad \square$$

Proposition 2.21. Let $A \in \mathbb{R}^{m\times n}$ and let $B \in \mathbb{R}^{n\times m}$. Then

$$(AB)^\dagger = B^\dagger A^\dagger \tag{2.23}$$

if at least one of the following holds:

(1) $A^T A = I$;

(2) $BB^T = I$;

(3) $B = A^T$.

In particular, we have that $(AA^T)^\dagger = A^{T\dagger} A^\dagger$.

Proof. If (1) holds, then one can verify that $A^\dagger = A^T$. Let us show that, in this case, (2.23) holds. Using the properties of the pseudoinverse of B, we have

$$(AB)(B^\dagger A^\dagger)(AB) = ABB^\dagger A^T AB = ABB^\dagger B \overset{\text{(i)}}{=} AB$$

and

$$(B^\dagger A^\dagger)(AB)(B^\dagger A^\dagger) = B^\dagger A^T ABB^\dagger A^\dagger = B^\dagger BB^\dagger A^\dagger \overset{\text{(ii)}}{=} B^\dagger A^\dagger.$$

Let us now show that $ABB^\dagger A^\dagger$ is symmetric. We have

$$(ABB^\dagger A^\dagger)^T = (ABB^\dagger A^T)^T = A(BB^\dagger)^T A^T \overset{\text{(iii)}}{=} ABB^\dagger A^T = ABB^\dagger A^\dagger.$$

Analogously, $B^\dagger A^\dagger AB$ is symmetric. Hence, (2.23) holds in this case. The proof is analogous in the case on which (2) holds. Let us analyze now the case which (3) holds. In this case, we have

$$(AA^T)(A^{T\dagger} A^\dagger)(AA^T) = A(A^\dagger A)^T A^\dagger AA^T \overset{\text{(iii)}}{=} AA^\dagger AA^\dagger AA^T \overset{\text{(i)}}{=} AA^\dagger AA^T \overset{\text{(i)}}{=} AA^T.$$

With a similar proof, one can verify that $(A^{T\dagger} A^\dagger)(AA^T)(A^{T\dagger} A^\dagger) = A^{T\dagger} A^\dagger$. Let us show that $AA^T A^{T\dagger} A^\dagger$ is symmetric. We have

$$(AA^T)(A^{T\dagger} A^\dagger) = A(A^\dagger A)^T A^\dagger \overset{\text{(iii)}}{=} AA^\dagger AA^\dagger \overset{\text{(i)}}{=} AA^\dagger,$$

and since $AA^\dagger$ is symmetric by the property (iii) of the pseudoinverse of A, we conclude that $AA^T A^{T\dagger} A^\dagger$ is symmetric. Analogously, $A^{T\dagger} A^\dagger AA^T$ is symmetric. Therefore, (2.23) holds in this case. $\square$

Theorem 2.22. If $A \in \mathbb{S}^n$, then there is an unique pseudoinverse $A^\dagger$ of A and it is given by

$$A^\dagger = Q \operatorname{Diag}(\mu) Q^T, \tag{2.24}$$

where $Q \in \mathbb{R}^{n\times n}$ is an orthogonal matrix, and $\mu \in \mathbb{R}^n$ is defined by

$$\mu_i := [\lambda_i^\downarrow(A) \neq 0] \lambda_i^\downarrow(A)^{-1}, \qquad \forall i \in [n].$$

Proof. Uniqueness follows from Proposition 2.20. Hence, it only remains to show that $A^\dagger$ as defined in (2.24) satisfies the properties of a pseudoinverse of A.

Suppose that $Q = I$. In this case, properties (i) and (ii) are easily verified using the fact that, for any $x, y \in \mathbb{R}^n$, $\operatorname{Diag}(x)\operatorname{Diag}(y) = \operatorname{Diag}(x \odot y)$. To show that property (iii) holds, it suffices to use the fact that the product of diagonal matrices is also diagonal. Therefore,

$$\operatorname{Diag}(\lambda^\downarrow(A))^\dagger = \operatorname{Diag}(\mu).$$

For the general case where Q is any orthogonal matrix, using Proposition 2.21 we have that

$$(Q \operatorname{Diag}(\lambda^\downarrow(A)) Q^T)^\dagger = (\operatorname{Diag}(\lambda^\downarrow(A)) Q^T)^\dagger Q^\dagger = (Q^T)^\dagger \operatorname{Diag}(\lambda^\downarrow(A))^\dagger Q^T = Q \operatorname{Diag}(\mu) Q^T. \qquad \square$$

With the above theorem, one may note that if $A \in \mathbb{S}^n_+$, then $(A^{1/2})^\dagger = (A^\dagger)^{1/2}$. Hence, we may write $A^{\dagger 1/2}$ without worrying about the order of the operators.

Proposition 2.23. If $A \in \mathbb{R}^{m\times n}$, then $AA^\dagger = \operatorname{Proj}_{\operatorname{Im}(A)}$ and $A^\dagger A = \operatorname{Proj}_{\operatorname{Im}(A^T)}$.

Proof. Let $P \coloneqq AA^\dagger$. By property (iii) of a pseudoinverse, $P^T = P$. Moreover, by property (ii),

$$P^2 = AA^\dagger AA^\dagger = AA^\dagger = P.$$

Hence, the matrix P is an orthogonal projector. Clearly $\operatorname{Im}(P) \subseteq \operatorname{Im}(A)$. Let us show that $\operatorname{Im}(A) \subseteq \operatorname{Im}(P)$.

Let $v \in \operatorname{Im}(A)$. By definition, there is $x \in \mathbb{R}^n$ such that $v = Ax$. Then,

$$Pv = AA^\dagger v = AA^\dagger Ax = Ax = v.$$

Hence, we have that $v \in \operatorname{Im}(P)$. Therefore, we conclude that $\operatorname{Im}(P) = \operatorname{Im}(A)$. Thus, the matrix P is the orthogonal projector onto $\operatorname{Im}(A)$. Let us now prove that $A^\dagger A$ is the orthogonal projector onto $\operatorname{Im}(A^T)$.

By property (iii) of the definition of a pseudoinverse, we know that $A^\dagger A$ is symmetric. Hence,

$$A^\dagger A = (A^\dagger A)^T = A^T(A^T)^\dagger.$$

Since we already proved that $A^T(A^T)^\dagger$ is the orthogonal projector onto $\operatorname{Im}(A^T)$, the result follows. □

Theorem 2.22 gives a way to compute the pseudoinverse of symmetric matrices. The next proposition shows how to reduce the computation the pseudoinverse of an arbitrary matrix to the symmetric case.

Proposition 2.24. If $A \in \mathbb{R}^{m\times n}$, then $A^\dagger = A^T(AA^T)^\dagger$.

Proof. Define $B \coloneqq A^T(AA^T)^\dagger$. To prove that $A^\dagger = B$, it suffices to show that B satisfies the properties of the pseudoinverse of A. To see that property (i) holds, note that

$$ABA = AA^T(AA^T)^\dagger A = \operatorname{Proj}_{\operatorname{Im}(AA^T)} A.$$

To prove that $\operatorname{Proj}_{\operatorname{Im}(AA^T)} A = A$, it suffices to show that

$$\operatorname{Im}(AA^T) = \operatorname{Im}(A). \tag{2.25}$$

Clearly, $\operatorname{Im}(AA^T) \subseteq \operatorname{Im}(A)$. Let $x \in \operatorname{Im}(A)$. Hence, there is $y \in \operatorname{Null}(A)^\perp$ such that $x = Ay$. By Theorem 2.1, we have $\operatorname{Null}(A)^\perp = \operatorname{Im}(A^T)$. Hence, there is $y' \in \mathbb{R}^m$ such that $y = A^Ty'$. Hence, $x = AA^Ty'$, and we conclude that $x \in \operatorname{Im}(AA^T)$. This ends the proof of (2.25). Let us now prove that property (ii) holds for B. Note that,

$$BAB = A^T(AA^T)^\dagger AA^T(AA^T)^\dagger \overset{\text{(ii)}}{=} A^T(AA^T)^\dagger = B.$$

This ends the proof of (ii) for B. It only remains to prove that (iii) holds for B. We have,

$$(BA)^T = (A^T(AA^T)^\dagger A)^T = A^T((AA^T)^\dagger)^T A = A^T((AA^T)^T)^\dagger A = A^T(AA^T)^\dagger A = BA.$$

Moreover, since $AB = AA^T(AA^T)^\dagger$, we have that AB is symmetric by property (iii) of the pseudoinverse of AA^T. This ends the proof of (iii) for B. □

Proposition 2.25. If $A \in \mathbb{R}^{m\times n}$, then $\operatorname{Null}(A^\dagger) = \operatorname{Null}(A^T)$ and $\operatorname{Im}(A^\dagger) = \operatorname{Im}(A^T)$.

Proof. Let $Q \coloneqq A^\dagger A$. By Proposition 2.23, the matrix Q is an orthogonal projector onto $\operatorname{Im}(A^T)$. Since $\operatorname{Im}(Q) = \operatorname{Im}(A^T)$, to prove $\operatorname{Im}(A^\dagger) = \operatorname{Im}(A^T)$ it suffices to show $\operatorname{Im}(A^\dagger) = \operatorname{Im}(Q)$.

By definition of Q, we have $\operatorname{Im}(Q) \subseteq \operatorname{Im}(A^\dagger)$. Moreover, we have

$$QA^\dagger = A^\dagger AA^\dagger = A^\dagger \implies \operatorname{Im}(A^\dagger) \subseteq \operatorname{Im}(Q).$$

Therefore, we conclude that $\operatorname{Im}(A^\dagger) = \operatorname{Im}(Q) = \operatorname{Im}(A^T)$. This, together with Theorem 2.1, implies

$$(\operatorname{Null}(A^\dagger))^\perp = \operatorname{Im}(A^{T\dagger}) = \operatorname{Im}(A) = (\operatorname{Null}(A^T))^\perp \implies \operatorname{Null}(A^\dagger) = \operatorname{Null}(A^T).$$ □

Theorem 2.26. Let $A, B \in \mathbb{S}^n_+$ be such that $A \succeq B$ and $\operatorname{rank}(A) = \operatorname{rank}(B)$. Then $B^\dagger \succeq A^\dagger$.

Proof. Let us first show that

$$\operatorname{Im}(A) = \operatorname{Im}(B). \tag{2.26}$$

To show (2.26), let us show that

$$\operatorname{Null}(A) \subseteq \operatorname{Null}(B). \tag{2.27}$$

Note that if there is $x \in \operatorname{Null}(A) \setminus \operatorname{Null}(B)$, by Proposition 2.12 we have $x^T Bx > 0$. Hence,

$$x^T Bx > 0 = x^T Ax,$$

what is a contradiction, since $A \succeq B$. This ends the proof of (2.27). Hence, by Theorem 2.1, we have $\operatorname{Im}(B) \subseteq \operatorname{Im}(A)$. This fact together with $\operatorname{rank}(A) = \operatorname{rank}(B)$ imply (2.26).

By Proposition 2.11, $\operatorname{Im}(B) = \operatorname{Im}(B^{1/2})$. Thus, by Proposition 2.23 and by the uniqueness of the orthogonal projector, we have $B^{\dagger 1/2} B^{1/2} = \operatorname{Proj}_{\operatorname{Im}(B)} = B^{1/2} B^{\dagger 1/2}$. Hence,

$$\operatorname{Proj}_{\operatorname{Im}(B)} = \operatorname{Proj}^2_{\operatorname{Im}(B)} = B^{\dagger 1/2} B^{1/2} B^{1/2} B^{\dagger 1/2} = B^{\dagger 1/2} B B^{\dagger 1/2} \preceq B^{\dagger 1/2} A B^{\dagger 1/2}. \tag{2.28}$$

Let $r := \operatorname{rank}(A) = \operatorname{rank}(B)$. Then, (2.28) implies that $1 \leq \lambda^{\downarrow}_r (B^{\dagger 1/2} A B^{\dagger 1/2})$. By Lemma 2.4, the eigenvalues of $B^{\dagger 1/2} A B^{\dagger 1/2} = (B^{\dagger 1/2} A^{1/2})(A^{1/2} B^{\dagger 1/2})$ are the same as the ones of $(A^{1/2} B^{\dagger 1/2})(B^{\dagger 1/2} A^{1/2}) = A^{1/2} B^\dagger A^{1/2}$. Therefore,

$$1 \leq \lambda^{\downarrow}_r (A^{1/2} B^\dagger A^{1/2}), \tag{2.29}$$

thus,

$$\operatorname{Proj}_{\operatorname{Im}(A^{1/2} B^\dagger A^{1/2})} \preceq A^{1/2} B^\dagger A^{1/2}. \tag{2.30}$$

Let us show that

$$\operatorname{Im}(A^{1/2} B^\dagger A^{1/2}) = \operatorname{Im}(A). \tag{2.31}$$

By Proposition 2.11,

$$\operatorname{Im}(A) = \operatorname{Im}(A^{1/2}). \tag{2.32}$$

Hence, $\operatorname{Im}(A^{1/2} B^\dagger A^{1/2}) \subseteq \operatorname{Im}(A)$. Moreover, by Proposition 2.25,

$$\operatorname{Im}(B^\dagger) = \operatorname{Im}(B) \overset{(2.26)}{=} \operatorname{Im}(A) = \operatorname{Im}(A^\dagger). \tag{2.33}$$

Let $x \in \operatorname{Im}(A)$. By (2.32), there is $y \in \operatorname{Im}(A) = \operatorname{Im}(A^{1/2})$ such that $x = A^{1/2} y$. By (2.33), there is $y' \in \operatorname{Im}(A^{1/2}) = \operatorname{Im}(B^\dagger)$ such that $y = B^\dagger y'$. Again by (2.33), there is $y'' \in \operatorname{Im}(B^\dagger) = \operatorname{Im}(A^{1/2})$ such that $y' = A^{1/2} y''$. Hence, $x = A^{1/2} B^\dagger A^{1/2} y''$. We conclude that $\operatorname{Im}(A) \subseteq \operatorname{Im}(B^{1/2} A^\dagger B^{1/2})$. This ends the proof of (2.31). Hence, $\operatorname{Proj}_{\operatorname{Im}(A)} = \operatorname{Proj}_{\operatorname{Im}(B^{1/2} A^\dagger B^{1/2})}$. This fact together with (2.30) implies that

$$\operatorname{Proj}_{\operatorname{Im}(A)} \preceq A^{1/2} B^\dagger A^{1/2}. \tag{2.34}$$

By Proposition 2.23, $A^{\dagger 1/2} A^{1/2} = \operatorname{Proj}_{\operatorname{Im}(A)} = A^{1/2} A^{\dagger 1/2}$. Hence, by Proposition 2.10,

$$A^\dagger = A^{\dagger 1/2} \operatorname{Proj}_{\operatorname{Im}(A)} A^{\dagger 1/2} \overset{(2.34)}{\preceq} A^{\dagger 1/2} A^{1/2} B^\dagger A^{1/2} A^{\dagger 1/2} = \operatorname{Proj}_{\operatorname{Im}(A)} B^\dagger \operatorname{Proj}_{\operatorname{Im}(A)} \overset{(2.33)}{=} B^\dagger. \qquad \square$$

Proposition 2.27. If $A \in \mathbb{R}^{m \times n}$ and $b \in \operatorname{Im}(A) \subseteq \mathbb{R}^m$, then

$$A^\dagger b = \arg\min\{\, \|x\| : x \in \mathbb{R}^n, Ax = b\}.$$

Proof. Define $x^* := A^\dagger b$, and let $y \in \mathbb{R}^n$ be such that $Ay = b$. Such y exists since $b \in \operatorname{Im}(A)$. Let us show that

$$\text{If } y \neq x^*, \text{ then } \|y\| > \|x^*\|. \tag{2.35}$$

By Proposition 2.23,

$$Ax^* = AA^\dagger b = \operatorname{Proj}_{\operatorname{Im}(A)} b = b.$$

Therefore,

$$0 = Ay - b = A(y - x^*). \tag{2.36}$$

Using the above equation with the properties of the pseudoinverse, we have

$$\begin{aligned}(x^*)^T(y - x^*) &= b^T(A^\dagger)^T(y - x^*) = b^T(A^\dagger A A^\dagger)^T(y - x^*)\\ &= b^T((A^\dagger A)^T A^\dagger)^T(y - x^*) = b^T(A^T A^{T\dagger} A^\dagger)^T(y - x^*)\\ &= b^T(A^\dagger)^T A^\dagger A(y - x^*) \overset{(2.36)}{=} 0.\end{aligned}$$

Therefore $(y - x^*) \perp x^*$. Thus,

$$\|y\|^2 = \|y - x^* + x^*\|^2 = \|x^*\|^2 + \|y - x^*\|^2 \geq \|x^*\|^2,$$

where equality holds if, and only if, $y = x^*$. This ends the proof of (2.35). □

2.5 The Spectrum of the Adjacency Matrix

The **adjacency matrix** of a graph $G = (V, E)$ is the matrix $A_G \in \mathbb{S}^V$ such that $(A_G)_{i,j} := [ij \in E]$ for every $i, j \in V$.

Theorem 2.28. If G is a graph, then

$$\|A_G\|_2 \leq \Delta(G), \tag{2.37}$$

and equality holds if and only if some component of G is regular with valency $\Delta(G)$.

Proof. First, let us show (2.37). By Corollary 2.9, we know that $\|A_G\|_2 = \max\{|\lambda_{\max}(A_G)|, |\lambda_{\min}(A_G)|\}$. Hence, it suffices to bound the maximum absolute value of the eigenvalues of A_G. Let $x \in \mathbb{R}^V$ be an eigenvector of A_G with associated eigenvalue λ. If $j \in \arg\max_{i\in V}|x_i|$, then

$$|\lambda||x_j| = |(Ax)_j| = \Big|\sum_{i\in N(j)} x_i\Big| \leq \sum_{i\in N(j)} |x_i| \leq \deg(j)|x_j| \leq \Delta(G)|x_j|. \tag{2.38}$$

Note that,

if $j \in \arg\max_{i\in V}|x_i|$, then equality holds in (2.38) if and only if all the entries of $x\restriction_{N(j)}$ have the same sign, $N(j) \subseteq \arg\max_{i\in V}|x_i|$ and $\deg(j) = \Delta(G)$. (2.39)

Hence, we have that $|\lambda| \leq \Delta(G)$. This concludes the proof of (2.37).

Suppose now that G has a component H that is regular with degree $\Delta(G)$, and set $G' := G - V(H)$. Then,

$$A_G \mathbb{1}_{V(H)} = \begin{pmatrix} A_H & 0 \\ 0 & A_{G'} \end{pmatrix}\begin{pmatrix} \mathbb{1} \\ 0 \end{pmatrix} = \begin{pmatrix} \Delta(H)\mathbb{1} \\ 0 \end{pmatrix} = \Delta(G)\mathbb{1}_{V(H)}.$$

To prove the other side of the implication, suppose that $\|A\|_2 = \Delta(G)$. Let $k \in R := \arg\max_{i\in V}|x_i|$. Let H be a component of G such that $k \in V(H)$. Suppose there exists $i \in V(H) \setminus R$, and choose such i with minimum distance to k. Clearly $i \neq k$. Let $(v_0, \ldots, v_\ell)$ be a path of minimum length in H from $v_0 = i$ to $v_\ell = k$. Such a path exists since H is connected. By our choice of i, we have that $j := v_1 \in R$. By (2.39), it follows that $i \in N(j) \subseteq R$, which is a contradiction. □

Theorem 2.29. If G is a bipartite graph and $\lambda \in \mathbb{R}$, then λ is an eigenvalue of A_G if and only if $-\lambda$ also is.

Proof. Let $\{X, Y\}$ be a bipartition of G. Then

$$A_G = \begin{pmatrix} 0 & M \\ M^T & 0 \end{pmatrix},$$

where $M \in \mathbb{R}^{X\times Y}$. Let

$$v = \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^V \setminus \{0\}$$

be such that $A_G v = \lambda v$, where $x \in \mathbb{R}^X$ and $y \in \mathbb{R}^Y$. Therefore,

$$\begin{pmatrix} \lambda x \\ \lambda y \end{pmatrix} = A_G v = \begin{pmatrix} 0 & M \\ M^T & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} My \\ M^T x \end{pmatrix}.$$

Define

$$u \coloneqq \begin{pmatrix} x \\ -y \end{pmatrix}.$$

Then,

$$A_G u = \begin{pmatrix} 0 & M \\ M^T & 0 \end{pmatrix} \begin{pmatrix} x \\ -y \end{pmatrix} = \begin{pmatrix} -My \\ M^T x \end{pmatrix} = \begin{pmatrix} -\lambda x \\ \lambda y \end{pmatrix} = -\lambda u$$

Thus, we have that $-\lambda$ is also an eigenvalue of A_G. □

2.6 Incidence Matrices

The **incidence matrix** $B_D \in \mathbb{R}^{V \times A}$ of a digraph $D = (V, A)$ is defined by

$$(B_D)_{i,a} \coloneqq \begin{cases} 1, & \text{if } a = (i, j) \text{ for some } j \in V, \\ -1, & \text{if } a = (j, i) \text{ for some } j \in V, \\ 0, & \text{otherwise,} \end{cases} \qquad \forall (i, a) \in V \times A. \tag{2.40}$$

It is easy to verify that $B_D e_{(i,j)} = e_i - e_j$ for every $(i, j) \in A$. Hence,

$$B_D = \sum_{(i,j) \in A} (e_i - e_j) e_{(i,j)}^T.$$

Proposition 2.30. If $D = (V, A)$ is a digraph, then $\mathrm{Null}(B_D^T)$ is the linear subspace spanned by

$$\{\, \mathbb{1}_{V(C)} : C \text{ is a component of } D\}. \tag{2.41}$$

Proof. If C is a component of D and $(i, j) \in A$, then $(e_i - e_j)^T \mathbb{1}_{V(C)} = 0$, so

$$B_D^T \mathbb{1}_{V(C)} = \sum_{(i,j) \in A} e_{(i,j)} (e_i - e_j)^T \mathbb{1}_{V(C)} = 0.$$

Hence, $\mathrm{span}\{\, \mathbb{1}_{V(C)} : C \text{ is a component of } G\} \subseteq \mathrm{Null}(B_D^T)$.

Let $x \in \mathrm{Null}(B_D^T)$, and let $i, j \in V$ be vertices of the same component C of G. Let $(v_0, \ldots, v_\ell)$ be a path in C from $v_0 = i$ to $v_\ell = j$. Note that

$$0 = (B_D^T x)_{(r,s)} = x_r - x_s \implies x_r = x_s \qquad \forall (r, s) \in A.$$

Thus, we have that $x(v_0) = x(v_1) = \cdots = x(v_\ell)$. Therefore, for every component of G its vertices will have the same value in x. Thus, x is in the subspace spanned by (2.41).

□

Chapter 3

The Graph Laplacian

The **Laplacian** of a graph $G = (V, E)$ is the function $\mathcal{L}_G \colon \mathbb{R}^E \to \mathbb{S}^V$ such that

$$\mathcal{L}_G(w) = \sum_{ij \in E} w_{ij} L_{ij}, \qquad \forall w \in \mathbb{R}^E,$$

where $L_{ij} := (e_i - e_j)(e_i - e_j)^T$ for $ij \in E$. Note that $L_G := \mathcal{L}_G(\mathbb{1}) = \mathrm{Diag}(\deg) - A_G$. A Laplacian system is any system of equations of the form $Lx = b$, where L is the Laplacian of a graph.

Proposition 3.1. If $\vec{G} = (V, A)$ is an orientation of a graph $G = (V, E)$ and $w \in \mathbb{R}^E$, then $\mathcal{L}_G(w) = B_{\vec{G}} \mathrm{Diag}(w) B_{\vec{G}}^T$.

Proof. We have

$$\begin{aligned} B_{\vec{G}} \mathrm{Diag}(w) B_{\vec{G}}^T &= B_{\vec{G}} \Big(\sum_{(i,j) \in A} w_{ij} e_{(i,j)} e_{(i,j)}^T \Big) B_{\vec{G}}^T \\ &= \sum_{(i,j) \in A} w_{ij} B_{\vec{G}} e_{(i,j)} \big(B_{\vec{G}} e_{(i,j)} \big)^T \\ &= \sum_{(i,j) \in A} w_{ij} (e_i - e_j)(e_i - e_j)^T \\ &= \sum_{ij \in E} w_{ij} (e_i - e_j)(e_i - e_j)^T = \mathcal{L}_G(w). \end{aligned}$$

□

Corollary 3.2. Let $G = (V, E)$ be a graph and $w \in \mathbb{R}^E_{++}$. Then $\mathcal{L}_G(w) \succeq 0$, and $\mathrm{Null}(\mathcal{L}_G(w))$ is the subspace spanned by $\{ \mathbb{1}_{V(C)} : C \text{ is a component of } G \}$.

Proof. Let $\vec{G} = (V, A)$ be an orientation of G. By Proposition 3.1 we have that $\mathcal{L}_G(w) = B_{\vec{G}} \mathrm{Diag}(w) B_{\vec{G}}^T$. Define $b_a := B_{\vec{G}} e_a$ for each $a \in A$. Then

$$\mathcal{L}_G(w) = B_{\vec{G}} \mathrm{Diag}(w) B_{\vec{G}}^T = B_{\vec{G}} \Big(\sum_{(i,j) \in A} w_{ij} e_{(i,j)} e_{(i,j)}^T \Big) B_{\vec{G}}^T = \sum_{(i,j) \in A} w_{ij} b_{(i,j)} b_{(i,j)}^T.$$

Therefore, by Theorem 2.13, we have that $\mathcal{L}_G(w) \succeq 0$. Let us now show that $\mathrm{Null}(\mathcal{L}_G(w))$ is the subspace spanned by $\{ \mathbb{1}_{V(C)} : C \text{ is a component of } G \}$.

By Proposition 2.12, for every $x \in \mathbb{R}^V$, we have $\mathcal{L}_G(w)x = 0$ if and only if $x^T \mathcal{L}_G(w) x = 0$. Since

$$x^T \mathcal{L}_G(w) x = x^T B_{\vec{G}} \mathrm{Diag}(w) B_{\vec{G}}^T x = \| \mathrm{Diag}(w)^{1/2} B_{\vec{G}}^T x \|_2^2,$$

and $\mathrm{Diag}(w)^{1/2}$ is nonsingular, then $\mathcal{L}_G(w)x = 0$ if and only if $B_{\vec{G}}^T x = 0$. The result now follows from Proposition 2.30. □

Proposition 3.3. If $G = (V, E)$ is a graph, then $\|L_G\|_2 \leq 2\Delta(G)$. In particular, $\lambda_{\max}(L_G) \leq 2\Delta(G)$.

Proof. By Corollary 2.9 and since $L_G \succeq 0$, we know that $\|L_G\|_2 = \lambda_{\max}(L_G)$. Hence, using Corollary 2.7,

$$\begin{aligned}\lambda_{\max}(L_G) &= \max_{x \in \mathbb{R}^V \setminus \{0\}} \frac{x^T L_G x}{x^T x} \leq \max_{x \in \mathbb{R}^V \setminus \{0\}} \frac{x^T \operatorname{Diag}(\deg) x}{x^T x} - \min_{x \in \mathbb{R}^V \setminus \{0\}} \frac{x^T A_G x}{x^T x} \\ &= \Delta(G) - \lambda_{\min}(A_G) \leq \Delta(G) + \|A_G\|_2.\end{aligned}$$

By Theorem 2.28, we have that $\|A_G\|_2 \leq \Delta(G)$. Hence,

$$\|L_G\| \leq \Delta(G) + \|A_G\|_2 \leq 2\Delta(G). \qquad \square$$

3.1 Flows in Graphs

Let $D = (V, A)$ be a digraph and let $s, t \in V$ be distinct. An (s,t)-**flow** in D is a function $f\colon A \to \mathbb{R}_+$ that obeys the flow-conservation constraints, that is,

$$\sum_{a \in \delta^{\text{out}}(v)} f_a - \sum_{a \in \delta^{\text{in}}(v)} f_a = 0 \qquad \forall v \in V \setminus \{s,t\},$$

and that

$$|f| \coloneqq \sum_{a \in \delta^{\text{out}}(s)} f_a - \sum_{a \in \delta^{\text{in}}(t)} f_a \geq 0.$$

The **value** of an (s,t)-flow f is $|f|$, and the vertices s and t are, respectively, the **source** and the **sink** of f. A **unit** (s,t)-flow is an (s,t)-flow of value 1. We may omit the vertices (s,t) when they are clear from context or when it is not necessary to know these vertices. A **circulation** is a flow of value 0. Note that we can also write the flow-conservation constraints as follows

$$B_D f = |f|(e_s - e_t). \tag{3.1}$$

Let $G = (V, E)$ be a graph and let $s, t \in V$ be distinct. Fix an orientation $\vec{G}$ of G. An (s,t)-**flow** in G (with respect to the orientation $\vec{G}$) is a function $f\colon E \to \mathbb{R}$ such that

$$B_{\vec{G}} f = |f|(e_s - e_t).$$

We may omit the fixed orientation of the graph which the flow is associated with when it is clear from context or when it is not important to explicitly name it. Note that, differently from the directed case, we allow edges to have negative flow in this case. Let f be an (s,t)-flow in a graph with respect to an orientation $\vec{G}$ and let $a = (u,v) \in A(\vec{G})$. Intuitively, if $f_a \geq 0$, one should interpret that the flow of a is going from u to v. Similarly, if $f_a < 0$, then the flow of a is going from v to u. To formalize this intuition, define the sets $A^+ \coloneqq \{\, (u,v) \in A : f(u,v) \geq 0 \}$ and $A^- \coloneqq \{\, (u,v) \in V \times V : (v,u) \in A \setminus A^+ \}$. Define the orientation $\vec{G}' \coloneqq (V, A^+ \cup A^-)$, and define $f' \coloneqq \operatorname{Diag}(\operatorname{sgn}(f)) f$. One can note that f' is an (s,t)-flow in the digraph $\vec{G}'$. We call such a orientation $\vec{G}'$ an **induced orientation** of G (with respect to the flow f). The reader may notice that many propositions about flows in digraphs can be extended to graphs by taking care of the signs of the flow component wise.

Proposition 3.4. Let $D = (V, A)$ be a digraph and let $f \in \mathbb{R}^A_+$. Then f is an (s,t)-flow in D if and only if there is a collection of directed circuits $\mathcal{C}$ in D, a collection of directed (s,t)-paths $\mathcal{P}$ in D with $|\mathcal{C}| + |\mathcal{P}| \leq |A|$, vectors $b \in \mathbb{R}^{\mathcal{C}}_+$ and $d \in \mathbb{R}^{\mathcal{P}}_+$ with $\|d\|_1 = |f|$ such that

$$f = \sum_{C \in \mathcal{C}} b(C) \mathbb{1}_C + \sum_{P \in \mathcal{P}} d(P) \mathbb{1}_P. \tag{3.2}$$

Moreover, let $G = (V, E)$ be a graph and let $f \in \mathbb{R}^E$. Then f is an (s,t)-flow in G if and only if there is a collection of circuits $\mathcal{C}$ in G, a collection of (s,t)-paths $\mathcal{P}$ in G with $|\mathcal{C}| + |\mathcal{P}| \leq |E|$, vectors $b \in \mathbb{R}^{\mathcal{C}}_+$ and $d \in \mathbb{R}^{\mathcal{P}}_+$ with $\|d\|_1 = |f|$ such that

$$f = \operatorname{Diag}(\operatorname{sgn}(f)) \sum_{C \in \mathcal{C}} b(C) \mathbb{1}_C + \sum_{P \in \mathcal{P}} d(P) \mathbb{1}_P.$$

Proof. Suppose that there is a collection of directed circuits $\mathcal{C}$ in D and a collection of directed (s,t)-paths $\mathcal{P}$ in D with $|\mathcal{C}| + |\mathcal{P}| \leq |A|$, and vectors $b \in \mathbb{R}_+^{\mathcal{C}}$ and $d \in \mathbb{R}_+^{\mathcal{P}}$ such that (3.2) holds. Let us show that f is an (s,t)-flow of value $\|d\|_1$. It is easy to see that if $C \in \mathcal{C}$ and $P \in \mathcal{P}$, then $\mathbb{1}_{A(C)}$ is a circulation in D and $\mathbb{1}_{A(P)}$ is an (s,t)-flow in D of value 1. Hence,

$$B_D f = \sum_{C\in\mathcal{C}} b(C) B_D \mathbb{1}_C + \sum_{P\in\mathcal{P}} d(P) B_D \mathbb{1}_P = 0 + \sum_{P\in\mathcal{P}} d(P)(e_s - e_t) = \|d\|_1 (e_s - e_t).$$

Suppose now that f is an (s,t)-flow in D of value α. We will prove the statement by induction on $|\mathrm{supp}(f)|$. If $|\mathrm{supp}(f)| = 0$, the statement holds trivially. Suppose $|\mathrm{supp}(f)| > 0$. Define $D_s := (V, \mathrm{supp}(f))$ and suppose there is a directed circuit C in D_s. Define $\beta := \min\{\, f_a : a \in A(C)\} > 0$, and set $f' := f - \beta \mathbb{1}_{A(C)}$. Note that $|\mathrm{supp}(f')| < |\mathrm{supp}(f)|$ by the choice of β. It is easy to see that $\mathbb{1}_{A(C)}$ is a circulation in D. Hence,

$$B_D f' = B_D f - \beta B_D \mathbb{1}_{A(C)} = B_D f = \alpha(e_s - e_t).$$

Therefore, f' is an (s,t)-flow in D, and the statement follows by the induction hypothesis.

Let us now analyze the case on which there are no directed circuits in D_s. Let $P = (v_0, \cdots, v_k)$ be a maximal path in D_s, where $v_i \in V$ for each $i \in [k]$. Let us prove that

$$P \text{ is an } (s,t)\text{-path.} \tag{3.3}$$

Suppose that $v_0 \notin \{s,t\}$. Since there are no circuits in D_s, and since P is maximal, we conclude that $\delta^{\mathrm{in}}(v_0) = \emptyset$ in D_s, but this would violate the flow-conservation constraints. Hence, $v_0 \in \{s,t\}$. Similarly, $v_k \in \{s,t\}$. Hence,

if P is a maximal path in D_s and D_s has no circuits, then P is either a directed (s,t)-path or a directed (t,s)-path. (3.4)

Suppose that P is a (t,s)-path. In this case, if $\delta^{\mathrm{out}}(s) = \emptyset$ in D_s, then $|f| < 0$, a contradiction. Hence, let $a \in \delta^{\mathrm{out}}(s)$ in D_s, and let P' be a maximal path in D_s such that $a \in P'$. By (3.4), we have that P' is an (s,t)-path. Since P is an (t,s)-path in D_s and P' is an (s,t)-path in D_s, this means that there is a directed circuit in D_s, a contradiction. This ends the proof of (3.3). Hence, define $\beta := \min\{\, f_a : a \in A(P)\} > 0$. Define $f' := f - \beta \mathbb{1}_{A(P)}$. Note that $|\mathrm{supp}(f')| < |\mathrm{supp}(f)|$ by the choice of β. It is easy to note that $\mathbb{1}_{A(P)}$ is an (s,t)-flow in D. Hence,

$$B_D f' = B_D f - \beta B_D \mathbb{1}_{A(P)} = (\alpha - \beta)(e_s - e_t).$$

If $\alpha \geq \beta$, then f' is an (s,t)-flow in D of value $\alpha - \beta$, and the statement follows by the induction hypothesis. Hence, it only remains to show that

$$\alpha \geq \beta. \tag{3.5}$$

Suppose that $\beta > \alpha$. Since P is an (s,t)-path, there is $a \in \delta^{\mathrm{out}}(s) \cap A(P)$. Hence, $f_a \geq \beta$ by the definition of β. Moreover, since $\beta > \alpha$, there must be $a \in \delta^{\mathrm{in}}(s)$ in D_s, otherwise $|f| = \alpha$ would be at least β. Let P' be a maximal path such that $a \in A(P')$. Since we are in the case where there are no circuits in D_s, we have that P' is a (t,s)-path by (3.4). Since P' is an (t,s)-path in D_s and P is an (s,t)-path in D_s, there must be a directed circuit in D_s, a contradiction. This ends the proof of (3.5). The undirected case follows from the directed case from the definition of induced orientation. ☐

3.2 Electrical Flows

If f is a flow in a weighted graph $G = (V, E, r)$, then the **energy** of f (with respect to r) is

$$\mathcal{E}(f) := f^T \operatorname{Diag}(r) f.$$

Let G be a weighted graph with weights $r \in \mathbb{R}_{++}^E$. An (s,t)-flow f of value $\alpha \in \mathbb{R}_+$ in G is **electrical** if f minimizes the energy over all the (s,t)-flows of value α in G. We say that r defines the **resistances** on the edges for the electrical flow f.

Proposition 3.5. Let $G = (V, E, r)$ be a weighted graph and let $f \in \mathbb{R}^E$ be an electrical (s,t)-flow in G of value $\alpha \in \mathbb{R}_{++}$. Then there is a collection of (s,t)-paths $\mathcal{P}$ in G with $|\mathcal{P}| \leq |E|$ and a vector $b \in \mathbb{R}^{\mathcal{P}}_+$ with $\|f\|_\infty \leq \|b\|_1 = \alpha$ such that

$$f = \mathrm{Diag}(\mathrm{sgn}(f)) \sum_{P \in \mathcal{P}} b(P) \mathbb{1}_{E(P)}. \tag{3.6}$$

Proof. Let $\vec{G}$ be an induced orientation of G with respect to the flow f, and define $g \coloneqq \mathrm{Diag}(\mathrm{sgn}(f))$ f. Note that, for any (s,t)-flow f' in G and any vector $u \in \{\pm 1\}^V$, we have

$$\mathcal{E}(f') = \mathcal{E}(\mathrm{Diag}(u) f').$$

Hence, it suffices to prove that if g has minimum energy among the (s,t)-flows of value α in $\vec{G}$, then there is a collection of directed (s,t)-paths $\mathcal{P}$ in $\vec{G}$ with $|\mathcal{P}| \leq |E|$ and a vector $b \in \mathbb{R}^{\mathcal{P}}_+$ with $\|b\|_1 = \alpha$ such that

$$g = \sum_{P \in \mathcal{P}} b(P) \mathbb{1}_{A(P)}. \tag{3.7}$$

By Proposition 3.4, there is a collection of directed circuits $\mathcal{C}$ in $\vec{G}$, a collection of (s,t)-paths $\mathcal{P}$ in $\vec{G}$ with $|\mathcal{C}| + |\mathcal{P}| \leq |E|$, vectors $b \in \mathbb{R}^{\mathcal{P}}$ and $d \in \mathbb{R}^{\mathcal{C}}$ with $\|b\|_1 = \alpha$ such that

$$g = \sum_{C \in \mathcal{C}} d(C) \mathbb{1}_{A(C)} + \sum_{P \in \mathcal{P}} b(P) \mathbb{1}_{A(P)}. \tag{3.8}$$

Suppose that $\mathcal{C} \neq \emptyset$, let $\overline{C} \in \mathcal{C}$ be fixed and define $g' \coloneqq g - d(\overline{C}) \mathbb{1}_{A(\overline{C})}$. Note that $d(\overline{C}) \mathbb{1}_{A(\overline{C})}$ is a circulation in $\vec{G}$. Hence,

$$B_{\vec{G}} g' = B_{\vec{G}} g - d(\overline{C}) B_{\vec{G}} \mathbb{1}_{A(\overline{C})} = B_{\vec{G}} g = \alpha.$$

Therefore, g' is an (s,t)-flow in $\vec{G}$ of value α. By (3.8), we have

$$g' = \sum_{C \in \mathcal{C} \setminus \{\overline{C}\}} d(C) \mathbb{1}_{A(C)} + \sum_{P \in \mathcal{P}} b(P) \mathbb{1}_{A(P)}.$$

Hence, $g' \geq 0$. Moreover, since $d(\overline{C}) > 0$, we have that $g'_a < g_a$ for each $a \in A(\overline{C})$. Therefore $\mathcal{E}(g') < \mathcal{E}(g)$, a contradiction. This ends the proof of equation (3.7). Let us now prove that

$$\|g\|_\infty \leq \|b\|_1 = \alpha.$$

By (3.6), for each $a \in A(\vec{G})$,

$$g_a = \sum_{P \in \mathcal{P}} [a \in \mathcal{P}] b(P) \leq \|b\|_1 = \alpha. \qquad \square$$

Theorem 3.6. Let $G = (V, E, r)$ be a connected weighted graph, let $s, t \in V$ be distinct, let $f^* \in \mathbb{R}^E$ be an electrical (s,t)-flow in G of value $\alpha \in \mathbb{R}_+$, and let $c \in \mathbb{R}^E_{++}$ be defined by $c_e \coloneqq r_e^{-1}$ for each $e \in E$. Then f^* is unique, and for every $v \in \mathcal{L}_G(c)^\dagger (e_s - e_t) + \mathrm{span}\{\mathbb{1}\}$,

$$f^* = \alpha \, \mathrm{Diag}(c) B^T_{\vec{G}} v. \tag{3.9}$$

In particular, if f_1 is the unit electric (s,t)-flow in G, then, for every $\alpha \in \mathbb{R}_+$ we have that αf_1 is the electrical (s,t)-flow of value α.

Proof. Let $\mathcal{F}_{s,t}$ be the set of all (s,t)-flows in G of value $\alpha \in \mathbb{R}_+$. Let $f \in \mathcal{F}_{s,t}$, and let $R \coloneqq \mathrm{Diag}(r)$. Note that

$$\mathcal{E}(f) = f^T R f = (R^{1/2} f)^T R^{1/2} f = \|R^{1/2} f\|_2^2.$$

Hence,

$$\begin{aligned}\arg\min\{\,\mathcal{E}(f) : f \in \mathcal{F}_{s,t}\} &= \arg\min\{\,\|R^{1/2}f\|_2^2 : f \in \mathcal{F}_{s,t}\}\\ &= \arg\min\{\,\|R^{1/2}f\|_2^2 : B_{\vec{G}}f = \alpha(e_s - e_t)\}\\ &= R^{-1/2}\arg\min\{\,\|g\|_2^2 : B_{\vec{G}}R^{-1/2}g = \alpha(e_s - e_t)\},\end{aligned}$$

where in the last equation we made the change of variables $g \coloneqq R^{1/2}f$. By Proposition 2.27, the solution of minimum norm of the system

$$B_{\vec{G}}R^{-1/2}g = \alpha(e_s - e_t)$$

is $g^* \coloneqq \alpha(B_{\vec{G}}R^{-1/2})^\dagger(e_s - e_t)$. By Proposition 2.24, we have $A^\dagger = A^T(AA^T)^\dagger$ for any $A \in \mathbb{R}^{m\times n}$. Hence,

$$\begin{aligned}(B_{\vec{G}}R^{-1/2})^\dagger = (B_{\vec{G}}R^{-1/2})^T\Big(B_{\vec{G}}R^{-1/2}(B_{\vec{G}}R^{-1/2})^T\Big)^\dagger &= R^{-1/2}B_{\vec{G}}^T\Big(B_{\vec{G}}R^{-1/2}R^{-1/2}B_{\vec{G}}^T\Big)^\dagger\\ = R^{-1/2}B_{\vec{G}}^T\big(B_{\vec{G}}\operatorname{Diag}(c)B_{\vec{G}}^T\big)^\dagger &= R^{-1/2}B_{\vec{G}}^T\mathcal{L}_G(c)^\dagger,\end{aligned}$$

where in the last equation we used Proposition 3.1. Therefore, we have $g^* = \alpha R^{-1/2}B_{\vec{G}}^T\mathcal{L}_G(w)^\dagger(e_s - e_t)$. Since we made the change of variables $g = R^{1/2}f$, we have

$$f^* = R^{-1/2}g^* = \alpha R^{-1}B_{\vec{G}}^T\mathcal{L}_G(c)^\dagger(e_s - e_t) = \alpha\operatorname{Diag}(c)B_{\vec{G}}^T\mathcal{L}_G(c)^\dagger(e_s - e_t).$$

Moreover, since G is connected, by Proposition 2.30 we have $\operatorname{Null}(B_{\vec{G}}^T) = \operatorname{span}\{\mathbb{1}\}$. Therefore, for every $\beta \in \mathbb{R}$,

$$f^* = \alpha\operatorname{Diag}(c)^{-1}B_{\vec{G}}^T(\mathcal{L}_G(c)^\dagger(e_s - e_t) + \beta\mathbb{1}).$$

Let $v \in \mathcal{L}_G(c)^\dagger(e_s - e_t) + \operatorname{span}\{\mathbb{1}\}$. In particular, if f_1 is the unit electrical (s,t)-flow in G, we have $f_1 = \operatorname{Diag}(c)B_{\vec{G}}^Tv$. Hence,

$$f^* = \alpha\operatorname{Diag}(c)B_{\vec{G}}^Tv = \alpha f_1.$$

□

Let G be a weighted graph with weights $r \in \mathbb{R}_{++}^E$, let $\vec{G}$ be a fixed orientation of G, and let f be the electrical (s,t)-flow in G with respect to the orientation $\vec{G}$. We say that a vector $v \in \mathbb{R}^V$ as in the above theorem is a vector of **vertex potentials** (with respect to the flow f and the vector r). Define $c \in \mathbb{R}^E$ such that $c_e \coloneqq 1/r_e$ for each $e \in E$. Suppose that f is a unit electrical (s,t)-flow in G, and let $v \in \mathbb{R}^V$ be vertex potentials of f. Hence, Proposition 3.1 together with equations (3.1) and (3.9) yield

$$\mathcal{L}_G(c)v = B_{\vec{G}}\operatorname{Diag}(c)B_{\vec{G}}^Tv = B_{\vec{G}}f = (e_s - e_t). \tag{3.10}$$

This means that vertex potentials can be obtained by solving a Laplacian system. Moreover, by Theorem 3.6, the electrical (s,t)-flow of value $\alpha \in \mathbb{R}_+$ in G is simply αf. Hence, it suffices to know how to compute the unit electrical flow in a graph to compute a electrical flow of an arbitrary value.

There is a physical intuition behind the language used in this section. Although it is not essential to understand this intuition for a good comprehension of the remaining of the section, the reader may find it useful as a memory aid for the definitions about electrical flows. For a moment, imagine that a weighted graph $G = (V, E, r)$ is a representation of an electrical network. Each edge $e \in E$ represents a resistor with resistance r_e and each vertex represents a node (or junction) of the circuit. The (s,t)-flow f represents the currents passing through the resistors when connecting s and t to the poles of an external current source. With that in mind, one can note that (3.9) is equivalent to *Ohm's law* [17], that states

The current through a resistor between two nodes is directly proportional to the potential difference across the two nodes, (3.11)

Moreover, the flow-conservation constraints are equivalent to *Kirchhoff's current law* [17], which states

At any node in an electrical circuit, the sum of currents flowing into that node is equal to the sum of currents flowing out of that node. (3.12)

Let $G = (V, E, r)$ be a weighted graph, let $s, t \in V$ be distinct, and let $c \in \mathbb{R}^E$ be given by $c_e := 1/r_e$ for each $e \in E$. The **effective resistance** of s and t (with respect to r) is

$$R_{\text{eff}}^{s,t}(r) = (e_s - e_t)^T \mathcal{L}_G(c)^\dagger (e_s - e_t).$$

We may omit the superscript whenever the vertices s, t are clear from context (usually they will be the source and sink of a flow).

Proposition 3.7. If G is a weighted graph with weights $r \in \mathbb{R}^E_{++}$ and f is the unit electrical (s,t)-flow in G, then

$$R_{\text{eff}}^{s,t}(r) = \mathcal{E}(f).$$

Proof. Define $c \in \mathbb{R}^E$ by $c_e := 1/r_e$ for each $e \in E$. Let $v \in \mathbb{R}^V$ be vertex potentials of f, and let $\vec{G}$ be an orientation of G for which f is a flow in G with respect to the orientation $\vec{G}$. Then, using the property (i) of the pseudoinverse of $\mathcal{L}_G(c)$,

$$\begin{aligned}
\mathcal{E}(f) &= f^T \operatorname{Diag}(r) f = v^T B_{\vec{G}} \operatorname{Diag}(r)^{-1} \operatorname{Diag}(r) \operatorname{Diag}(r)^{-1} B_{\vec{G}}^T v \\
&= v^T B_{\vec{G}} \operatorname{Diag}(c) B_{\vec{G}}^T v = v^T \mathcal{L}_G(c) v \overset{\text{(i)}}{=} v^T \mathcal{L}_G(c) \mathcal{L}_G(c)^\dagger \mathcal{L}_G(c) v \\
&= (\mathcal{L}_G(c) v)^T \mathcal{L}_G(c)^\dagger \mathcal{L}_G(c) v \overset{(3.10)}{=} (e_s - e_t)^T \mathcal{L}_G(c)^\dagger (e_s - e_t) = R_{\text{eff}}^{s,t}(r).
\end{aligned}$$

□

Proposition 3.8 (Rayleigh Monotonicity)**.** If $G = (V, E)$ is a graph, $s, t \in V$ are distinct, and $r, r' \in \mathbb{R}^E_{++}$ are such that $r \geq r'$, then

$$R_{\text{eff}}^{s,t}(r) \geq R_{\text{eff}}^{s,t}(r').$$

Proof. Define $c, c' \in \mathbb{R}^E$ by $c_e := 1/r_e$ and $c'_e := 1/r'_e$ for every $e \in E$. Note that $c' \geq c \geq 0$. Hence, $\mathcal{L}_G(c') \succeq \mathcal{L}_G(c) \succeq 0$, and by Theorem 2.26,

$$(e_s - e_t)^T \mathcal{L}_G(c)^\dagger (e_s - e_t) \geq (e_s - e_t)^T \mathcal{L}_G(c')^\dagger (e_s - e_t).$$

□

3.3 Counting Spanning Trees

Lemma 3.9 (Matrix determinant lemma)**.** If $A \in \mathbb{R}^{n \times n}$ is invertible and $u, v \in \mathbb{R}^n$, then

$$\det(A + uv^T) = \det(A)(1 + v^T A^{-1} u).$$

Proof. Let

$$R := \begin{pmatrix} A & -u \\ v^T & 1 \end{pmatrix}.$$

Since A is invertible, the matrix R has both a block LDU decomposition and a block UDL decomposition, that is,

$$\begin{pmatrix} I & 0 \\ v^T A^{-1} & 1 \end{pmatrix} \begin{pmatrix} A & 0 \\ 0 & 1 + v^T A^{-1} u \end{pmatrix} \begin{pmatrix} I & -A^{-1} u \\ 0 & 1 \end{pmatrix} = R = \begin{pmatrix} I & -u \\ 0 & 1 \end{pmatrix} \begin{pmatrix} A + uv^T & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} I & 0 \\ v^T & 1 \end{pmatrix}.$$

The result now follows by taking the determinant of both sides of the equation. □

The **adjugate** matrix $\operatorname{adj}(A)$ of A has entries defined by

$$\operatorname{adj}(A)_{i,j} := (-1)^{i+j} \det\bigl(A[\bar{j}, \bar{i}]\bigr), \qquad \forall i, j \in [n],$$

where we recall that, for each $S \subseteq V$, we have $\overline{S} := V \setminus S$, and if $S = \{i\}$, we may write $\bar{i}$ instead of $\overline{\{i\}}$. Using the Laplace expansion to calculate the determinant of a matrix, one may verify that $A \operatorname{adj}(A) = \det(A) I$. Therefore, if A is invertible we have that $\operatorname{adj}(A) = \det(A) A^{-1}$.

Lemma 3.10. If $A \in \mathbb{R}^{n \times n}$ and $u, v \in \mathbb{R}^n$, then

$$\det(A + uv^T) = \det(A) + v^T \operatorname{adj}(A) u.$$

Proof. Let $(A_k)_{k=0}^{\infty}$ be a sequence of invertible matrices as in Theorem 2.3 that converges to A. By Lemma 3.9, for each $i \in \mathbb{N}$ we have

$$\begin{aligned}\det(A_i + uv^T) &= \det(A_i)(1 + v^T A_i^{-1} u) = \det(A_i) + v^T \det(A_i) A_i^{-1} u \\ &= \det(A_i) + v^T \operatorname{adj}(A_i) u.\end{aligned}$$

Since the determinant of a matrix is a continuous function, we can take limit on both sides. Hence, the result follows. □

Let $G = (V, E, \psi)$ be a multigraph. Define $\tau(G)$ as the number of spanning trees of G. Note that if G is not connected, then $\tau(G) = 0$. Let $e \in E$. Define $G \setminus e := (V, E \setminus \{e\}, \psi')$, where ψ' is the restriction of ψ to $E \setminus \{e\}$. Let $S \subseteq V$. The multigraph $G/S := (V', E, \psi')$ is the multigraph with the subset of vertices S **contracted**, where $V' := (V \setminus S) \cup \{S\}$ and

$$\psi'(f) := \begin{cases} \psi(f) & \text{if } \psi(f) \cap S = \emptyset, \\ \{S\} & \text{if } \psi(f) \subseteq S, \\ \{k, S\} & \text{if } \psi(f) \setminus S = \{k\}, \end{cases} \qquad \forall f \in E.$$

If $e \in E(G)$, the multigraph $G/e := (G \setminus e)/\psi_G(e)$ is the multigraph with the edge e **contracted**.

Proposition 3.11. If G is a multigraph with $|E(G)| \geq 1$, and $e \in E(G)$ is not a loop, then

$$\tau(G) = \tau(G \setminus e) + \tau(G/e).$$

Although we do not give a formal proof of the above proposition, which would be unnecessarily long to formalize, we give a sketch of the proof. Let G be a multigraph, and let $e \in E$ be a non-loop edge. To see that the above proposition holds, note that the set of spanning trees of G that do not contain e is the set of spanning trees of $G \setminus e$. In a similar way, for every spanning tree T of G that contains e, the tree T/e is a spanning tree of G/e. The converse also holds, that is, for every spanning tree of G/e we can construct a spanning tree of G that contains e.

Theorem 3.12 (Matrix Tree Theorem)**.** Let $G = (V, E)$ be a multigraph with $|V| \geq 2$. If $i \in V$, then

$$\tau(G) = \det\big(L_G[\bar{i}]\big). \tag{3.13}$$

Proof. Let $G = (V, E)$ be a counterexample that minimizes $|V| + |E|$. Let $i \in V$, and let $G' := G - i$. If i is isolated, then $L_G[\bar{i}] = L_{G'}$. Since $\mathbb{1} \in \operatorname{Null}(L_{G'})$, we have that $L_{G'}$ is singular, and $\det(L_{G'}) = 0 = \tau(G)$. So suppose that i is not isolated. Let $e = ij \in E$ for some $j \in V \setminus \{i\}$. Then,

$$\det(L_G[\bar{i}]) = \det(L_{G\setminus e}[\bar{i}] + L_e[\bar{i}])) = \det(L_{G\setminus e}[\bar{i}] + e_j e_j^T).$$

By Lemma 3.10,

$$\begin{aligned}\det(L_{G\setminus e}[\bar{i}] + e_j e_j^T) &= \det(L_{G\setminus e}[\bar{i}]) + e_j^T \operatorname{adj}(L_{G\setminus e}[\bar{i}]) e_j \\ &= \det(L_{G\setminus e}[\bar{i}]) + \big(\operatorname{adj}(L_{G\setminus e}[\bar{i}])\big)_{j,j} \\ &= \det(L_{G\setminus e}[\bar{i}]) + \det\Big(L_{G\setminus e}[\overline{\{i,j\}}]\Big).\end{aligned}$$

Let $k := \psi_G(e) \in V(G/e)$. One may verify that

$$L_{G\setminus e}[\overline{\{i,j\}}] = L_{G/e}[\bar{k}].$$

Equation (3.13) holds for $G \setminus e$ and G/e since both of these multigraphs are not counterexamples. By Proposition 3.11,

$$\det\big(L_G[\bar{i}]\big) = \det\big(L_{G\setminus e}[\bar{i}]\big) + \det\big(L_{G/e}[\bar{k}]\big) = \tau(G \setminus e) + \tau(G/e) = \tau(G). \qquad \square$$

3.4 Sparse Cuts

Let $G = (V, E)$ be a graph and let $S \subseteq V$. The **volume** of S is $\mathrm{vol}(S) \coloneqq \sum_{i \in S} \deg(i)$. Moreover, define $\mathrm{vol}(G) \coloneqq \mathrm{vol}(V) = 2|E|$. The **sparsity** $\Phi(S)$ of the cut associated with S is

$$\Phi(S) \coloneqq \frac{|\delta(S)|}{\min\{\mathrm{vol}(S), \mathrm{vol}(\overline{S})\}},$$

where we consider $\Phi(S) = \infty$ if the denominator is zero. The **sparsity** of G is

$$\Phi(G) \coloneqq \min_{\emptyset \neq S \subsetneq V} \Phi(S).$$

The **relaxed sparsity** $h(S)$ of S is

$$h(S) \coloneqq \frac{|\delta(S)|}{\mathrm{vol}(S) \cdot \mathrm{vol}(\overline{S})} \cdot \mathrm{vol}(G),$$

where we consider $h(S) = \infty$ if the denominator is zero. The **relaxed sparsity** of G is

$$h(G) \coloneqq \min_{\emptyset \neq S \subsetneq V} h(S).$$

Proposition 3.13. Let $G = (V, E)$ be a graph and let $S \subseteq V$. Then $\Phi(S) \leq h(S) \leq 2\Phi(S)$. In particular, $\Phi(G) \leq h(G) \leq 2\Phi(G)$.

Proof. For each $S \subseteq V$, note that

$$\mathrm{vol}(G) \geq \max\{\mathrm{vol}(S), \mathrm{vol}(\overline{S})\} \geq \frac{\mathrm{vol}(G)}{2}$$

and that

$$\max\{\mathrm{vol}(S), \mathrm{vol}(\overline{S})\} \cdot \min\{\mathrm{vol}(S), \mathrm{vol}(\overline{S})\} = \mathrm{vol}(S) \cdot \mathrm{vol}(\overline{S}).$$

Therefore,

$$\Phi(S) = \frac{|\delta(S)|}{\min\{\mathrm{vol}(S), \mathrm{vol}(\overline{S})\}} = \frac{|\delta(S)| \max\{\mathrm{vol}(S), \mathrm{vol}(\overline{S})\}}{\mathrm{vol}(S)\,\mathrm{vol}(\overline{S})} \leq \frac{|\delta(S)|\,\mathrm{vol}(G)}{\mathrm{vol}(S)\,\mathrm{vol}(\overline{S})} = h(S),$$

and

$$h(S) = \frac{|\delta(S)|\,\mathrm{vol}(G)}{\mathrm{vol}(S)\,\mathrm{vol}(\overline{S})} \leq \frac{|\delta(S)| 2 \max\{\mathrm{vol}(S), \mathrm{vol}(\overline{S})\}}{\mathrm{vol}(S)\,\mathrm{vol}(\overline{S})} = 2\Phi(S).$$

□

Let $G = (V, E)$ be a graph. Define $\nu : E \to [0, 1]$ to be the uniform probability mass function over E, so

$$\nu(e) \coloneqq \frac{1}{|E|}, \qquad \forall e \in E.$$

Define $\mu : V \to [0, 1]$ to be a probability mass function over V such that

$$\mu(i) \coloneqq \frac{\deg(i)}{\mathrm{vol}(G)} = \frac{\deg(i)}{2|E|}, \qquad \forall i \in V.$$

Moreover, we recall that if $p : \Omega \to [0, 1]$ is a probability mass function, then the **expected value** of a random variable $X : \Omega \to \mathbb{R}$ over p is

$$\mathbb{E}_{X \sim p}(X) \coloneqq \sum_{v \in \Omega} X(v) p(v).$$

Theorem 3.14. If $G = (V, E)$ is a graph, then

$$h(G) = \min_{x \in \{0,1\}^V \setminus \{0, \mathbb{1}\}} \frac{\mathbb{E}_{ij \sim \nu}[(x_i - x_j)^2]}{\mathbb{E}_{(i,j) \sim \mu \times \mu}[(x_i - x_j)^2]}.$$

Proof. Let $S \subseteq V$. Note that $(\mathbb{1}_S(i) - \mathbb{1}_S(j))^2 = 1$ if and only if $(i,j) \in S \times V \setminus S$ or $(i,j) \in V \setminus S \times S$. Therefore,

$$\mathbb{E}_{ij\sim\nu}[(\mathbb{1}_S(i) - \mathbb{1}_S(j))^2] = \mathbb{E}_{ij\sim\nu}\big[[(i,j) \in (S \times V \setminus S) \cup (V \setminus S \times S)]\big] = \frac{|\delta(S)|}{|E|} = \nu(\delta(S))$$

and

$$\begin{aligned}\mathbb{E}_{(i,j)\sim\mu\times\mu}[(\mathbb{1}_S(i) - \mathbb{1}_S(j))^2] &= \mathbb{P}_{(i,j)\sim\mu\times\mu}[(\mathbb{1}_S(i) - \mathbb{1}_S(j))^2 = 1] \\ &= \mathbb{P}_{(i,j)\sim\mu\times\mu}[i \in S, j \in V \setminus S] + \mathbb{P}_{(i,j)\sim\mu\times\mu}[i \in V \setminus S, j \in S] \\ &= 2\mathbb{P}_{i\sim\mu}[i \in S]\mathbb{P}_{i\sim\mu}[i \in V \setminus S] = 2\mu(S)\mu(V \setminus S).\end{aligned}$$

Hence,

$$h(S) = \frac{|\delta(S)| \operatorname{vol}(G)}{\operatorname{vol}(S)\operatorname{vol}(V \setminus S)} = \frac{2|E|}{2|E|}\frac{|\delta(S)|2|E|}{\operatorname{vol}(S)\operatorname{vol}(V \setminus S)} = \frac{\nu(\delta(S))}{2\mu(S)\mu(V \setminus S)} = \frac{\mathbb{E}_{ij\sim\nu}[(\mathbb{1}_S(i) - \mathbb{1}_S(j))^2]}{\mathbb{E}_{(i,j)\sim\mu\times\mu}[(\mathbb{1}_S(i) - \mathbb{1}_S(j))^2]}.$$

Notice that $\{0,1\}^V \setminus \{0, \mathbb{1}\} = \{\, \mathbb{1}_S : \emptyset \neq S \subsetneq V\}$. Therefore,

$$h(G) = \min_{\emptyset \neq S \subsetneq V} \frac{\mathbb{E}_{ij\sim\nu}[(\mathbb{1}_S(i) - \mathbb{1}_S(j))^2]}{\mathbb{E}_{(i,j)\sim\mu\times\mu}[(\mathbb{1}_S(i) - \mathbb{1}_S(j))^2]} = \min_{x \in \{0,1\}^V \setminus \{0,\mathbb{1}\}} \frac{\mathbb{E}_{ij\sim\nu}[(x_i - x_j)^2]}{\mathbb{E}_{(i,j)\sim\mu\times\mu}[(x_i - x_j)^2]}. \qquad \square$$

Let $G = (V, E)$ be a graph. The **real sparsity** of G is

$$h_{\mathbb{R}}(G) := \inf_{x \in \mathbb{R}^V \setminus \operatorname{span}(\{\mathbb{1}\})} \frac{\mathbb{E}_{ij\sim\nu}[(x_i - x_j)^2]}{\mathbb{E}_{(i,j)\sim\mu\times\mu}[(x_i - x_j)^2]}.$$

If G has no isolated vertices, then the **normalized Laplacian** of G is the matrix

$$\widetilde{L}_G := \operatorname{Diag}(\deg_G)^{-\frac{1}{2}} L_G \operatorname{Diag}(\deg_G)^{-\frac{1}{2}}.$$

Theorem 3.15. If G is a graph, then $\lambda_2^{\uparrow}(\widetilde{L}_G) = h_{\mathbb{R}}(G)$.

Proof. Let $G = (V, E)$ and let $x \in \mathbb{R}^V$. If $x' = x + \alpha\mathbb{1}$ for some $\alpha \in \mathbb{R}$, then

$$\frac{\mathbb{E}_{ij\sim\nu}[(x_i - x_j)^2]}{\mathbb{E}_{(i,j)\sim\mu\times\mu}[(x_i - x_j)^2]} = \frac{\mathbb{E}_{ij\sim\nu}[((x_i + \alpha\mathbb{1}) - (x_j + \alpha\mathbb{1}))^2]}{\mathbb{E}_{(i,j)\sim\mu\times\mu}[((x_i + \alpha\mathbb{1}) - (x_j + \alpha\mathbb{1}))^2]} = \frac{\mathbb{E}_{ij\sim\nu}[(x'_i - x'_j)^2]}{\mathbb{E}_{(i,j)\sim\mu\times\mu}[(x'_i - x'_j)^2]}.$$

Hence, if $f(x)$ denotes the argument of the infimum in (3.4),

$$\begin{aligned}h_{\mathbb{R}}(G) &= \inf\big\{\, f(x) : x \in \mathbb{R}^V \setminus \operatorname{span}(\{\mathbb{1}\})\big\} \\ &= \inf\big\{\, f(x) : \alpha \in \mathbb{R}, y \in \mathbb{R}^V \setminus \operatorname{span}(\{\mathbb{1}\}), x = y + \alpha\mathbb{1}\big\} \\ &= \inf\big\{\, f(x) : y \in \mathbb{R}^V \setminus \operatorname{span}(\{\mathbb{1}\}), x = y - \mathbb{E}_{i\sim\mu}[y_i]\mathbb{1}\big\} \\ &= \inf\big\{\, f(x) : x \in \mathbb{R}^V \setminus \operatorname{span}(\{\mathbb{1}\}), \mathbb{E}_{i\sim\mu}[x_i] = 0\big\} \\ &= \inf\big\{\, f(x) : x \in \mathbb{R}^V \setminus \operatorname{span}(\{\mathbb{1}\}), \langle x, \deg_G\rangle = 0\big\}.\end{aligned}$$

Furthermore, for every $x \in \mathbb{R}^V$ such that $\langle x, \deg_G\rangle = 0$,

$$\begin{aligned}\mathbb{E}_{(i,j)\sim\mu\times\mu}[(x_i - x_j)^2] &= \mathbb{E}_{(i,j)\sim\mu\times\mu}[x_i^2 + x_j^2 - 2x_ix_j] \\ &= \mathbb{E}_{i\sim\mu}[x_i]^2 + \mathbb{E}_{j\sim\mu}[x_j]^2 - 2\mathbb{E}_{(i,j)\sim\mu\times\mu}[x_ix_j] \\ &= 2\mathbb{E}_{i\sim\mu}[x_i]^2 - 2\mathbb{E}_{i\sim\mu}[x_i]\mathbb{E}_{j\sim\mu}[x_j] \\ &= 2\mathbb{E}_{i\sim\mu}[x_i]^2 = \frac{2\sum_{i\in V}\deg(i)x_i^2}{\operatorname{vol}(G)} \\ &= \frac{2x^T \operatorname{Diag}(\deg_G)x}{\operatorname{vol}(G)}.\end{aligned}$$

Moreover,

$$\mathbb{E}_{ij\sim\nu}[(x_i - x_j)^2] = \frac{\sum_{ij\in E}(x_i - x_j)^2}{|E|} = \frac{2x^T L_G x}{\mathrm{vol}(G)}.$$

Therefore,

$$h_{\mathbb{R}}(G) = \inf_{\substack{x\in\mathbb{R}^V\setminus\mathrm{span}(\{\mathbb{1}\})\\ \langle x,\deg_G\rangle=0}} \frac{x^T L_G x}{x^T\,\mathrm{Diag}(\deg_G)x}.$$

Let $D \coloneqq \mathrm{Diag}(\deg_G)$ and let $y \coloneqq D^{\frac{1}{2}}x$. Hence,

$$\begin{aligned} h_{\mathbb{R}}(G) &= \inf_{\substack{x\in\mathbb{R}^V\setminus\mathrm{span}(\{\mathbb{1}\})\\ \langle x,\deg_G\rangle=0}} \frac{x^T L_G x}{x^T D x} = \inf_{\substack{y\in\mathbb{R}^V\setminus\mathrm{span}(\{D^{1/2}\mathbb{1}\})\\ \langle D^{-1/2}y,\deg_G\rangle=0}} \frac{(D^{-\frac{1}{2}}y)^T L_G (D^{-\frac{1}{2}}y)}{(D^{-\frac{1}{2}}y)^T D (D^{-\frac{1}{2}}y)} \\ &= \inf_{\substack{y\in\mathbb{R}^V\setminus\mathrm{span}(\{\mathbb{1}\})\\ \langle y,D^{-1/2}\deg_G\rangle=0}} \frac{y^T D^{-\frac{1}{2}} L_G D^{-\frac{1}{2}} y}{y^T y} = \inf_{\substack{y\in\mathbb{R}^V\setminus\mathrm{span}(\{\mathbb{1}\})\\ \langle y,D^{1/2}\mathbb{1}\rangle=0}} \frac{y^T \widetilde{L}_G y}{y^T y}. \end{aligned}$$

It is easy to verify that $D^{1/2}\mathbb{1} \in \mathrm{Null}(\widetilde{L}_G)$. So $\lambda_{\min}(\widetilde{L}_G) = 0$, and by Theorem 2.8 we have that

$$\inf_{\substack{y\in\mathbb{R}^V\setminus\mathrm{span}(\{\mathbb{1}\})\\ \langle y,D^{-1/2}\mathbb{1}\rangle=0}} \frac{y^T \widetilde{L}_G y}{y^T y} = \lambda_2^{\uparrow}(\widetilde{L}_G).$$

□

Let $G = (V, E)$ be a graph. The above theorem together with Proposition 3.13 implies that

$$\frac{\lambda_2^{\uparrow}(\widetilde{L}_G)}{2} \le \Phi(G).$$

A natural question that arises is how small can $\lambda_2^{\uparrow}(\widetilde{L}_G)$ get when compared to $\Phi(G)$. If $\lambda_2^{\uparrow}(\widetilde{L}_G)$ gets too small in some cases, it is of no use as an approximation to the sparsity of the graph. Luckily, there is a lower bound of $\lambda_2^{\uparrow}(\widetilde{L}_G)$ when comparing it to the sparsity, known as Cheeger's inequality. Although we do not prove it since this is not the focus of this text, we state the result for the sake of completeness.

Theorem 3.16 ([16])**.** If $G = (V, E)$ is a graph, then

$$\frac{\lambda_2^{\uparrow}(\widetilde{L}_G)}{2} \le \Phi(G) \le 2\sqrt{\lambda_2^{\uparrow}(\widetilde{L}_G)}.$$

Chapter 4

The Conjugate Gradient Method

Solving a linear system is a fundamental and important task, and it is an essential subroutine in many algorithms in computer science. Sometimes, solving a linear system with direct methods, i.e. methods that find an exact solution in a finite number of operations, may be prohibitively expensive due to the size of the matrix. Moreover, one may be willing to accept an approximate solution if it can be obtained efficiently.

In this chapter we describe the Conjugate Gradient method, which is an improvement over the Gradient Descent technique when applied to solving an important class of linear systems. This is an iterative algorithm, meaning that at each iteration we have a candidate solution, and its error decreases as the algorithm executes more iterations. We shall see that the number of iterations required to find a solution with accuracy ε depends only on $\ln(1/\varepsilon)$ and on the square root of the condition number of the matrix. Moreover, each iteration runs in time proportional to the number of non-zero entries of the matrix. In Chapter 5 we will use the Conjugate Gradient method to construct a very efficient approximate solver for Laplacian systems.

4.1 Improving Gradient Descent

Gradient Descent is a widely known iterative method to approximately minimize a smooth function $f : \mathbb{R}^n \to \mathbb{R}$. The method starts with a given point $x_0 \in \mathbb{R}^n$ and, for $t \in \mathbb{N} \setminus \{0\}$, it iteratively computes

$$x_t \coloneqq x_{t-1} - \eta_t \nabla f(x_{t-1}), \tag{4.1}$$

where $\eta_t \in \mathbb{R}_+$ is the *step size* at iteration t chosen according to some rule. The idea of the method is that f decreases most rapidly at a point $x \in \mathbb{R}^n$ in the direction $-\nabla f(x)$. Depending on the function f it may even be possible to compute η_t that maximizes the decrease $f(x_{t-1}) - f(x_t)$ at iteration $t \in \mathbb{N} \setminus \{0\}$.

Suppose now that we want to solve the linear system of equations $Ax = b$, where $A \in \mathbb{S}^n_+$ and $b \in \operatorname{Im}(A)$. Let

$$f(x) \coloneqq \tfrac{1}{2} x^T A x - b^T x, \qquad \forall x \in \mathbb{R}^n. \tag{4.2}$$

The next proposition shows that even when $b \in \mathbb{R}^n$ is not necessarily in $\operatorname{Im}(A)$, it is enough to minimize f over $\operatorname{Im}(A)$ to find the minimum norm solution of $Ax = b$.

Proposition 4.1. Let $A \in \mathbb{S}^n_+$, let $b \in \mathbb{R}^n$ and let f be defined as in (4.2). Then every point in $A^\dagger b + \operatorname{Null}(A)$ is a global minimizer of f. In particular, $A^\dagger b$ is the unique global minimizer of f over $\operatorname{Im}(A)$.

Proof. Let $x^* \coloneqq A^\dagger b$. Then, for every $d \in \mathbb{R}^n$,

$$\begin{aligned}
f(x^* + d) &= \tfrac{1}{2}(x^* + d)^T A (x^* + d) - (x^* + d)^T b \\
&= \tfrac{1}{2}\big((x^*)^T A x^* + 2 d^T A x^* + d^T A d\big) - (x^*)^T b - d^T b \\
&= \tfrac{1}{2}(x^*)^T A x^* - (x^*)^T b + d^T A x^* - d^T b + \tfrac{1}{2} d^T A d \\
&= f(x^*) + d^T b - d^T b + \tfrac{1}{2} d^T A d \\
&= f(x^*) + \tfrac{1}{2} d^T A d \geq f(x^*),
\end{aligned}$$

with equality if and only if $d \in \operatorname{Null}(A)$. □

Suppose that $b \in \mathrm{Im}(A)$. Note that $\nabla f(x) = Ax - b \in \mathrm{Im}(A)$. Hence, if we pick an initial point x_0 in the image of A, then by equation (4.1) we have that every iterate x_t is also in the image of A. Therefore, choosing such a initial point causes gradient descent to minimize f over the image of A. Hence, we can solve $Ax = b$ using gradient descent in this case. However, we shall improve this method.

Let $A \in \mathbb{R}^{n\times n}$ and let $b \in \mathbb{R}^n$. For each $t \in \mathbb{N} \setminus \{0\}$, the t-th **Krylov subspace** (generated by A and b) is

$$\mathcal{K}_t(A, b) \coloneqq \mathrm{span}\{A^0 b, A^1 b, \dots, A^{t-1} b\},$$

and define $\mathcal{K}_0(A, b) \coloneqq \{0\}$. For every $t \in \mathbb{N}$ the **residual** (at iteration t) is $r_t \coloneqq A(x^* - x_t) = -\nabla f(x_t)$, where $x^* \coloneqq A^\dagger b$. Note that $\mathcal{K}_t(A, b) \subseteq \mathrm{Im}(A)$ if $b \in \mathrm{Im}(A)$. Moreover, note that for every $i \in \mathbb{N}$,

$$\mathcal{K}_i(A, b) \subseteq \mathcal{K}_{i+1}(A, b).$$

Proposition 4.2. Let $A \in \mathbb{S}^n_+$, let $b \in \mathrm{Im}(A)$, and let $x_0 \in \mathbb{R}^n$. Define f as in (4.2) and define

$$x_t \coloneqq x_{t-1} - \eta_t \nabla f(x_{t-1}), \qquad \forall t \in \mathbb{N} \setminus \{0\},$$

where $\eta_t \in \mathbb{R}$ for each $t \in \mathbb{N} \setminus \{0\}$. Then $x_t \in x_0 + \mathcal{K}_t(A, r_0)$ for each $t \in \mathbb{N}$, where r_0 is the residual.

Proof. Let r_t be the residual at the t-th iteration. Let us prove that

$$r_t \in \mathcal{K}_{t+1}(A, r_0) \qquad \forall t \in \mathbb{N} \tag{4.3}$$

by induction on t. For $t = 0$, we have $r_t = r_0 \in \mathrm{span}\{r_0\} = \mathcal{K}_1(A, r_0)$. Let $t \in \mathbb{N} \setminus \{0\}$. Define $x^* \coloneqq A^\dagger b$. Note that

$$r_t = A(x^* - x_t) = A(x^* - x_{t-1}) - \eta_t A r_{t-1} = r_{t-1} - \eta_t A r_{t-1}.$$

By the induction hypothesis $r_{t-1} \in \mathcal{K}_t(A, r_0) \subseteq \mathcal{K}_{t+1}(A, r_0)$, therefore $Ar_{t-1} \in A\mathcal{K}_t(A, r_0) \subseteq \mathcal{K}_{t+1}(A, r_0)$. Thus $r_t \in \mathcal{K}_{t+1}(A, r_0)$. This concludes the proof of (4.3).

Let $t \in \mathbb{N}$. By induction, we have

$$x_t = x_0 + \sum_{i=0}^{t-1} \eta_{i+1} r_i.$$

Hence, by (4.3) we conclude that $x_t \in x_0 + \mathcal{K}_t(A, r_0)$ for every $t \in \mathbb{N}$. □

Proposition 4.2 shows that the search space in the first t iterations is the affine space $x_0 + \mathcal{K}_t(A, r_0)$. In an ideal scenario, we would like x_t to minimize f over this affine space. But this need not be the case, and this leads us to an idea of how to improve gradient descent. Namely, at iteration t, we will find a minimizer of f over $x_0 + \mathcal{K}_t(A, r_0)$. This is the main idea behind the Conjugate Gradient method.

Suppose we have a set $\{p_1, \dots, p_t\} \subseteq \mathbb{R}^n$ such that $\{p_1 \dots, p_i\}$ is a basis of $\mathcal{K}_i(A, r_0)$ for each $i \in [t]$ and

$$f\Big(x_0 + \sum_{i=1}^{t} c_i p_i\Big) - f(x_0) = \sum_{i=1}^{t} (f(x_0 + c_i p_i) - f(x_0)) \qquad \forall c \in \mathbb{R}^t, \tag{4.4}$$

that is, the function f is separable over $\{p_1, \dots, p_t\}$. Let $i \in [t-1]$ and let $x_i \in x_0 + \mathcal{K}_i(A, r_0)$ be a minimizer of f over $x_0 + \mathcal{K}_i(A, r_0)$. We can write $x_i - x_0$ with respect to the basis $\{p_1, \dots, p_i\}$, that is, there is $c \in \mathbb{R}^i$

such that $x_i = x_0 + \sum_{j=1}^{i} c_j p_j$. Therefore, using the separability property in (4.4),

$$
\begin{aligned}
\min\{ f(x) - f(x_0) : x \in x_0 + \mathcal{K}_{i+1}(A, r_0)\} \\
&= \min_{b \in \mathbb{R}^{i+1}} \Big(f\Big(x_0 + \sum_{j=1}^{i+1} b_j p_j \Big) - f(x_0) \Big) \\
&= \min_{b \in \mathbb{R}^{i+1}} \sum_{j=1}^{i+1} (f(x_0 + b_j p_j) - f(x_0)) \\
&= \min_{b \in \mathbb{R}^{i}} \sum_{j=1}^{i} (f(x_0 + b_j p_j) - f(x_0)) + \min_{\alpha \in \mathbb{R}} \Big(f(x_0 + \alpha p_{i+1}) - f(x_0) \Big) \\
&= \min_{b \in \mathbb{R}^{i}} \Big(f\Big(x_0 + \sum_{j=1}^{i} b_j p_j \Big) - f(x_0) \Big) + \min_{\alpha \in \mathbb{R}} \Big(f(x_0 + \alpha p_{i+1}) - f(x_0) \Big) \\
&= \min\{ f(x) - f(x_0) : x \in x_0 + \mathcal{K}_{i}(A, r_0)\} + \min_{\alpha \in \mathbb{R}} \Big(f(x_0 + \alpha p_{i+1}) - f(x_0) \Big) \\
&= f(x_i) - f(x_0) + \min_{\alpha \in \mathbb{R}} \Big(f(x_0 + \alpha p_{i+1}) - f(x_0) \Big).
\end{aligned}
$$

This means that if we have a minimizer of f over $x_0 + \mathcal{K}_i(A, r_0)$, then it is enough to solve the one-dimensional problem $\min_{\alpha \in \mathbb{R}} f(x_0 + \alpha p_{i+1})$ to find a minimizer of f over $x_0 + \mathcal{K}_{i+1}(A, r_0)$. Hence, in this case it is possible to solve $\min_{x \in x_0 + \mathcal{K}_t(A, r_0)} f(x)$ iteratively. The idea of the Conjugate Gradient method is that, at iteration $t \in \mathbb{N}$, the method has a minimizer of f over $x \in x_0 + \mathcal{K}_{t-1}(A, r_0)$, as well as a basis of $x_0 + \mathcal{K}_t(A, r_0)$ that satisfies the separability property in (4.4). With this information, the method efficiently computes a minimizer of f over $x_0 + \mathcal{K}_t(A, r_0)$, as well as a basis of $x_0 + \mathcal{K}_{t+1}(A, r_0)$ that satisfies the separability property in (4.4). We will see that a minimizer of f over $x_0 + \mathcal{K}_t(A, r_0)$ is an approximate solution to the linear system $Ax = b$, with the error decreasing as t approximates n. Hence, every iteration of the Conjugate Gradient method has an approximate solution to the linear system $Ax = b$. We will see that, if $x_0, b \in \operatorname{Im}(A)$ then x_n is an exact solution of $Ax = b$.

Throughout the remainder of the chapter, we will use $A \in \mathbb{S}^n_+$, a vector $b \in \operatorname{Im}(A)$, the function

$$
f(x) \coloneqq \frac{1}{2} x^T A x - x^T b,
$$

and $x^* \coloneqq A^\dagger b$. Any exceptions on the use of this notation, if they exist, will be clearly stated.

4.2 The Gram-Schmidt Method and Krylov subspaces

Theorem 4.3 (Gram-Schmidt Method)**.** *Let V be a vector space over $\mathbb{R}$. Let $\{v_1, \dots, v_k\} \subseteq V$ be linearly independent and let $\langle \cdot, \cdot \rangle$ be an inner-product on V. Define $w_1, \dots, w_k \in V$, in this order, by the formula*

$$
w_i \coloneqq v_i - \sum_{j=1}^{i-1} \frac{\langle v_i, w_j \rangle}{\langle w_j, w_j \rangle} w_j \qquad \forall i \in [k]. \tag{4.5}
$$

Then for every $i \in [k]$ the set $\{w_1, \dots, w_i\}$ is an orthogonal basis of $\operatorname{span}\{v_1, \cdots, v_i\}$ with respect to $\langle \cdot, \cdot \rangle$.

Proof. Our proof is by induction on i. For $i = 1$ the statement holds since $v_1 = w_1$. Let $r \in [k] \setminus \{1\}$. Let us first show that

$$
\{w_1, \dots, w_{r-1}, v_r\} \text{ is linearly independent.} \tag{4.6}
$$

Let $b \in \mathbb{R}^r$ be such that

$$
\sum_{i=1}^{r-1} b_i w_i + b_r v_r = 0.
$$

Let us show that $b = 0$. If $b_r = 0$, we have that $b = 0$ since $\{w_1, \ldots, w_{r-1}\}$ is linearly independent by induction hypothesis. Hence, suppose that $b_r \neq 0$. In this case, $v_r \in \operatorname{span}\{w_1, \ldots, w_{r-1}\} = \operatorname{span}\{v_1, \ldots, v_{r-1}\}$, where the equality follows by induction hypothesis. Hence $v_r \in \operatorname{span}\{v_1, \ldots, v_{r-1}\}$, which is a contradiction since $\{v_1, \ldots, v_r\}$ is linearly independent. Therefore, $b = 0$. This concludes the proof of (4.6). For every $i \in [r-1]$, notice that

$$\begin{aligned}
\langle w_r, w_i \rangle &= \left\langle v_r - \sum_{j=1}^{r-1} \frac{\langle v_r, w_j \rangle}{\langle w_j, w_j \rangle} w_j, w_i \right\rangle \\
&= \langle v_r, w_i \rangle - \sum_{j=1}^{r-1} \frac{\langle v_r, w_j \rangle}{\langle w_j, w_j \rangle} \langle w_j, w_i \rangle \\
&= \langle v_r, w_i \rangle - \frac{\langle v_r, w_i \rangle}{\langle w_i, w_i \rangle} \langle w_i, w_i \rangle \qquad \text{Since } \langle w_j, w_i \rangle = 0 \text{ if } i \neq j. \\
&= \langle v_r, w_i \rangle - \langle v_r, w_i \rangle = 0.
\end{aligned}$$

Therefore, the set $\{w_1, \ldots, w_r\}$ is orthogonal by induction hypothesis and $\dim(\operatorname{span}\{w_1, \ldots, w_r\}) = r$. It only remains to prove that

$$\operatorname{span}\{v_1, \ldots, v_r\} = \operatorname{span}\{w_1, \ldots, w_r\}. \tag{4.7}$$

By equation (4.5), we know that $v_r \in \operatorname{span}\{w_1, \ldots, w_r\}$. Moreover, using the induction hypothesis we have that $v_i \in \operatorname{span}\{w_1, \ldots, w_i\}$ for every $i \in [r-1]$. Therefore $\operatorname{span}\{v_1, \ldots, v_r\} \subseteq \operatorname{span}\{w_1, \ldots, w_r\}$, but since $\{v_1, \ldots, v_r\}$ is a linearly independent set, we have that $\dim(\operatorname{span}\{v_1, \ldots, v_r\}) = \dim(\operatorname{span}\{w_1, \ldots, w_r\})$. This ends the proof of (4.7). ☐

Define $\langle \cdot, \cdot \rangle_A \colon \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ as

$$\langle x, y \rangle_A \coloneqq x^T A y, \qquad \forall x, y \in \mathbb{R}^n.$$

Since A is symmetric, $\langle x, y \rangle_A = \langle y, x \rangle_A$. If $x, y, z \in \operatorname{Im}(A)$ and $\alpha \in \mathbb{R}$, it is easy to see that $\langle \alpha(x+y), z \rangle = \alpha\langle x, z \rangle + \alpha\langle y, z \rangle$. Moreover, if $x \in \operatorname{Im}(A)$, then $\langle x, x \rangle_A \geq 0$ since A is positive semidefinite, and by Proposition 2.12, equality holds if end only if $x \in \operatorname{Im}(A) \cap \operatorname{Null}(A) = \{0\}$. Hence, $\langle \cdot, \cdot \rangle_A$ is an inner-product on $\operatorname{Im}(A)$. A set $S \subseteq \operatorname{Im}(A)$ is ***A*-orthogonal** if it is orthogonal with respect to $\langle \cdot, \cdot \rangle_A$. Therefore, if we have a basis of $\mathcal{K}_t(A, b)$, we can A-orthogonalize it using the Gram-Schmidt method with $\langle \cdot, \cdot \rangle_A$.

Lemma 4.4. Let $A \in \mathbb{S}^n_+$, let $b \in \mathbb{R}^n$, and let $\{w_1, \ldots, w_k\} \subseteq \mathbb{R}^n$ such that $\operatorname{span}\{w_1, \ldots, w_i\} = \mathcal{K}_i(A, b)$ for every $i \in [k]$. Then $\mathcal{K}_{k+1}(A, b) = \operatorname{span}\{w_1, \ldots, w_k, Aw_k\}$.

Proof. If $i \in [k-1]$, then $w_i \in \mathcal{K}_i(A, b)$. Hence,

$$Aw_i \in A\mathcal{K}_i(A, b) \subseteq \mathcal{K}_{i+1}(A, b) = \operatorname{span}\{w_1, \ldots, w_{i+1}\} \subseteq \operatorname{span}\{w_1, \ldots, w_k\}, \qquad \forall i \in [k-1] \tag{4.8}$$

Since $\operatorname{span}\{w_1, \ldots, w_k\} = \mathcal{K}_k(A, b)$, we have $\operatorname{span}\{w_1, \ldots, w_k, A^k b\} = \mathcal{K}_{k+1}(A, b)$. Hence,

$$A^k b = A(A^{k-1} b) \in A\mathcal{K}_k(A, b) = A \operatorname{span}\{w_1, \ldots, w_k\} = \operatorname{span}\{Aw_1, \ldots, Aw_k\} \subseteq \operatorname{span}\{w_1, \ldots, w_k, Aw_k\},$$

where in the last inequality we used (4.8). Thus, $A^k b \in \operatorname{span}\{w_1, \ldots, w_k, Aw_k\}$. Hence, $\mathcal{K}_{k+1}(A, b) \subseteq \operatorname{span}\{w_1, \ldots, w_k, Aw_k\}$. Note that $Aw_k \in \mathcal{K}_{k+1}(A, b)$. Since $w_i \in \mathcal{K}_i(A, b)$ for every $i \in [k]$, we have that $\operatorname{span}\{w_1, \ldots, w_k, Aw_k\} \subseteq \mathcal{K}_{k+1}(A, b)$. Therefore, $\operatorname{span}\{w_1, \ldots, w_k, Aw_k\} = \mathcal{K}_{k+1}(A, b)$. ☐

Corollary 4.5. Let $A \in \mathbb{S}^n_+$ and $b \in \operatorname{Im}(A)$ be such that $\{A^0 b, A^1 b, \ldots, A^{t-1} b\} \subseteq \operatorname{Im}(A)$ is linearly independent. Set $w_1 \coloneqq b$ and define $w_2, \ldots, w_t \in \operatorname{Im}(A)$, in this order, by the formula

$$w_{i+1} \coloneqq Aw_i - \frac{\langle Aw_i, w_i \rangle_A}{\langle w_i, w_i \rangle_A} w_i - [i \geq 2] \frac{\langle Aw_i, w_{i-1} \rangle_A}{\langle w_{i-1}, w_{i-1} \rangle_A} w_{i-1} \qquad \forall i \in [t-1].$$

Then for each $i \in [t]$ the set $\{w_1, \ldots, w_i\}$ is an A-orthogonal basis of $\mathcal{K}_i(A, b)$.

Proof. Our proof is by induction on t. If $t = 1$, the statement clearly holds. Let $t > 1$. By the induction hypothesis,

$$\{w_1, \dots, w_i\} \text{ is an } A\text{-orthogonal basis of } \mathcal{K}_i(A, b) \text{ for every } i \in [t-1]. \tag{4.9}$$

Hence, by Lemma 4.4,

$$\operatorname{span}\{w_1, \dots, w_{t-1}, Aw_{t-1}\} = \mathcal{K}_t(A, b). \tag{4.10}$$

Moreover, $\dim(\mathcal{K}_t(A, b)) = t$ since $\{A^0 b, A^1 b, \dots, A^{t-1} b\}$ is linearly independent. Hence, $\dim(\mathcal{K}_t(A, b)) = t$ and (4.10) imply that $\{w_1, \dots, w_{t-1}, Aw_{t-1}\}$ is linearly independent. We are in position now to apply Theorem 4.3. Define

$$w'_i \coloneqq w_i - \sum_{j=1}^{t-1} \frac{\langle w_i, w_j \rangle_A}{\langle w_j, w_j \rangle_A} w_j, \qquad \forall i \in [t],$$

and

$$w'_t \coloneqq Aw_{t-1} - \sum_{j=1}^{t-1} \frac{\langle Aw_{t-1}, w_j \rangle_A}{\langle w_j, w_j \rangle_A} w_j.$$

By Theorem 4.3, we have that $\{w'_1, \dots, w'_i\}$ is an A-orthogonal basis of $\mathcal{K}_i(A, b)$ for every $i \in [t]$. It is easy to see that (4.9) implies that $w'_i = w_i$ for every $i \in [t-1]$. Thus, we will be done once we prove that $w'_t = w_t$.

Let $j \in [t-3]$. Note that $Aw_j \in A\mathcal{K}_j(A, b) \subseteq \mathcal{K}_{j+1}(A, b) = \operatorname{span}\{w_1, \dots, w_{j+1}\}$. Hence, there are scalars $\alpha_1, \dots, \alpha_{j+1} \in \mathbb{R}$ such that $Aw_j = \sum_{i=1}^{j+1} \alpha_i w_i$. Moreover, we have $\langle w_{t-1}, w_i \rangle_A = 0$ for every $i \in [t-2]$ by (4.9). Therefore,

$$\langle Aw_{t-1}, w_j \rangle_A = w_{t-1}^T A^T A w_j = \sum_{i=1}^{j+1} w_{t-1}^T A \alpha_i w_i = \sum_{i=1}^{j+1} \alpha_i \langle w_{t-1}, w_i \rangle_A = 0.$$

Note that in the second equation we used the crucial property that $A = A^T$. Hence,

$$w' = Aw_{t-1} - \sum_{j=1}^{t-1} \frac{\langle Aw_{t-1}, w_j \rangle_A}{\langle w_j, w_j \rangle_A} w_j = Aw_{t-1} - \frac{\langle Aw_{t-1}, w_{t-1} \rangle_A}{\langle w_{t-1}, w_{t-1} \rangle_A} w_{t-1} - [t \geq 3] \frac{\langle Aw_{t-1}, w_{t-2} \rangle_A}{\langle w_{t-2}, w_{t-2} \rangle_A} w_{t-2} = w_t. \qquad \square$$

Proposition 4.6. Let $A \in \mathbb{S}^n_+$, let $b \in \operatorname{Im}(A)$, and let f be defined as in (4.2). Let $\{p_1, \dots, p_t\} \subseteq \mathbb{R}^n$ be an A-orthogonal set, let $x_0 \in \mathbb{R}^n$ and let $c \in \mathbb{R}^t$. Then

$$f\Big(x_0 + \sum_{i=1}^{t} c_i p_i\Big) - f(x_0) = \sum_{i=1}^{t} (f(x_0 + c_i p_i) - f(x_0)).$$

Proof. We have

$$\begin{aligned}
f\Big(x_0 + \sum_{i=1}^{t} c_i p_i\Big) - f(x_0) &= \tfrac{1}{2}\Big(x_0 + \sum_{i=1}^{t} c_i p_i\Big)^T A \Big(x_0 + \sum_{i=1}^{t} c_i p_i\Big) - \Big(x_0 + \sum_{i=1}^{t} c_i p_i\Big)^T b - \tfrac{1}{2} x_0^T A x_0 + x_0^T b \\
&= \sum_{i=1}^{t} c_i p_i^T A x_0 + \tfrac{1}{2}\Big(\sum_{i=1}^{t} c_i p_i\Big)^T A \Big(\sum_{i=1}^{t} c_i p_i\Big) - \sum_{i=1}^{t} c_i p_i^T b \\
&= \sum_{i=1}^{t} c_i p_i^T A x_0 + \tfrac{1}{2} \sum_{i=1}^{t} (c_i p_i)^T A (c_i p_i) - \sum_{i=1}^{t} c_i p_i^T b \\
&= \sum_{i=1}^{t} \big(c_i p_i^T A x_0 + \tfrac{1}{2}(c_i p_i)^T A (c_i p_i) - c_i p_i^T b + \tfrac{1}{2} x_0^T A x_0 - \tfrac{1}{2} x_0^T A x_0 + x_0^T b - x_0^T b\big) \\
&= \sum_{i=1}^{t} \big(\tfrac{1}{2}(x_0 + c_i p_i)^T A (x_0 + c_i p_i) - (x_0 + c_i p_i)^T b - \tfrac{1}{2} x_0^T A x_0 + x_0^T b\big) \\
&= \sum_{i=1}^{t} (f(x_0 + c_i p_i) - f(x_0)).
\end{aligned} \qquad \square$$

4.3 The Conjugate Gradient Iteration

Proposition 4.7. Let $A \in \mathbb{S}^n_+$, let $b, d \in \operatorname{Im}(A)$ with $d \neq 0$, and let $x_0 \in \mathbb{R}^n$. Let f be defined as in (4.2). Define the function

$$g(\alpha) \coloneqq f(x_0 + \alpha d) - f(x_0) \qquad \forall \alpha \in \mathbb{R}.$$

Then

$$\operatorname*{arg\,min}_{\alpha \in \mathbb{R}} g(\alpha) = \frac{d^T r_0}{d^T A d},$$

where r_0 is the residual at iteration 0.

Proof. For each $\alpha \in \mathbb{R}$ we have

$$\begin{aligned} g(\alpha) &= f(x_0 + \alpha d) - f(x_0) \\ &= \tfrac{1}{2}\big((x_0 + \alpha d)^T A (x_0 + \alpha d)\big) - (x_0 + \alpha d)^T b - \tfrac{1}{2} x_0^T A x_0 + x_0^T b \\ &= \frac{\alpha^2}{2} d^T A d + \alpha d^T A x_0 - \alpha d^T b = \frac{\alpha^2}{2} d^T A d - \alpha d^T (b - A x_0) \\ &= \frac{\alpha^2}{2} d^T A d - \alpha d^T r_0 \end{aligned}$$

Proposition 2.12, $d^T A d = 0$ if and only if $d \in \operatorname{Null}(A)$. Hence, since $d \in \operatorname{Im}(A) \setminus \{0\}$ and $A \succeq 0$, we have $d^T A d > 0$. Therefore, g is a quadratic function of α and the coefficient of α^2 is positive. Thus, the unique minimizer of g is $\frac{d^T r_0}{d^T A d}$. ☐

Proposition 4.8. Let $A \in \mathbb{S}^n_+$ and let $x_0, b \in \operatorname{Im}(A)$. Let $T \in \mathbb{N}$ and let $\{w_1, \ldots, w_T\} \subseteq \mathbb{R}^n$ be an A-orthogonal set such that $\{w_1, \ldots, w_t\}$ is a basis of $\mathcal{K}_t(A, r_0)$ for each $t \in [T]$, where r_0 is the residual. Define $x_1, \ldots, x_T$, in this order, by the formula

$$x_t \coloneqq x_{t-1} + \eta_t w_t \qquad \forall t \in [T],$$

where

$$\eta_t \coloneqq \frac{w_t^T r_0}{\langle w_t, w_t \rangle_A}.$$

Then x_t is a global minimizer of f over $x_0 + \mathcal{K}_t(A, r_0)$ for each $t \in \{0, \ldots, T\}$.

Proof. Let $g(x) \coloneqq f(x) - f(x_0)$. Note that $x \in \mathbb{R}^n$ is a global minimizer of f if and only if x is a global minimizer of g since x_0 is fixed. Therefore, to prove that x_t is a global minimizer of f over $x_0 + \mathcal{K}_t(A, r_0)$ for each $t \in [T]$, it suffices to prove that x_t is a global minimizer of g over $\mathcal{K}_t(A, r_0)$ for each $t \in [T]$.

For $t = 0$ the statement follows trivially. Let $t \geq 1$. Then

$$\begin{aligned} &\min\{\, g(x) : x \in x_0 + \mathcal{K}_t(A, r_0)\} \\ &\quad = \min\{\, f(x) - f(x_0) : x \in x_0 + \mathcal{K}_t(A, r_0)\} \\ &\quad = \min_{b \in \mathbb{R}^t} f\Big(x_0 + \sum_{i=1}^{t} b_i w_i\Big) - f(x_0) \\ &\quad = \min_{b \in \mathbb{R}^{t-1}} f\Big(x_0 + \sum_{i=1}^{t-1} b_i w_i\Big) - f(x_0) + \min_{\beta \in \mathbb{R}} f(x_0 + \beta w_t) - f(x_0) && \text{by Prop. 4.6} \\ &\quad = f(x_{t-1}) - f(x_0) + \min_{\beta \in \mathbb{R}} f(x_0 + \beta w_t) - f(x_0) && \text{by induction hypothesis} \\ &\quad = f(x_{t-1} - x_0 + x_0) - f(x_0) + f(x_0 + \eta_t w_t) - f(x_0) && \text{by Prop. 4.7} \end{aligned}$$

Note that $x_{t-1} \in x_0 + \mathcal{K}_{t-1}(A, r_0)$ by the induction hypothesis. Hence, we can write $x_{t-1} - x_0$ in the A-orthonormal basis $\{w_1, \ldots, w_{t-1}\}$. Therefore, by Proposition 4.6, we have

$$\begin{aligned} f(x_{t-1} - x_0 + x_0) - f(x_0) + f(x_0 + \eta_t w_t) - f(x_0) &= f(x_{t-1} - x_0 + \eta_t w_t + x_0) - f(x_0) \\ &= f(x_{t-1} + \eta_t w_t) - f(x_0) \\ &= f(x_t) - f(x_0) = g(x_t). \end{aligned}$$

☐

Lemma 4.9. Let $A \in \mathbb{S}^n_+$, let $b, x_0 \in \operatorname{Im}(A)$, let $r_0 \coloneqq b - Ax_0$ be the residual and let $t \in \mathbb{N} \setminus \{0\}$. If $\{A^0 r_0, A^1 r_0, \dots, A^{t-1} r_0\}$ is linearly dependent, then $A^\dagger b \in x_0 + \mathcal{K}_{t-1}(A, r_0)$.

Proof. Suppose that $\{A^0 r_0, A^1 r_0, \dots, A^{t-1} r_0\}$ is linearly dependent. Then there is $c \in \mathbb{R}^{[t]-1} \setminus \{0\}$ such that

$$\sum_{i=0}^{t-1} c_i A^i r_0 = 0. \tag{4.11}$$

Let $k \coloneqq \min\{\, i \in [t] - 1 : c_i \neq 0\}$. Then,

$$c_k A^k r_0 = -\sum_{i=k+1}^{t-1} c_i A^i r_0 \implies A^k r_0 = -\sum_{i=k+1}^{t-1} \frac{c_i}{c_k} A^i r_0. \tag{4.12}$$

By Proposition 2.23, we know that $A^\dagger A$ is the orthogonal projector onto $\operatorname{Im}(A)$. Hence, since $r_0 \in \operatorname{Im}(A)$, left-multiplying the rightmost equation in (4.12) by $(A^\dagger)^{k+1}$ yields

$$A^\dagger r_0 = \sum_{i=k+1}^{t-1} \frac{c_i}{c_k} A^{i-(k+1)} r_0 \in \mathcal{K}_{t-1}(A, r_0).$$

Note that $A^\dagger r_0 = A^\dagger b - x_0$. Hence, $A^\dagger b \in x_0 + \mathcal{K}_{t-1}(A, r_0)$. □

Algorithm 4.1 The Conjugate Gradient method

Input: A matrix $A \in \mathbb{S}^n_+$, vectors $b, x_0 \in \operatorname{Im}(A)$, and $T \in \mathbb{N}$.
Output: A global minimizer of f over $x_0 + \mathcal{K}_T(A, r_0)$.

  $r_0 \leftarrow b - Ax_0$
  $p_0 \leftarrow r_0$
  for $t = 0$ to $T - 1$ **do**
    if $Ax_t = b$ **then return** x_t
    $\eta_{t+1} \leftarrow \frac{p_t^T r_0}{\|p_t\|_A^2}$
    $x_{t+1} \leftarrow x_t + \eta_{t+1} p_t$
    $p_{t+1} \leftarrow Ap_t - \frac{\langle Ap_t, p_t \rangle_A}{\langle p_t, p_t \rangle_A} p_t - [t \geq 1] \frac{\langle Ap_t, p_{t-1} \rangle_A}{\langle p_{t-1}, p_{t-1} \rangle_A} p_{t-1}$
  return x_T

The pseudocode for the Conjugate Gradient method is given in Algorithm 4.1. Let us show the correctness of this algorithm by arguing that, at the beginning of iteration $t \in \mathbb{N}$ of the for-loop in Algorithm 4.1, the following invariants hold:

(i) $\{p_0, \dots, p_t\}$ is an A-orthogonal basis of $\mathcal{K}_{t+1}(A, r_0)$;

(ii) x_t is a global minimizer of f over $x_0 + \mathcal{K}_t(A, r_0)$.

For $t = 0$, both invariants hold trivially. Suppose that these invariants hold at the beginning of iteration t for some $t \in \mathbb{N}$. Let us argue that, if the algorithm does not terminate, the invariants still hold at the beginning of iteration $t + 1$.

If $\{A^0 r_0, \dots, A^t r_0\}$ is linearly dependent, it follows from Lemma 4.9 that $A^\dagger b \in x_0 + \mathcal{K}_t(A, r_0)$. By invariant (ii), we have that x_t is a global minimizer of f over $x_0 + \mathcal{K}_t(A, r_0)$. Hence, $x_t = A^\dagger b$ in this case, and the algorithm terminates. Suppose now that $\{A^0 r_0, \dots, A^t r_0\}$ is linearly independent. By Corollary 4.5, the set $\{p_0, \dots, p_t, p_{t+1}\}$ is an A-orthogonal basis of $\mathcal{K}_{t+2}(A, b)$. Moreover, invariants (i) and (ii), together with Proposition 4.8, imply that x_{t+1} is a global minimizer of f over $x_0 + \mathcal{K}_t(A, r_0)$. Therefore, invariants (i) and (ii) hold in the beginning of iteration $t + 1$. Let us analyze the running time of Algorithm 4.1.

At each iteration, the algorithm makes a constant number of left-multiplications of vectors in $\operatorname{Im}(A)$ by A, besides a constant number of dot products in $\mathbb{R}^n$. Therefore, if t_A is the time it takes to left-multiply a vector in $\operatorname{Im}(A)$ by A, then Conjugate Gradient runs in time $O(T(t_A + n))$. It is important to notice that we use $x_0 = 0$ in Section 4.5. This is done to simplify our analysis of Conjugate Gradient, and this choice does not affect the asymptotic worst-case running time of the Method, neither its correctness.

It remains now to study the speed of convergence of the Conjugate Gradient method. That is, given $t \in \mathbb{N}$, how well the iterate x_t approximates a solution to the linear system $Ax = b$.

4.4 Error Analysis with Polynomials

Let $\mathbb{R}[\lambda]$ be the set of all the polynomials on the indeterminate λ with real coefficients and let $\mathbb{R}[\lambda]_{\leq t} \subset \mathbb{R}[\lambda]$ be the set of all polynomials of the indeterminate λ of degree at most $t \in \mathbb{N}$. If $t \in \mathbb{N}$, $c \in \mathbb{R}^{0 \cup [t]}$, and $p \in \mathbb{R}[\lambda]_{\leq t}$ is such that

$$p(\lambda) = \sum_{i=0}^{t} c_i \lambda^i,$$

then

$$p(A) := \sum_{i=0}^{t} c_i A^i.$$

Let

$$\mathcal{Q}_t := \{\, 1 - \lambda p(\lambda) : p \in \mathbb{R}[\lambda]_{\leq t-1} \}$$

be the set of polynomials of degree at most t that evaluate to 1 at 0, where $t \in \mathbb{N}$ with $t \geq 1$. Recall that $x^* := A^{\dagger} b$ and that $f : \mathbb{R}^n \to \mathbb{R}$ is given by

$$f(x) := \frac{1}{2} x^T A x - x^T b,$$

where $A \in \mathbb{S}^n$ and $b \in \operatorname{Im}(A)$. Let $y \in \mathbb{R}^n$. Note that

$$\begin{aligned} f(y) - f(x^*) &= \tfrac{1}{2} y^T A y - y^T b - \tfrac{1}{2} (x^*)^T A x^* + (x^*)^T b \\ &= \tfrac{1}{2} y^T A y - y^T A x^* + \tfrac{1}{2} (x^*)^T A x^* \\ &= \tfrac{1}{2} (y^T A y - 2 y^T A x^* + (x^*)^T A x^*) \\ &= \tfrac{1}{2} (y - x^*)^T A (y - x^*) = \tfrac{1}{2} \|y - x^*\|_A^2. \end{aligned} \tag{4.13}$$

Lemma 4.10. If $A \in \mathbb{S}^n_+$ and $p \in \mathbb{R}[\lambda]$, then

$$\|p(A)v\|_A^2 \leq \|v\|_A^2 \cdot \max_{i \in [\operatorname{rank}(A)]} p\Big(\lambda_i^{\downarrow}(A)\Big)^2 \qquad \forall v \in \operatorname{Im}(A).$$

Proof. Define $\Lambda := \operatorname{Diag}(\lambda^{\downarrow}(A))$. By the spectral decomposition theorem (see Theorem 2.5), there is an orthogonal matrix $Q \in \mathbb{R}^{n \times n}$ such that $A = Q \Lambda Q^T$. Note that $p(A) = Q p(\Lambda) Q^T$. Moreover, it is easy to see that a polynomial applied to a diagonal matrix yields a diagonal matrix. Hence, $p(\Lambda)$ is diagonal. Let $v \in \operatorname{Im}(A)$ and define $c := Q^T v$. We have

$$\begin{aligned} \|p(A)v\|_A^2 &= \|Q p(\Lambda) Q^T v\|_A^2 = v^T Q p(\Lambda) Q^T Q \Lambda Q^T Q p(\Lambda) Q^T v \\ &= v^T Q \Lambda p(\Lambda)^2 Q^T v = c^T \Lambda p(\Lambda)^2 c = \sum_{j=1}^{n} c_j^2 \lambda_j^{\downarrow}(A) p(\lambda_j^{\downarrow}(A))^2 \\ &= \sum_{j=1}^{\operatorname{rank}(A)} c_j^2 \lambda_j^{\downarrow}(A) p(\lambda_j^{\downarrow}(A))^2 \leq \max_{i \in [\operatorname{rank}(A)]} p(\lambda_i^{\downarrow}(A))^2 \sum_{j=1}^{\operatorname{rank}(A)} c_j^2 \lambda_j^{\downarrow}(A) \\ &= c^T \Lambda c \max_{i \in [\operatorname{rank}(A)]} p(\lambda_i^{\downarrow}(A))^2 = v^T Q \Lambda Q^T v \max_{i \in [\operatorname{rank}(A)]} p(\lambda_i^{\downarrow}(A))^2 \\ &= v^T A v \max_{i \in [\operatorname{rank}(A)]} p(\lambda_i^{\downarrow}(A))^2 = \|v\|_A^2 \max_{i \in [\operatorname{rank}(A)]} p(\lambda_i^{\downarrow}(A))^2, \end{aligned}$$

where in the fourth equation we used that diagonal matrices commute.

□

Theorem 4.11. Let $A \in \mathbb{S}^n_+$ and let $b, x_0 \in \operatorname{Im}(A)$. Let $x^* = A^{\dagger} b$ and let r_0 be the residual. Let f be defined as in (4.2) and define

$$x_t := \arg\min\{\, f(x) : x \in x_0 + \mathcal{K}_t(A, r_0) \}. \tag{4.14}$$

Then

$$\begin{aligned}\|x_t - x^*\|_A^2 &\le (f(x_0) - f(x^*)) \min_{q\in\mathcal{Q}_t} \max_{i\in[\operatorname{rank}(A)]} 2q(\lambda_i^{\downarrow}(A))^2 \\ &\le (f(x_0) - f(x^*)) \min_{q\in\mathcal{Q}_t} \max\{\, 2q^2(\lambda) : \lambda \in [\lambda_{\min}^{+}, \lambda_{\max}]\},\end{aligned}$$

where $\lambda_{\min}^{+} \coloneqq \lambda_{\operatorname{rank}(A)}^{\downarrow}(A)$ and $\lambda_{\max} \coloneqq \lambda_{\max}(A)$. In particular, if $x_0 = 0$, then

$$\begin{aligned}\|x_t - x^*\|_A^2 &\le \|x^*\|_A^2 \min_{q\in\mathcal{Q}_t} \max_{i\in[\operatorname{rank}(A)]} |q(\lambda_i^{\downarrow}(A))|^2 \\ &\le \|x^*\|_A^2 \min_{q\in\mathcal{Q}_t} \max\{\, q(\lambda)^2 : \lambda \in [\lambda_{\min}^{+}, \lambda_{\max}]\}.\end{aligned}$$

Proof. If $\{A^0 r_0, A^1 r_0, \ldots, A^{t-1} r_0\}$ is linearly dependent, by Lemma 4.9 we have that $x^* \in x_0 + \mathcal{K}_{t-1}(A, r_0) \subseteq x_0 + \mathcal{K}_t(A, r_0)$. Hence, $\|x_t - x^*\|_A^2 = 0$ and the statement clearly holds in this case.

Suppose now that $\{A^0 r_0, A^1 r_0, \ldots, A^{t-1} r_0\}$ is linearly independent. Thus, the map $y\colon c \in \mathbb{R}^t \mapsto x_0 + \sum_{i=0}^{t-1} c_{i+1} A^i r_0 \in x_0 + \mathcal{K}_t(A, r_0)$ is a bijection. Moreover, define

$$p_d(\lambda) = \sum_{i=0}^{t-1} d_{i+1}\lambda^i, \qquad \forall d \in \mathbb{R}^t.$$

Note that $p\colon \mathbb{R}^t \to \mathbb{R}[\lambda]_{\le t-1}$ is also a bijection. Hence, $p \circ y^{-1}$ is a bijection from $x_0 + \mathcal{K}_t(A, r_0)$ to $\mathbb{R}[\lambda]_{\le t-1}$. Let $c \in \mathbb{R}^t$. Then

$$y_c = x_0 + p_c(A) r_0 = x_0 + p_c(A) A (x^* - x_0).$$

Let $q_c(\lambda) \coloneqq 1 - \lambda p_c(\lambda)$. Note that q is a bijection from $\mathbb{R}^t$ to $\mathcal{Q}_t$. Moreover,

$$\begin{aligned}y_c - x^* &= x_0 + p_c(A) A (x^* - x_0) - x^* = (x_0 - x^*) - p_c(A) A (x_0 - x^*) \\ &= (I - p_c(A) A)(x_0 - x^*) = q_c(A)(x_0 - x^*).\end{aligned}$$

By equation (4.13), we know that $\|w - x^*\|_A^2 = 2(f(w) - f(x^*))$ for each $w \in \mathbb{R}^n$. By definition, x_t is a minimizer of f over $x_0 + \mathcal{K}_t(A, r_0)$. Hence,

$$\|x_t - x^*\|_A^2 = 2(f(x_t) - f(x^*)) = \min_{w\in x_0+\mathcal{K}_t(A,r_0)} 2(f(w) - f(x^*)) = \min_{w\in x_0+\mathcal{K}_t(A,r_0)} \|w - x^*\|_A^2.$$

Therefore,

$$\begin{aligned}\|x_t - x^*\|_A^2 &= \min_{w\in x_0+\mathcal{K}_t(A,r_0)} \|w - x^*\|_A^2 && \text{by definition of } x_t \\ &= \min_{c\in\mathbb{R}^t} \|y_c - x^*\|_A^2 && \text{using the bijection } y \\ &= \min_{z\in\mathcal{Q}_t} \|z(A)(x_0 - x^*)\|_A^2. && \text{using the bijection } q\end{aligned}$$

Lemma 4.10 applied to the above equation yields

$$\begin{aligned}\|x_t - x^*\|_A^2 &\le \min_{q\in\mathcal{Q}_t} \max_{i\in[\operatorname{rank}(A)]} |q(\lambda_i^{\downarrow}(A))|^2 \|x_0 - x^*\|_A^2 \\ &= \min_{q\in\mathcal{Q}_t} \max_{i\in[\operatorname{rank}(A)]} |q(\lambda_i^{\downarrow}(A))|^2 2(f(x_0) - f(x^*)) \\ &\le \min_{q\in\mathcal{Q}_t} \max_{\lambda\in[\lambda_{\min}^{+},\lambda_{\max}]} |q(\lambda)|^2 2(f(x_0) - f(x^*)).\end{aligned}$$ ☐

Corollary 4.12. Let $A \in \mathbb{S}_+^n$, let $b, x_0 \in \operatorname{Im}(A)$, and let r_0 be the residual. Let f be defined as in (4.2). Then,

$$\arg\min\{\, f(x) : x \in x_0 + \mathcal{K}_{\operatorname{rank}(A)}(A, r_0)\} = A^{\dagger} b. \tag{4.15}$$

Proof. Set $k \coloneqq \operatorname{rank}(A)$ and define $q \in \mathcal{Q}_k$ by

$$q(\lambda) \coloneqq \prod_{i=1}^{k}\left(1 - \frac{\lambda}{\lambda_i^{\downarrow}(A)}\right).$$

Set $x^* \coloneqq \arg\min\{\, f(x) : x \in x_0 + \mathcal{K}_{\operatorname{rank}(A)}(A, r_0)\}$. Note that $\lambda_i^{\downarrow}(A)$ is a root of q for every $i \in [k]$. By Theorem 4.11,

$$\|x^* - A^{\dagger}b\|_A^2 \leq \max_{i \in [\operatorname{rank}(A)]} 2\big|q\big(\lambda_i^{\downarrow}(A)\big)\big|^2 (f(x_0) - f(x^*)) = 0. \qquad \square$$

4.5 Improving the Analysis with Chebyshev Polynomials

For $d \in \mathbb{N}$, the degree-d **Chebyshev polynomial** (of the first kind) $T_d \in \mathbb{R}[\lambda]_{\leq d}$ is defined by

$$T_d(\lambda) \coloneqq \begin{cases} 1, & \text{if } d = 0, \\ \lambda, & \text{if } d = 1, \\ 2\lambda T_{d-1}(\lambda) - T_{d-2}(\lambda), & \text{if } d \geq 2. \end{cases}$$

Proposition 4.13. If $\theta \in [-\pi/2, \pi/2]$ and $d \in \mathbb{N}$, then $T_d(\cos\theta) = \cos(d\theta)$.

Proof. Our proof is by induction on $d \in \mathbb{N}$. By definition, if $d = 0$, then $T_0(\cos\theta) = 1 = \cos(0) = \cos(0\theta)$, and if $d = 1$, then $T_1(\cos\theta) = \cos(\theta)$.

Let $d \geq 1$. Let us prove that $T_{d+1}(\cos\theta) = \cos((d+1)\theta)$. We have

$$\begin{aligned} \cos((d+1)\theta) &= \cos\theta\cos(d\theta) - \sin(d\theta)\sin\theta, \\ \cos((d-1)\theta) &= \cos\theta\cos(d\theta) + \sin(d\theta)\sin\theta. \end{aligned}$$

Therefore,

$$\begin{aligned} T_{d+1}(\cos\theta) &= 2\cos\theta\; T_d(\cos\theta) - T_{d-1}(\cos\theta) = 2\cos\theta\cos(d\theta) - \cos((d-1)\theta) \\ &= 2\cos\theta\cos(d\theta) - \cos\theta\cos(d\theta) - \sin(d\theta)\sin(\theta) \\ &= \cos\theta\cos(d\theta) - \sin(d\theta)\sin(\theta) = \cos((d+1)\theta). \end{aligned} \qquad \square$$

Lemma 4.14. For every $\lambda \in \mathbb{R}$ with $|\lambda| \geq 1$ and for every $d \in \mathbb{N}$,

$$T_d(\lambda) = \frac{1}{2}\left(\left(\lambda + \sqrt{\lambda^2 - 1}\right)^d + \left(\lambda - \sqrt{\lambda^2 - 1}\right)^d\right).$$

Proof. Let $\lambda \in \mathbb{R}$ be such that $|\lambda| \geq 1$ and let $\mu \coloneqq \lambda + \sqrt{\lambda^2 - 1}$. Note that

$$\mu(\lambda - \sqrt{\lambda^2 - 1}) = \lambda^2 - \lambda^2 + 1 = 1.$$

Therefore, $\mu^{-1} = \lambda - \sqrt{\lambda^2 - 1}$. Hence, to prove the statement of the lemma is equivalent to prove that

$$T_d(\lambda) = \tfrac{1}{2}(\mu^d + \mu^{-d}) \qquad \forall d \in \mathbb{N}.$$

Let us prove the above claim by induction on d. For the base cases where $d \in \{0, 1\}$, we have $\frac{1}{2}(\mu^0 + \mu^0) = 1 = T_0(\lambda)$ and $\frac{1}{2}(\mu + \mu^{-1}) = \lambda = T_1(\lambda)$.

Let $d > 1$. Note that

$$\mu^2 = \lambda^2 + 2\lambda\sqrt{\lambda^2 - 1} + \lambda^2 - 1 = 2\lambda\mu - 1.$$

Similarly, we have $\mu^{-2} = 2\lambda\mu^{-1} - 1$. Therefore,

$$\begin{aligned} T_d(\lambda) &= 2\lambda T_{d-1}(\lambda) - T_{d-2}(\lambda) = 2\lambda\frac{1}{2}(\mu^{d-1} + \mu^{-(d-1)}) - \frac{1}{2}(\mu^{d-2} + \mu^{-(d-2)}) \\ &= \frac{1}{2}\Big(\mu^{d-2}(2\lambda\mu - 1) + y^{-(d-2)}(2\lambda\mu^{-1} - 1)\Big) = \frac{1}{2}\big(\mu^d + \mu^{-d}\big). \end{aligned} \qquad \square$$

Let $d \in \mathbb{N}$ and let $\alpha, \beta \in \mathbb{R}_{++}$ be such that $\alpha < \beta$. Define the polynomial

$$Q_{\alpha,\beta,d}(\lambda) \coloneqq \frac{T_d\left(\frac{\beta+\alpha-2\lambda}{\beta-\alpha}\right)}{T_d\left(\frac{\beta+\alpha}{\beta-\alpha}\right)}.$$

Note that $Q_{\alpha,\beta,d} \in \mathcal{Q}_d$. Since $\frac{\beta+\alpha}{\beta-\alpha} > 1$, by Lemma 4.14 we have that

$$T_d\left(\frac{\beta+\alpha}{\beta-\alpha}\right) > 0, \qquad \forall k \in \mathbb{N}. \tag{4.16}$$

Lemma 4.15. If $\alpha, \beta \in \mathbb{R}_{++}$ are such that $\alpha < \beta$, then for every $d \in \mathbb{N}$,

$$Q_{\alpha,\beta,d}(\lambda) \leq 2\left(\frac{\sqrt{\beta/\alpha}-1}{\sqrt{\beta/\alpha}+1}\right)^d \qquad \forall \lambda \in [\alpha, \beta].$$

In particular, let $A \in \mathbb{S}^n_+$ be such that $\lambda^+_{\min} \coloneqq \lambda^{\downarrow}_{\mathrm{rank}(A)}(A) < \lambda_{\max} \coloneqq \lambda_{\max}(A)$. Then, for every $d \in \mathbb{N}$,

$$Q_{\lambda^+_{\min},\lambda_{\max},d}(\lambda) \leq 2\left(\frac{\sqrt{\kappa^+(A)}-1}{\sqrt{\kappa^+(A)}+1}\right)^d \qquad \forall \lambda \in [\lambda^+_{\min}, \lambda_{\max}],$$

where $\kappa^+(A) \coloneqq \lambda_{\max}/\lambda^+_{\min}$.

Proof. Note that for every $\lambda \in [\alpha, \beta]$,

$$\frac{\beta+\alpha-2\lambda}{\beta-\alpha} \in [-1, 1].$$

Let $d \in \mathbb{N}$. Thus, by Proposition 4.13,

$$T_t\left(\frac{\beta+\alpha-2\lambda}{\beta-\alpha}\right) \in [-1, 1].$$

Let $\kappa \coloneqq \beta/\alpha$. Therefore, using Lemma 4.14, for every $\lambda \in [\alpha, \beta]$,

$$\begin{aligned} Q_{\alpha,b,d}(\lambda) = \frac{T_d\left(\frac{\beta+\alpha-2\lambda}{\beta-\alpha}\right)}{T_d\left(\frac{\beta+\alpha}{\beta-\alpha}\right)} &\overset{(4.16)}{\leq} T_d\left(\frac{\beta+\alpha}{\beta-\alpha}\right)^{-1} \\ &= T_d\left(\frac{\kappa+1}{\kappa-1}\right)^{-1} = 2\left(\left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}\right)^d + \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^d\right)^{-1} \\ &\leq 2\left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}\right)^{-d} = 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^d, \end{aligned}$$

where in the last inequality we used the fact that $\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^t \geq 0$. ☐

Corollary 4.16. Let $A \in \mathbb{S}^n_+$ be such that $\lambda^{\downarrow}_{\mathrm{rank}(A)}(A) < \lambda_{\max}(A)$, let $b \in \mathrm{Im}(A)$, let $\varepsilon > 0$, and let $x^* = A^{\dagger}b$. Let f be defined as in (4.2) and define

$$x_t \coloneqq \operatorname*{arg\,min}_{x \in \mathcal{K}_t(A,b)} f(x) \tag{4.17}$$

for every $t \in \mathbb{N}$. Then

$$\|x_t - x^*\|_A^2 \leq 2\varepsilon \|x^*\|_A^2$$

for every $t \geq \frac{\sqrt{\kappa^+(A)}}{2} \ln 2/\varepsilon$, where $\kappa^+(A) \coloneqq \lambda_{\max}(A)/\lambda^{\downarrow}_{\mathrm{rank}(A)}(A)$.

Proof. Let $\lambda_{\min}^{+} \coloneqq \lambda_{\operatorname{rank}(A)}^{\downarrow}(A)$, let $\lambda_{\max} \coloneqq \lambda_{\max}(A)$, and let $\kappa \coloneqq \kappa^{+}(A)$. Since $Q_{\lambda_{\min}^{+},\lambda_{\max},t}(x) \in \mathcal{Q}_t$, by Theorem 4.11 we have that

$$\|x_t - x^*\|_A^2 \leq \|x^*\|_A^2 \min_{q\in\mathcal{Q}_t} \max_{\lambda\in[\lambda_{\min}^{+},\lambda_{\max}]} q(\lambda)^2 \leq \|x^*\|_A^2 \max_{\lambda\in[\lambda_{\min}^{+},\lambda_{\max}]} Q_{\lambda_{\min}^{+},\lambda_{\max},t}(\lambda)^2.$$

Using Lemma 4.15 and Lemma 2.2, we get that

$$\begin{aligned}\|x_t - x^*\|_A^2 &\leq \|x^*\|_A^2 \max_{\lambda\in[\lambda_{\min}^{+},\lambda_{\max}]} Q_{\lambda_{\min}^{+},\lambda_{\max},t}(\lambda)^2 \\ &\leq 2\|x^*\|_A^2 \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^{2t} \leq 2\|x^*\|_A^2 \left(1 - \frac{1}{\sqrt{\kappa}}\right)^{2t} \\ &\leq 2\|x^*\|_A^2 \exp\left(-\frac{2t}{\sqrt{\kappa}}\right) \leq \epsilon\|x^*\|_A^2. \qquad \square\end{aligned}$$

Corollary 4.17. Let $A \in \mathbb{S}_{+}^{n}$, let $b \in \operatorname{Im}(A)$, let $\varepsilon > 0$, and let $x^* = A^{\dagger}b$. Let f be defined as in (4.2) and define

$$x_t \coloneqq \operatorname*{arg\,min}_{x\in\mathcal{K}_t(A,b)} f(x) \tag{4.18}$$

for every $t \in \mathbb{N}$. If there are $\alpha, \beta \in \mathbb{R}_{++}$ with $\alpha < \beta$ such that all but c eigenvalues of A are in $[\alpha,\beta] \cup \{0\}$, and the remaining c eigenvalues are all greater than β, then

$$\|x_t - x^*\|_A^2 \leq \varepsilon\|x^*\|_A^2$$

for every $t \geq c + \frac{\sqrt{\beta/\alpha}}{2}\ln\frac{2}{\varepsilon}$.

Proof. Let $\lambda_1,\dots,\lambda_c \in (\beta,+\infty)$ be the eigenvalues of A that are not in $[\alpha,\beta]\cup\{0\}$ and let $r \in \mathbb{N}$. Define the polynomial

$$q_r(\lambda) = Q_{\alpha,\beta,r}(\lambda)\prod_{i=1}^{c}\left(1 - \frac{\lambda}{\lambda_i}\right).$$

Notice that $q_r(\lambda) \in \mathcal{Q}_{r+c}$. Moreover,

$$q_r(\lambda_i) = 0, \qquad \forall i \in [c]. \tag{4.19}$$

Let $\kappa \coloneqq \beta/\alpha$. Note that

$$\prod_{i=1}^{c}\left(1 - \frac{\lambda}{\lambda_i}\right) \leq 1, \qquad \forall \lambda \in [\alpha,\beta]\cup\{0\}.$$

Hence, we have

$$q_r(\lambda) \leq Q_{\alpha,\beta,r}(\lambda), \qquad \forall \lambda \in [\alpha,\beta]\cup\{0\}. \tag{4.20}$$

Therefore, if $r \coloneqq \frac{\sqrt{\kappa}}{2}\ln 2/\varepsilon$, then $q_r \in \mathcal{Q}_{r+c}$. Hence, by Theorem 4.11, for every $t \geq r + c$ we have

$$\begin{aligned}\|x_t - x^*\|_A^2 &\leq \|x^*\|_A^2 \min_{q\in\mathcal{Q}_t} \max_{i\in[\operatorname{rank}(A)]} q(\lambda_i^{\downarrow}(A))^2 \overset{(4.19)}{\leq} \|x^*\|_A^2 \min_{q\in\mathcal{Q}_t}\max_{\lambda\in[\alpha,\beta]} q(\lambda)^2 \\ &\leq \|x^*\|_A^2 \max_{\lambda\in[\alpha,\beta]} |q_r(\lambda)|^2 \overset{(4.20)}{\leq} \|x^*\|_A^2 \max_{\lambda\in[\alpha,\beta]} Q_{\alpha,\beta,r}(\lambda)^2 \\ &\overset{\text{Le. 4.15}}{\leq} 2\|x^*\|_A^2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^{2r} \leq 2\|x^*\|_A^2\left(1 - \frac{1}{\sqrt{\kappa}}\right)^{2r} \\ &\overset{\text{Le. 2.2}}{\leq} 2\|x^*\|_A^2 \exp\left(-\frac{2r}{\sqrt{\kappa}}\right) \leq \epsilon\|x^*\|_A^2. \qquad \square\end{aligned}$$

Theorem 4.18. Let $A \in \mathbb{S}_{+}^{n}$ such that $\lambda_{\operatorname{rank}(A)}^{\downarrow}(A) \geq 1$ and $\operatorname{Tr}(A) \leq \tau \in \mathbb{R}_{++}$. Let $b \in \operatorname{Im}(A)$, let $\varepsilon > 0$, and let $x^* = A^{\dagger}b$. Let f be defined as in (4.2) and define

$$x_t \coloneqq \operatorname*{arg\,min}_{x\in\mathcal{K}_t(A,b)} f(x) \tag{4.21}$$

for $t \in \mathbb{N}$. Then

$$\|x_t - x^*\|_A^2 \leq \varepsilon \|x^*\|_A^2$$

for every $t \geq \tau^{1/3}(1 + \ln 1/\varepsilon)$.

Proof. By Theorem 2.6, we know that $\mathrm{Tr}(A)$ is the sum of the eigenvalues of A. Therefore, it is easy to see that for any $\beta \in \mathbb{R}_{++}$, the number of eigenvalues that are greater than β is at most $\mathrm{Tr}(A)/\beta$. Hence, if we set $\beta \coloneqq \tau^{2/3}$, at most $\mathrm{Tr}(A)^{1/3}$ of the eigenvalues of A will be outside the range $[1, \tau^{2/3}]$. Let c be the number of eigenvalues of A that are not in the range $[1, \tau^{2/3}]$. Hence, by Corollary 4.17, we have $\|x_t - x^*\|_A^2 \leq \varepsilon \|x^*\|_A^2$ for $t \geq c + \sqrt{\tau^{2/3}} \ln 1/\varepsilon$. Since $c \leq \mathrm{Tr}(A)^{1/3} \leq \tau^{1/3}$, the result follows. ☐

Chapter 5

Fast Laplacian Solvers

Many interesting algorithms for graph problems, including the algorithm for the maximum flow problem that we study in Chapter 6, use a solver for a Laplacian system as a subroutine. Spielman and Teng described in seminal work [11, 12, 13, 14] the first nearly-linear time solver for Laplacian systems. We state their result in the following theorem.

Theorem 5.1. There is an algorithm that takes as input

- a weighted connected graph $G = (V, E, w)$;
- a vector $b \in \mathbb{R}^V$ such that $b \perp \mathbb{1}$;
- a value $\varepsilon > 0$,

and computes as output $x \in \mathbb{R}^V$ such that

$$\left\|x - L^{\dagger}b\right\|_L \leq \varepsilon \|L^{\dagger}b\|_L,$$

where $L \coloneqq \mathcal{L}_G(w)$. This algorithm runs in time $\tilde{O}(m \log(1/\varepsilon))$, where $m \coloneqq |E|$.

The algorithm constructed by Spielman and Teng is intricate, makes use of complex graph-theoretic structures, and the power of $\log n$ hidden by the soft-O notation is quite large, but their solver opened the floodgates. Following their work, many authors were able to simplify and improve the algorithm of Theorem 5.1 (see [4, 8, 9]), and research for simpler Laplacian solvers is still active. One recent development in the area is due to Kyng and Sachdeva [10], who constructed a simple nearly-linear time Laplacian solver based purely on random sampling, not depending on any graph-theoretic construction. Moreover, fast Laplacian solvers have been used in the development of very efficient algorithms for a host of combinatorial problems (see [15]).

Although we shall not prove Theorem 5.1, in this chapter we describe a $\tilde{O}(m^{4/3} \log(1/\varepsilon))$ Laplacian solver. This solver already has quite a respectable running time, and its construction contains many ideas used in the solver of Spielman and Teng.

5.1 Preconditioning

Let $A \in \mathbb{R}^{m \times n}$ and let $b \in \operatorname{Im}(A)$. Usually, the time it takes to solve the system $Ax = b$ with iterative methods depends on some properties of the matrix A such as its condition number. The idea of preconditioning is to build a matrix $M \in \mathbb{R}^{m \times m}$ such that applying an iterative method to the system $MAx = Mb$ is considerably faster than applying the same method to the original system. Moreover, constructing a solution to the system $Ax = b$ from a solution to the system $MAx = Mb$ should be efficient. We call the matrix M a preconditioner of the system $Ax = b$. In this section, we will focus on preconditioners that decrease the condition number of the matrix.

Note that $A^{\dagger}$ is an excellent preconditioner for $Ax = b$, but computing it exactly boils down to solving the system itself. Moreover, the main operation that depends on A in many iterative algorithms, such as the Conjugate Gradient method or the Power Method, is left-multiplying A by a vector in its image. Hence, a preconditioner M such that it is time-consuming to left-multiply it by a vector may make an iterative algorithm slower. Therefore, when choosing a preconditioner of a linear system, there are trade-offs involving the time it takes to compute it, the time it takes to apply it to a vector (by left-multiplying it), and the decrease it yields on the condition number of the system.

A problem that one may be concerned is that, if $A \in \mathbb{S}^n$, preconditioning the system $Ax = b$ may not preserve the symmetry of the matrix A. This is a problem in some cases, as in the Conjugate Gradient method, which depends in a fundamental way on the symmetry of the matrix A. In this case, we may choose a preconditioner $M \in \mathbb{S}^n_+$. This implies, by Theorem 2.13, that there exists $E \in \mathbb{R}^{n\times n}$ such that $M = EE^T$. In this case, we will precondition the system $Ax = b$ by considering the system $EAE^T = b$. Hence, we will first show that we can obtain an approximate solution to the system $Ax = b$ from an approximate solution to $EAE^T = b$ when some conditions are met. In the next section, we will show how this preconditioning affects the condition number of the matrix, specially when considering the case of preconditioning a Laplacian system.

Lemma 5.2. Let $A, B \in \mathbb{S}^n_+$ be such that $\operatorname{Null}(A) = \operatorname{Null}(B)$, and let $E \in \mathbb{R}^{n\times n}$ be such that $B = EE^T$. Define $W := E^\dagger A E^{T\dagger}$. Then $\operatorname{Im}(E) = \operatorname{Im}(B)$, and $\operatorname{Im}(E^T) = \operatorname{Im}(W)$.

Proof. First, let us prove that $\operatorname{Im}(E) = \operatorname{Im}(B)$. Let $x \in \mathbb{R}^n$. Note that

$$x \in \operatorname{Null}(B) \overset{\text{Prop. 2.12}}{\iff} x^T B x = 0 \iff x^T E E^T x = 0 \iff \|E^T x\|^2 = 0 \iff x \in \operatorname{Null}(E^T).$$

Hence, $\operatorname{Null}(B) = \operatorname{Null}(E^T)$, and by Theorem (2.1), we have $\operatorname{Im}(B) = \operatorname{Im}(E)$. This yields

$$\operatorname{Im}(E^{T\dagger}) \overset{\text{Prop. 2.25}}{=} \operatorname{Im}(E) = \operatorname{Im}(B) = \operatorname{Im}(A). \tag{5.1}$$

Let us now prove that $\operatorname{Null}(W) = \operatorname{Null}(E)$. Let $x \in \mathbb{R}^n$. Since $A \succeq 0$, we have $W \succeq 0$. Hence,

$$x \in \operatorname{Null}(W) \overset{\text{Prop. 2.12}}{\iff} x^T W x = 0 \iff x^T E^\dagger A E^{T\dagger} x = 0 \overset{\text{Prop. 2.12}}{\iff} E^{T\dagger} x \in \operatorname{Null}(A).$$

By (5.1), we have $E^{T\dagger} x \in \operatorname{Im}(A)$. Therefore, $E^{T\dagger} x \in \operatorname{Null}(A) \cap \operatorname{Im}(A) = \{0\}$. This is the case if and only if $x \in \operatorname{Null}(E^{T\dagger})$, and by Proposition 2.25 we have $x \in \operatorname{Null}(E)$. Hence, $\operatorname{Null}(W) = \operatorname{Null}(E)$. Theorem 2.1 implies that $\operatorname{Im}(W) = \operatorname{Im}(E^T)$. ☐

Theorem 5.3. Let $A, B \in \mathbb{S}^n_+$ be such that $\operatorname{Null}(A) = \operatorname{Null}(B)$. Let $E \in \mathbb{R}^{n\times n}$ be such that $B = EE^T$, and define $W := E^\dagger A E^{T\dagger}$. Define $\phi\colon y \in \operatorname{Im}(W) \mapsto E^{T\dagger} y \in \operatorname{Im}(A)$. Then

$$\langle x, y\rangle_W = \langle \phi(x), \phi(y)\rangle_A, \qquad \forall x, y \in \operatorname{Im}(W).$$

In particular, let $b \in \operatorname{Im}(A)$ and let $\varepsilon > 0$. If, for each $y \in \mathbb{R}^n$,

$$\|y^* - y\|_W \le \varepsilon \|y^*\|_W,$$

where $y^* := W^\dagger E^\dagger b$, then

$$\|x^* - \phi(y)\|_A \le \epsilon \|x^*\|_A,$$

where $x^* := A^\dagger b$.

Proof. By Theorem 2.1, since $\operatorname{Null}(A) = \operatorname{Null}(B)$, we have $\operatorname{Im}(B) = \operatorname{Im}(A)$. By Proposition 2.25, we have $\operatorname{Im}(E^{T\dagger}) = \operatorname{Im}(E)$, and by Lemma 5.2 we have $\operatorname{Im}(E) = \operatorname{Im}(B) = \operatorname{Im}(A)$. Therefore,

$$\operatorname{Im}(E) = \operatorname{Im}(B) = \operatorname{Im}(A). \tag{5.2}$$

Let $x, y \in \operatorname{Im}(W)$. We have

$$\langle \phi(x), \phi(y)\rangle_A = \phi(x)^T A \phi(y) = x^T E^\dagger A E^{T\dagger} y = x^T W y = \langle x, y\rangle_W.$$

In particular,

$$\|y^* - y\|_W^2 \le \epsilon \|y^*\|_W^2 \iff \|\phi(y^*) - \phi(y)\|_A^2 = \|\phi(y^* - y)\|_A^2 \le \epsilon \|\phi(y^*)\|_A^2.$$

Therefore, it only remains to show that $\phi(y^*) = x^*$. By Proposition 2.23 and by (5.2),

$$AA^\dagger = EE^\dagger = \mathrm{Proj}_{\mathrm{Im}(A)}. \tag{5.3}$$

Moreover, by Propositions 2.23 and 2.25,

$$W^\dagger W \text{ is an orthogonal projector onto } \mathrm{Im}(W^\dagger) = \mathrm{Im}(W) = \mathrm{Im}(E^\dagger). \tag{5.4}$$

Therefore,

$$\begin{aligned}\phi(y^*) &= E^{T\dagger} y^* = E^{T\dagger} W^\dagger E^\dagger b \\ &\overset{(5.3)}{=} E^{T\dagger} W^\dagger E^\dagger A A^\dagger b \overset{(5.3)}{=} E^{T\dagger} W^\dagger E^\dagger A (EE^\dagger)^T A^\dagger b \\ &= E^{T\dagger} W^\dagger E^\dagger A E^{T\dagger} E^T A^\dagger b = E^{T\dagger} W^\dagger W E^T A^\dagger b \\ &\overset{(5.4)}{=} E^{T\dagger} E^T A^\dagger b = (EE^\dagger)^T A^\dagger b \\ &\overset{(5.3)}{=} A^\dagger b = x^*.\end{aligned}$$

□

5.2 A Fast Solver

For every $A \in \mathbb{S}^V$ we define the weighted graph $G(A) \coloneqq (V, E, w)$, where

$$E \coloneqq \left\{ ij \in \binom{V}{2} : A_{i,j} \neq 0 \right\},$$

and $(w)_{ij} \coloneqq A_{i,j}$ for every $ij \in E$. A **permutation matrix** is an orthogonal matrix with entries in $\{0,1\}$. Given a bijection $\sigma \colon V \to V$, define the permutation matrix $P_\sigma \in \{0,1\}^{V \times V}$ by

$$P_\sigma e_i \coloneqq e_{\sigma(i)}, \qquad \forall i \in V.$$

Proposition 5.4. If $A \in \mathbb{S}^n_+$ is such that $G(A)$ is a tree, then there are a lower triangular matrix $L \in \mathbb{R}^{n \times n}$ with at most $2n$ non-zero entries and a bijection $\sigma \colon [n] \to [n]$ such that $P_\sigma A P_\sigma^T = LL^T$. Moreover, we can compute the matrix L and a the bijection $\sigma \colon [n] \to [n]$ in time $O(m)$, where $m \coloneqq |E(G(A))|$, using the data structures from Section 2.2.

Proof. Let $\sigma \colon [n] \to [n]$ be a permutation such that, for each $i \in [n]$, $\sigma(i)$ is a leaf in $G - \{\sigma(1), \dots, \sigma(i-1)\}$. Such a permutation can be found easily in linear time using depth-first search. Hence, each $i \in [n]$ is a leaf in $G\big((P_\sigma A P_\sigma^T)[[n] \setminus [i-1]]\big)$. For simplicity, we may assume that $P_\sigma = I$. Let $L \in \mathbb{R}^{n \times n}$ be a lower triangular matrix and let $D \in \mathbb{S}^n_+$ be a diagonal as in Theorem 2.18 such that $A = LDL^T$. Since $A = \tilde{L}\tilde{L}^T$, where $\tilde{L} \coloneqq (LD^{1/2})$ is lower triangular and has the same number of nonzero elements as L, it suffices to prove that

$$\text{the matrix } L \text{ has } n-1 \text{ nonzero off-diagonal elements and can be computed in } O(n) \text{ time.} \tag{5.5}$$

Let us first prove that L has $n-1$ nonzero off-diagonal entries by induction on n. For $n = 1$ the statement clearly holds. Suppose that $n > 1$. Let $\alpha \in \mathbb{R}_+$, let $i \in N_{G(A)}(1) \subseteq [n] \setminus \{1\}$ be the only neighbor of 1, and let $B \in \mathbb{S}^{[n]\setminus\{1\}}_+$ be such that

$$A = \begin{pmatrix} \alpha & e_i^T \\ e_i & B \end{pmatrix}.$$

Let $\tilde{B} \coloneqq B - [\alpha > 0]\frac{1}{\alpha} e_i e_i^T \in \mathbb{R}^{[n]\setminus\{1\}}$. Note that $\tilde{B}$ can be computed in constant time, since $e_i e_i^T$ has only one nonzero entry. By Theorem 2.18,

$$L = \begin{pmatrix} [\alpha > 0] & 0 \\ [\alpha > 0]\frac{1}{\alpha} e_i & \tilde{L} \end{pmatrix}, \tag{5.6}$$

where $\tilde{L} \in \mathbb{R}^{([n]\setminus\{1\}) \times ([n]\setminus\{1\})}$ is a lower triangular matrix and $\tilde{D} \in \mathbb{S}^{[n]\setminus\{1\}}_+$ is a diagonal matrix such that $\tilde{B} = \tilde{L}\tilde{D}\tilde{L}^T$. Note that $G(\tilde{B}) = G(A) - 1$, and since 1 is a leaf in $G(A)$, we have that $G(\tilde{B})$ is a tree.

Moreover, $E(G(\tilde{B})) = E(G(A)) \setminus \{1i\}$. Hence, the property that $j \in [n] \setminus \{1\}$ is a leaf in $G(\tilde{B})[[n-1] \setminus [j]]$ is preserved. Therefore, by the induction hypothesis, the matrix $\tilde{L}$ has $n-2$ nonzero off-diagonal elements. Hence, by equation (5.6), we conclude that L has $n-1$ nonzero entries. It only remains to show that the matrix L can be computed in $O(n)$ time.

We can compute L recursively by the definitions in Theorem 2.18. Since we can find $i \in N_G(1)$ and compute $\tilde{L}$ in constant time, each recursive call takes constant time to be computed. At each recursive call, the dimension of the matrix we have to process decreases by 1. Hence, we make a total of n recursive calls, and thus we can compute L in $O(n)$ time. ☐

Let $G = (V, E, w)$ be a weighted graph, and let T be a spanning tree of G. For every $ij \in E$, let $T(i, j)$ be the unique path between i and j in T. For every $e = ij \in E$, the **stretch** of e (with respect to T and w) is

$$\mathrm{st}_T(e) \coloneqq w_e \sum_{f \in E(T(i,j))} \frac{1}{w_f}.$$

We also define $\mathrm{st}_T(G) \coloneqq \sum_{e \in E} \mathrm{st}_T(e)$. We will use the following theorem without proof.

Theorem 5.5 ([1])**.** There is an algorithm that takes as input a connected weighted graph G, and computes as output a spanning tree T of G such that $\mathrm{st}_T(G) \in O(m \log n \log\log n (\log\log\log n)^3)$, where $n \coloneqq |V(G)|$ and $m \coloneqq |E(G)|$, and runs in time $\tilde{O}(m)$.

Lemma 5.6. If $A \in \mathbb{S}^n$ and $B \in \mathbb{R}^{n \times n}$ are such that $\mathrm{Null}(A) = \mathrm{Null}(B)$, then $\lambda^{\uparrow}(A) = \lambda^{\uparrow}(BAB^{\dagger})$.

Proof. By Theorem 2.1, we know that

$$\mathrm{Im}(A) = \mathrm{Null}(A)^{\perp} = \mathrm{Null}(B)^{\perp} = \mathrm{Im}(B^T).$$

Hence, $B^{\dagger}B$ is an orthogonal projector onto $\mathrm{Im}(A)$ by Proposition 2.23. Therefore, $B^{\dagger}BA = A$, and the statement follows from Lemma 2.4. ☐

Lemma 5.7. If $A, B \in \mathbb{S}^n_+$ and $A \succeq B$, then all nonzero eigenvalues of $B^{\dagger}A$ are at least 1.

Proof. Set $W \coloneqq (B^{\dagger})^{1/2} A (B^{\dagger})^{1/2}$. First, let us prove that

$$W \ \text{ has the same eigenvalues of } B^{\dagger}A. \tag{5.7}$$

By Lemma 5.6, the matrix W has the same eigenvalues of $(B^{\dagger})^{1/2} W B^{1/2}$. Note that

$$(B^{\dagger})^{1/2} W B^{1/2} = (B^{\dagger})^{1/2} (B^{\dagger})^{1/2} A (B^{\dagger})^{1/2} B^{1/2} = B^{\dagger} A (B^{1/2})^{\dagger} B^{1/2} = B^{\dagger} A,$$

where we used in the last equation Proposition 2.23, which states that $(B^{1/2})^{\dagger} B^{1/2}$ is the orthogonal projector onto $\mathrm{Im}(B^{1/2}) = \mathrm{Im}(B)$ and that

$$\mathrm{Im}(B) \subseteq \mathrm{Im}(A). \tag{5.8}$$

To see that the above claim holds, suppose there is $x \in \mathrm{Null}(A) \setminus \mathrm{Null}(B)$. Then, by Proposition 2.12, we have $x^T B x > 0$. Hence,

$$x^T B x > 0 = x^T A x,$$

what is a contradiction since $A \succeq B$. This ends the proof of (5.8), and therefore ends the proof of (5.7). Thus, it suffices to show that

$$\lambda^{\downarrow}_{\mathrm{rank}(W)}(W) \geq 1.$$

By Theorem 2.22, one can verify that $(B^{\dagger})^{1/2} B^{1/2} = (B^{\dagger}B)^{1/2}$. Hence, using that $A \succeq B$, we have

$$W = (B^{\dagger})^{1/2} A (B^{\dagger})^{1/2} \succeq (B^{\dagger})^{1/2} B (B^{\dagger})^{1/2} = (B^{\dagger})^{1/2} B^{1/2} B^{1/2} (B^{\dagger})^{1/2} = (B^{\dagger}B)^{1/2} (BB^{\dagger})^{1/2} = BB^{\dagger},$$

where in the last equation we used that

$$BB^{\dagger} = (BB^{\dagger})^T = B^{T\dagger} B^T = B^{\dagger} B,$$

where in the first equation we used property (iii) of the pseudoinverse, and in the last equation we used that B is symmetric. Let $x \in \operatorname{Im}(W)$ be an eigenvector of W associated with $\lambda(W)^{\downarrow}_{\operatorname{rank}(W)}$. By Proposition 2.25, $\operatorname{Im}(B^\dagger) = \operatorname{Im}(B)$. Since $W = (B^\dagger)^{1/2} A (B^\dagger)^{1/2}$, we have $\operatorname{Im}(W) \subseteq \operatorname{Im}(B^\dagger)$. By Proposition 2.25, $\operatorname{Im}(B^\dagger) = \operatorname{Im}(B)$. Hence, $\operatorname{Proj}_{\operatorname{Im}(B)} x = x$. By Proposition 2.23, we have that $BB^\dagger = \operatorname{Proj}_{\operatorname{Im}(B)}$. Therefore,

$$\lambda \|x\|^2 = x^T W x \geq x^T B B^\dagger x == x^T \operatorname{Proj}_{\operatorname{Im}(B)} x = \|x\|^2 \implies \lambda \geq 1. \qquad \square$$

Lemma 5.8. Let T be a weighted tree with weights $r \in \mathbb{R}^E_{++}$. If $s, t \in V(T)$ are distinct, and P is the unique (s,t)-path in T, then

$$R^{s,t}_{\text{eff}}(r) = \sum_{e \in E(P)} r_e.$$

Proof. Let f be the unit electrical (s,t)-flow in T. By Theorem 3.7, we have $R^{s,t}_{\text{eff}}(r) = \mathcal{E}(f)$. By Proposition 3.4, we can write $\operatorname{Diag}(\operatorname{sgn}(f))f$ as a linear combination of incidence vectors of (s,t)-paths. Since P is the unique (s,t)-path in T, we have $\operatorname{Diag}(\operatorname{sgn}(f))f = |f|\mathbb{1}_{E(P)} = \mathbb{1}_{E(P)}$. Then,

$$\begin{aligned} R^{s,t}_{\text{eff}}(r) &= \mathcal{E}(f) = f^T \operatorname{Diag}(r) f = \operatorname{Diag}(\operatorname{sgn}(f))^2 f^T \operatorname{Diag}(r) f \\ &= (\operatorname{Diag}(f) f)^T \operatorname{Diag}(r) \operatorname{Diag}(f) f = \mathbb{1}^T_{E(P)} \operatorname{Diag}(r) \mathbb{1}_{E(P)} \\ &= \sum_{e \in E(P)} r_e. \end{aligned} \qquad \square$$

Proposition 5.9. If G is a weighted graph with weights $w \in \mathbb{R}^E_{++}$, and T is a spanning tree of G, then

$$\operatorname{Tr}\big(\mathcal{L}_T(w)^\dagger \mathcal{L}_G(w)\big) = \operatorname{st}_T(G).$$

Proof. We have

$$\operatorname{Tr}\big(\mathcal{L}_T(w)^\dagger \mathcal{L}_G(w)\big) = \sum_{ij \in E(G)} w_{ij} \operatorname{Tr}\big(\mathcal{L}_T(w)^\dagger (e_i - e_j)(e_i - e_j)^T\big) = \sum_{ij \in E(G)} w_{ij} (e_i - e_j)^T \mathcal{L}_T(w)^\dagger (e_i - e_j).$$

Note that $(e_i - e_j)^T \mathcal{L}_T(w)^\dagger (e_i - e_j)$ is, by definition, the effective resistance of ij in T with edge weights $r \in \mathbb{R}^E$ such that $r_e := 1/w_e$ for each $e \in E$. For every $ij \in E(G)$, let $T(i,j)$ be the unique (i,j)-path in T. Then, by Lemma 5.8,

$$\operatorname{Tr}\big(\mathcal{L}_T(w)^\dagger \mathcal{L}_G(w)\big) = \sum_{ij \in E(G)} w_{ij} (e_i - e_j)^T \mathcal{L}_T(w)^\dagger (e_i - e_j) = \sum_{ij \in E(G)} \left(w_{ij} \sum_{e \in E(T(i,j))} \frac{1}{w_e} \right) = \operatorname{st}_T(G). \qquad \square$$

Theorem 5.10. There is an algorithm that takes as input

- a weighted connected graph $G = (V, E, w)$;
- a vector $b \in \mathbb{R}^V$ such that $b \perp \mathbb{1}$;
- a value $\varepsilon > 0$,

and computes as output $x \in \mathbb{R}^V$ such that

$$\big\| x - L^\dagger b \big\|_L \leq \varepsilon \| L^\dagger b \|_L,$$

where $L := \mathcal{L}_G(w)$. This algorithm runs in time $\tilde{O}(m^{4/3} \log(1/\varepsilon))$ using the data structures from Section 2.2, where $m := |E|$.

Proof. Set $n := |V|$ and let T be a spanning tree of G such that $\operatorname{st}_T(G) \in \tilde{O}(m)$. Such a spanning tree exists and can be computed in $\tilde{O}(m)$ time by Theorem 5.5. By Proposition 5.4, there are a lower triangular matrix $E \in \mathbb{R}^{V \times V}$ with at most $2n$ nonzero off-diagonal entries and a bijection $\sigma \colon V \to V$ such that $P_\sigma \mathcal{L}_T(w) P_\sigma^T = EE^T$. Moreover, this proposition states that the matrix E and the permutation that corresponds to the action of P can computed in $O(m)$ time. Since we can fix any permutation of the vertices, suppose that $P = I$. Define $W \coloneqq E^\dagger \mathcal{L}_G(w) E^{T\dagger}$. The idea now is to approximately solve the system $Wy = E^\dagger b$ using the Conjugate Gradient method. Before invoking this method, we need to show that

$$\lambda^{\downarrow}_{\mathrm{rank}(W)}(W) \geq 1 \text{ and } \mathrm{Tr}(W) = \mathrm{st}_T(G). \tag{5.9}$$

By Lemma 5.6, we have that the eigenvalues of W are the same of $E^\dagger W E = \mathcal{L}_T(w)^\dagger \mathcal{L}_G(w)$, and by Lemma 5.7, we have that all nonzero eigenvalues of $\mathcal{L}_T(w)^\dagger \mathcal{L}_G(w)$ are at least 1. Hence, $\lambda^{\downarrow}_{\mathrm{rank}(W)}(W) \geq 1$. Moreover, note that

$$\mathrm{Tr}(W) = \mathrm{Tr}(E^\dagger \mathcal{L}_G(w) E^{T\dagger}) = \mathrm{Tr}(E^{T\dagger} E^\dagger \mathcal{L}_G(w)) \overset{\text{Prop. 2.21}}{=} \mathrm{Tr}((EE^T)^\dagger \mathcal{L}_G(w)) = \mathrm{Tr}(\mathcal{L}_T(w)^\dagger \mathcal{L}_G(w)) = \mathrm{st}_T(G),$$

where in the last equation we used Proposition 5.9. This ends the proof of (5.9).

Hence, by Proposition 4.8 and by Theorem 4.18, after $\omega \coloneqq \mathrm{st}_T(G)^{1/3}(1+\ln 1/\varepsilon)$ iterations of the Conjugate Gradient method on the system $Wy = E^\dagger b$, it yields $y \in \mathbb{R}^V$ such that

$$\|y^* - y\|_W \leq \varepsilon \|y^*\|_W,$$

where $y^* \coloneqq W^\dagger E^\dagger b$. Define $x \coloneqq E^{T\dagger} y$. By Theorem 5.3, we have that

$$\|x^* - x\|_{\mathcal{L}_G(w)} \leq \varepsilon \|x^*\|_{\mathcal{L}_G(w)},$$

where $x^* \coloneqq \mathcal{L}_G(w)^\dagger b$.

It only remains to show that

the execution of the Conjugate Gradient method (CGM) to find a solution to the system $Wy = E^\dagger b$ takes time $\tilde{O}(m^{4/3} \log 1/\varepsilon)$. (5.10)

In each iteration, the CGM computes a constant number of matrix-vector multiplications of W with a vector in $\mathcal{K}_t(W, E^\dagger b) \subseteq \mathrm{Im}(W)$ for some $t \in \mathbb{N}$. Therefore, let us first show that

it takes time $O(m)$ to compute Wv for any $v \in \mathrm{Im}(W)$. (5.11)

Since $E^\dagger \mathcal{L}_G(w) E^{T\dagger}$, left-multiplying a vector $v \in \mathrm{Im}(W)$ by W can be broken down into three steps. First, one needs to left-multiply v by $E^{T\dagger}$. By Lemma 5.2, we have that $\mathrm{Im}(W) = \mathrm{Im}(E^T)$. Therefore,

$$r \coloneqq E^{T\dagger} p \implies E^T r = E^T E^{T\dagger} p = p,$$

where in the last equation we used that $E^T E^{T\dagger}$ is an orthogonal projector onto $\mathrm{Im}(E^T)$. Hence, to calculate r it suffices to solve a linear system on E^T. Since E has at most $2n$ nonzero entries and is lower triangular, solving this system takes $O(n)$ time. Next, to compute $\mathcal{L}_G(w)(E^{T\dagger} v)$ we need time $O(m)$. This is due to the fact that $\mathcal{L}_G(w)$ is a sparse matrix with $O(m)$ nonzero entries. Finally, to calculate $E^\dagger(\mathcal{L}_G(w) E^T v)$, by Lemma 5.2, we have that $\mathrm{Im}(\mathcal{L}_G(w)) = \mathrm{Im}(\mathcal{L}_T(w)) = \mathrm{Im}(E)$. Since $\mathcal{L}_G(w) E^{T\dagger} v \in \mathrm{Im}(\mathcal{L}_G(w)) = \mathrm{Im}(E)$, we only have to solve a linear system over the matrix E. Since E has at most $2n$ nonzero entries and is lower triangular, solving this system takes $O(n)$ time. This ends the proof of (5.11).

Hence, by (5.11), each iteration of the CGM takes time $O(m)$. Since we execute $\omega = \mathrm{st}_T(G)^{1/3}(1 + \ln 1/\varepsilon)$ iterations of the CGM, and using the fact that $\mathrm{st}_T(G) \in \tilde{O}(m)$, we have that the CGM, in this case, takes time $O(m\omega) = \tilde{O}(m^{4/3} \log 1/\varepsilon)$. This ends the proof of (5.10), and the theorem follows. □

Chapter 6

Maximum Flow in Graphs using Electrical Flows

In this chapter we describe the algorithm from [3], which computes an approximately maximum flow in a graph in a quite respectable running time with the aid of nearly-linear time Laplacian solvers. Not only that, but its general idea is also is relatively simple. Intuitively, it uses electrical flows, which do not necessarily satisfy the capacity constraints, as approximations to flows of some desired value that respect the edge capacities. The basic idea is to first compute an electrical flow with some initial resistances on the edges of the graph. Since an electrical flow may not respect the edge capacities, we modify the resistances over the edges, penalizing edges on which there is too much flow compared to their capacities, and then repeat the process. We compute these multiple electrical flows and combine them with the aid of the Multiplicative Weights Update Method [2]. Intuitively, this method is a meta-algorithm that takes an algorithm which solves a given problem very crudely and, by repeatedly calling this crude algorithm with new parameters, it computes a good approximate solution to the problem. After computing sufficiently many electrical flows, we will be able to compute a feasible flow whose value is close to the desired one. Since computing electrical flows exactly is costly, we only compute electrical flows approximately.

In Section 6.1, we study how to compute an approximately electrical flow, that is, a flow that has almost minimum energy, in nearly-linear time by using a nearly-linear time Laplacian solver. In Section 6.2, we describe how to use approximately electrical flows as crude approximations to feasible (s,t)-flows of a target value $\alpha \in \mathbb{R}_+$, and use such approximately electrical flows in the Multiplicative Weights Update Method to compute a flow of value close to α in time $\tilde{O}(m^{3/2}\varepsilon^{-5/2})$. In Section 6.3, we show how to compute an approximately maximum flow via binary search using the algorithm from the preceding section as a subroutine.

6.1 Computing Approximately Electrical Flows

Recall from Section 3.2 that one can find an electrical (s,t)-flow in a graph G by solving a Laplacian system, which yields the vertex potentials of the unit electrical flow of the graph. By Theorem 5.1, we can find an approximate solution to a Laplacian system in nearly-linear time, and hence we may approximately find these vertex potentials in nearly-linear time. The problem that arises is that the vector induced by this approximate solution may not be an (s,t)-flow, i.e. it need not satisfy the flow conservation constraints. In this section, we will show how to round in nearly-linear time the vector induced by the approximate vertex potentials to an (s,t)-flow with almost minimum energy.

Let $G = (V, E, r)$ be a weighted graph. Let $s, t \in V$ be distinct and let $f^* \in \mathbb{R}^E$ be the unit electrical (s,t)-flow in G (see Section 3.2 for the definition of electrical flow). Let $\delta \in \mathbb{R}_+$. An (s,t)-flow f in G is **δ-approximately electrical** if $\mathcal{E}(f) \leq (1+\delta)\mathcal{E}(|f|f^*)$.

In the next lemma we describe an algorithm that rounds a given vector to a flow in the graph. This algorithm will be used in the main theorem of this section to round the non-flow induced by the vertex potentials that forms an approximate solution of a Laplacian system.

Lemma 6.1. There is an algorithm that takes as input

- a weighted connected graph $G = (V, E, r)$,
- distinct vertices $s, t \in V$,
- a target flow value $\alpha \in \mathbb{R}_+$,
- an orientation $\vec{G}$ of G,
- a vector $\hat{f} \in \mathbb{R}^E_+$,

and computes as output an (s,t)-flow f of G with respect to $\vec{G}$ of value $\alpha \in \mathbb{R}_+$ such that

$$\|\hat{f} - f\|_\infty \leq n\|i_{\text{ext}} - \alpha(e_s - e_t)\|_\infty,$$

where $n \coloneqq |V|$ and

$$i_{\text{ext}} \coloneqq B_{\vec{G}}\hat{f}.$$

Moreover, this algorithm runs in time $O(m)$ when using the data structures from Section 2.2, where $m \coloneqq |E|$.

Proof. Let us show how to compute $\Delta f \in \mathbb{R}^E$ such that

$$B_{\vec{G}}\Delta f = \alpha(e_s - e_t) - i_{\text{ext}} \eqqcolon d,$$

so that $f \coloneqq \hat{f} + \Delta f$ satisfies $B_{\vec{G}} f = \alpha(e_s - e_t)$. Note that

$$\mathbb{1}^T d = \alpha\mathbb{1}^T(e_s - e_t) - \mathbb{1}^T i_{\text{ext}} = \alpha\mathbb{1}^T(e_s - e_t) - \mathbb{1}^T B_{\vec{G}}\hat{f} = 0,$$

where in the last equation we used that $\mathbb{1} \in \text{Null}(B^T_{\vec{G}})$ by Proposition 2.30. Let $T = (V, F)$ be a spanning tree of G and let $\vec{T}$ be an orientation of T such that $A(\vec{T}) \subseteq A(\vec{G})$. Such a spanning tree can be found in $O(m)$ by a depth-first search. Actually,

> one can easily compute a spanning tree T of G and a function $\psi : [n] \to V$ in $O(m)$ time such that $\psi(i)$ is a leaf in the tree $T[\{\psi(i), \psi(i+1), \ldots, \psi(n)\}]$ by using depth-first search. (6.1)

If there is $g \in \mathbb{R}^F$ such that $B_{\vec{T}} g = d$, then defining $\Delta f \in \mathbb{R}^E$ by $\Delta f_e \coloneqq [e \in F] g_e$ for every $e \in E$ yields $B_{\vec{G}}\Delta f = B_{\vec{T}} g = d$. Hence, it suffices to prove the following claim:

> Let $T = (V, F)$ be a spanning tree of G with orientation $\vec{T}$ and let $d \in \mathbb{R}^V$ be such that $\mathbb{1}^T d = 0$. Then we can compute $g \in \mathbb{R}^F$ such that $B_{\vec{T}} g = d$ in time $O(n)$. (6.2)

First of all, note that

If $n = 1$, then $d = \mathbb{1}^T d = 0$ and $g = 0$ satisfies the claim. Suppose that $n > 1$. Let $v \in V$ be a leaf in T, let $\{u\} \coloneqq N_T(v)$ and define $V' \coloneqq V \setminus \{v\}$. Let $T' \coloneqq T - v$ and let $\vec{T'} \coloneqq \vec{T} - v$. Define $d' \in \mathbb{R}^{V'}$ by setting $d'_i \coloneqq d_i + [i = u] d_v$ for each $i \in V'$. Hence,

$$d + d_v(e_u - e_v) = \begin{pmatrix} 0 \\ d' \end{pmatrix} \in \mathbb{R}^V. \tag{6.3}$$

Note that

$$\mathbb{1}^T d' = \mathbb{1}^T \begin{pmatrix} 0 \\ d' \end{pmatrix} = \mathbb{1}^T d + \mathbb{1}^T(e_u - e_v) d_v = 0$$

Hence, $d' \perp \mathbb{1}$, and we can recursively compute $g' \in \mathbb{R}^{E(T')}_+$ such that $B_{\vec{T'}} g' = d'$. Let $a \in A(\vec{T})$ be the only arc incident to v in $\vec{T}$. Extend $g' \in \mathbb{R}^{E(T')}_+$ to $g \in \mathbb{R}^{E(T)}_+$ by setting

$$g_{uv} \coloneqq (-1)^{[a \in \delta^{\text{in}}(v)]} d_v. \tag{6.4}$$

Therefore,

$$\begin{aligned}
B_{\vec{T}} g &= B_{\vec{T}} \begin{pmatrix} (-1)^{[a\in\delta^{\text{in}}(v)]} d_v \\ g' \end{pmatrix} \\
&= B_{\vec{T}} e_a (-1)^{[a\in\delta^{\text{in}}(v)]} d_v + \begin{pmatrix} 0 \\ B_{\vec{T'}} g' \end{pmatrix} \\
&= (-1)^{[a\in\delta^{\text{in}}(v)]}(e_v - e_u)(-1)^{[a\in\delta^{\text{in}}(v)]} d_v + \begin{pmatrix} 0 \\ d' \end{pmatrix} \\
&= (e_v - e_u) d_v + \begin{pmatrix} 0 \\ d' \end{pmatrix} \overset{(6.3)}{=} d.
\end{aligned}$$

One may note that it is possible to perform the calculations from (6.4) during the depth first search from (6.1) (more specifically, one may perform the calculation from (6.4) when ending visiting a vertex). Hence, the algorithm takes a constant amount of computation for each node during the depth-first search. Hence, we conclude the the algorithm runs in time $O(m)$. ☐

We will now prove some bounds on the energy of an electrical flow, which will be useful in the proof of the main theorem of this section.

Lemma 6.2. Let $G = (V, E, r)$ be a weighted connected graph such that $r_e \in [1, \omega]$ for each $e \in E$, set $m := |E|$, and let $s, t \in V$ be distinct. If f is the electrical (s,t)-flow in G of value $\alpha \in \mathbb{R}_+$, then

$$\frac{\alpha^2}{m} \le \mathcal{E}(f) \le \alpha^2 \omega m.$$

Proof. Let us first prove that

$$\mathcal{E}(f) \le \alpha^2 \omega m. \tag{6.5}$$

Note that

$$\mathcal{E}(f) = \sum_{e\in E} f_e^2 r_e \le \omega \sum_{e\in E} f_e^2 \le \|f\|_\infty^2 \omega m.$$

Hence, to prove (6.5), it suffices to prove that

$$\|f\|_\infty \le \alpha. \tag{6.6}$$

By Proposition 3.5, there is a collection $\mathcal{P}$ of (s,t)-paths in G with $|\mathcal{P}| \le |E|$ and a vector $c \in \mathbb{R}_+^{\mathcal{P}}$ with $\|c\|_1 = \alpha$ such that

$$f = \text{Diag}(\text{sgn}(f)) \sum_{P\in\mathcal{P}} c(P) \mathbb{1}_{E(P)}.$$

Hence, for every $e \in E$,

$$|f_e| = \sum_{P\in\mathcal{P}} [e \in E(P)] c(P) \le \|c\|_1 = \alpha,$$

i.e., $\|f\|_\infty \le \alpha$. This ends the proof of (6.6), and thus that of (6.5).

Let us now prove that

$$\frac{\alpha^2}{m} \le \mathcal{E}(f). \tag{6.7}$$

Let f_1 be the unit electrical (s,t)-flow in G. By Theorem 3.6, $\alpha f_1 = f$, and hence $\mathcal{E}(f) = \alpha^2 \mathcal{E}(f_1)$. Moreover, by Proposition 3.7, $\mathcal{E}(f_1) = R_{\text{eff}}^{s,t}(r)$. Hence, to prove (6.7), it suffices to show that

$$\frac{1}{m} \le R_{\text{eff}}^{s,t}(r).$$

By Proposition 3.3, we have $\lambda_{\max}(L_G) \le 2\Delta(G) \le 2m$. Hence, by Proposition 3.8 and Theorem 2.8,

$$R_{\text{eff}}^{s,t}(r) \ge R_{\text{eff}}^{s,t}(\mathbb{1}) = (e_s - e_t)^T L_G^\dagger (e_s - e_t) \ge 2\lambda_2^{\uparrow}(L_G^\dagger) = \frac{2}{\lambda_1^{\downarrow}(L_G)} \ge \frac{1}{m}.$$ ☐

Theorem 6.3. There is an algorithm that takes as input

- a connected weighted graph $G = (V, E, r)$,
- an orientation $\vec{G}$ of G,
- distinct vertices $s, t \in V$,
- scalars $\alpha \in \mathbb{R}_+$ and $\delta \in (0, 1]$,

and computes a δ-approximately electrical (s,t)-flow with respect to $\vec{G}$ of value α in G. Moreover, the algorithm runs in time $\tilde{O}(m \log \omega/\delta)$, where $\omega := \kappa(\mathrm{Diag}(r))$ and $m := |E|$.

Proof. Define $r_{\min} := \min\{ r_e : e \in E\}$. By Theorem 3.6, if f is an δ-approximately electrical (s,t)-flow of value α in G with weights given by $(\frac{1}{r_{\min}})r$, then $(r_{\min})f$ is an δ-approximately electrical (s,t)-flow of value α in G. Hence, we may assume that $r_e \in [1, \omega]$. Define $c \in \mathbb{R}^E$ where $c_e := 1/r_e$ for each $e \in E$, define $L := \mathcal{L}_G(c)$ and let $v := \alpha L^\dagger(e_s - e_t)$. Let $\varepsilon > 0$ and let $\hat{v} \in \mathbb{R}^V$ be such that

$$\|\hat{v} - v\|_L \le \varepsilon \|v\|_L.$$

By Theorem 5.1 we can compute such a vector in time $\tilde{O}(m \log 1/\varepsilon)$. By Theorem 3.6, if we define

$$f := \mathrm{Diag}(c) B_{\vec{G}}^T v, \tag{6.8}$$

then f is the electrical (s,t)-flow of value α in G. Moreover, define

$$\hat{f} := \mathrm{Diag}(c) B_{\vec{G}}^T \hat{v}.$$

Note that

$$\mathcal{E}(f) = f^T \mathrm{Diag}(c)^{-1} f \overset{(6.8)}{=} v^T B_{\vec{G}} \mathrm{Diag}(c) \mathrm{Diag}(c)^{-1} \mathrm{Diag}(c) B_{\vec{G}}^T v = v^T L v = \|v\|_L^2.$$

Similarly, $\mathcal{E}(\hat{f}) = \|\hat{v}\|_L^2$. Hence, using these facts and the triangle inequality,

$$\begin{aligned} \|\hat{v}\|_L &\le \|v\|_L + \|\hat{v} - v\|_L \le (1+\varepsilon)\|v\|_L \\ &\implies \mathcal{E}(\hat{f}) \le (1+\varepsilon)^2 \mathcal{E}(f). \end{aligned} \tag{6.9}$$

Note that $\hat{f}$ is not necessarily an (s,t)-flow, since it may not satisfy the flow conservation constraints. Define

$$i_{\text{ext}} := B_{\vec{G}}^T \hat{f} = L\hat{v},$$

set $n := |V|$ and let $\eta := \|i_{\text{ext}} - \alpha(e_s - e_t)\|_\infty$. By Lemma 6.1, given $\hat{f}$ we may compute an (s,t)-flow $\tilde{f}$ in G of value α such that

$$\|\hat{f} - \tilde{f}\|_\infty \le n\eta.$$

Let us show that

$$\text{the flow } \tilde{f} \text{ is a } \delta\text{-approximately electrical } (s,t)\text{-flow if } \varepsilon \le \delta/32n^4 m^{3/2}\omega. \tag{6.10}$$

By Proposition 3.3, we have $\|L\|_2 \le 2n$. Moreover, by Proposition 2.11, $\lambda_{\max}(L^{1/2}) = \lambda_{\max}(L)^{1/2}$. Thus, since $L \succeq 0$ and by Corollary 2.9, we have

$$\|L^{1/2}\|_2 = \lambda_{\max}(L^{1/2}) = \lambda_{\max}(L)^{1/2} = \|L\|_2^{1/2} \le 2n. \tag{6.11}$$

Hence,

$$\begin{aligned} \eta &\le \|i_{\text{ext}} - \alpha(e_s - e_t)\|_2 = \|L\hat{v} - Lv\|_2 \\ &\le \|L^{1/2}\|_2 \|L^{1/2}(\hat{v} - v)\|_2 = \|L^{1/2}\|_2 \|\hat{v} - v\|_L \\ &\overset{(6.11)}{\le} 2n\varepsilon\sqrt{\mathcal{E}(f)}. \end{aligned} \tag{6.12}$$

By Proposition 3.5, $\|\tilde{f}\|_\infty \le \alpha$. By the triangle inequality,

$$\|\hat{f}\|_\infty \le \|\tilde{f}\|_\infty + \|\hat{f} - \tilde{f}\| \le \alpha + n\eta. \tag{6.13}$$

Therefore,

$$\begin{aligned}
\mathcal{E}(\tilde{f}) &= \sum_{e\in E} r_e \tilde{f}_e^2 \le \sum_{e\in E} r_e(\hat{f}_e + n\eta)^2 = \mathcal{E}(\hat{f}) + 2n\eta \sum_{e\in E} r_e \hat{f}_e + n^2\eta^2 \sum_{e\in E} r_e \\
&\overset{(6.13)}{\le} \mathcal{E}(\hat{f}) + (2n\eta(\alpha + n\eta) + n^2\eta^2)\sum_{e\in E} r_e = \mathcal{E}(\hat{f}) + (2n\eta\alpha + 3n^2\eta^2)\sum_{e\in E} r_e \\
&\le \mathcal{E}(\hat{f}) + (2n\eta\alpha + 3n^2\eta^2)m\omega.
\end{aligned} \tag{6.14}$$

Note that

$$2n\eta\alpha \overset{(6.12)}{\le} 2n\alpha\Big(2n\varepsilon\sqrt{\mathcal{E}(f)}\Big) = 4\alpha n^2\varepsilon\sqrt{\mathcal{E}(f)} \le 4n^2\varepsilon\mathcal{E}(f)\sqrt{m}, \tag{6.15}$$

where in the last inequality we used the fact that $\alpha \le \sqrt{\mathcal{E}(f)m}$, which is a consequence of Lemma 6.2. Moreover, since $\delta \le 1$ and our hypothesis in (6.10), we have $\varepsilon \le 1$, and thus,

$$3n^2\eta^2 \overset{(6.12)}{\le} 3n^2\Big(2n\varepsilon\sqrt{\mathcal{E}(f)}\Big)^2 = 12n^4\varepsilon^2\mathcal{E}(f) \le 12n^4\varepsilon\mathcal{E}(f). \tag{6.16}$$

Hence,

$$\begin{aligned}
\mathcal{E}(\tilde{f}) &\le \mathcal{E}(\hat{f}) + (2n\eta\alpha + n^2\eta^2)m\omega && \text{by (6.14)} \\
&\le \mathcal{E}(\hat{f}) + \big(4n^2\varepsilon\sqrt{m} + 12n^4\varepsilon\big)\mathcal{E}(f)m\omega && \text{by (6.15) and (6.16)} \\
&\le \Big((1+\varepsilon^2) + 4\varepsilon n^2 m^{3/2}\omega + 12n^4\varepsilon m\omega\Big)\mathcal{E}(f) && \text{Since } \mathcal{E}(\hat{f}) \le (1+\varepsilon)^2\mathcal{E}(f) \text{ by (6.9)} \\
&\le \Big((1+\varepsilon^2) + 16\varepsilon n^4 m^{3/2}\omega\Big)\mathcal{E}(f).
\end{aligned}$$

Thus, if

$$\varepsilon \le \frac{\delta}{2(16n^4m^{3/2}\omega)},$$

then, using that $\delta \le 1$,

$$\begin{aligned}
\mathcal{E}(\tilde{f}) &\le \Big((1+\varepsilon^2) + \frac{\delta}{2}\Big)\mathcal{E}(f) \le \left(\Big(1 + \frac{\delta}{8}\Big)^2 + \frac{\delta}{2}\right)\mathcal{E}(f) = \left(1 + \frac{\delta}{4} + \Big(\frac{\delta}{8}\Big)^2 + \frac{\delta}{2}\right)\mathcal{E}(f) \\
&\le \Big(1 + \frac{\delta}{4} + \frac{\delta}{8} + \frac{\delta}{2}\Big)\mathcal{E}(f) = \Big(1 + \frac{7\delta}{8}\Big)\mathcal{E}(f) \le (1+\delta)\mathcal{E}(f).
\end{aligned}$$

This ends the proof of (6.10). It only remains to calculate the running time of the algorithm. We can compute $\hat{v}$ using the algorithm from Theorem 5.1, which runs in time $\tilde{O}(m\log 1/\varepsilon) = \tilde{O}(m\log\omega/\delta)$. Since the algorithm of Lemma 6.1, which was used to compute $\tilde{f}$ from $\hat{f}$, runs in time $O(m)$, the whole algorithm runs in time $\tilde{O}(m\log\omega/\delta)$. □

6.2 Multiplicative Weights Update Method

In this section, we describe how to apply the Multiplicative Weights Update Method (see [2]) to the problem of finding a feasible flow in a graph with value close to some target value. We first define an (ε, ρ)-oracle, which is, informally, a black box that returns a flow in a graph which is close to being feasible. These oracles will be used as a source of crude approximations to feasible flows to be used by the Multiplicative Weights Update method. Then, we show how to construct such an oracle by using the algorithm from Section 6.1 to compute approximately electrical flows. Next, we describe the application of the Multiplicative Weights

Update method, which computes a feasible flow by repeatedly calling an (ε, ρ)-oracle. Finally, we show then that using the oracle built using the algorithm from Section 6.1 in the Multiplicative Weights Update Method yields an efficient algorithm for finding a feasible flow in a graph with value close to some target value.

Let $G = (V, E)$ be a graph, let $s, t \in V$ be distinct, and let $u \in \mathbb{R}^E_{++}$. The **capacity** of $e \in E$ is u_e. Let f be an (s,t)-flow in G. The flow f is **feasible** (with respect to u) if $|f_e| \le u_e$ for every $e \in E$. The **congestion** of an edge $e \in E$ in f (with respect to the capacities u) is

$$\mathrm{cong}_f(e) \coloneqq \frac{|f_e|}{u_e}.$$

Note that f is a feasible flow in G if and only if $\mathrm{cong}_f(e) \le 1$ for every $e \in E$.

Let $G = (V, E, w)$ be a weighted graph with capacities $u \in \mathbb{R}^E_{++}$, let $\varepsilon > 0$, let $\alpha \in \mathbb{R}_+$, and let $\rho \in \mathbb{R}_{++}$. If $s, t \in V$ are distinct, an (s,t)-flow f in G is (ε, ρ)**-quasi-feasible** if

(i) $w^T \mathrm{cong}_f \le (1+\varepsilon)\|w\|_1$, and

(ii) $\|\mathrm{cong}_f\|_\infty \le \rho$.

An (ε, ρ)**-oracle** is an algorithm which takes as input a weighted graph $G = (V, E, w)$ with edge capacities $u \in \mathbb{R}^E_{++}$, distinct $s, t \in V$, and a target value $\alpha \in \mathbb{R}_+$, then

- if $\alpha \le \mathrm{OPT}$, where $\mathrm{OPT} \in \mathbb{R}_+$ is the value of a maximum (s,t)-flow in G, then the algorithm returns an (ε, ρ)-quasi-feasible (s,t)-flow f of G of value α;
- otherwise, the algorithm either returns an (ε, ρ)-quasi-feasible (s,t)-flow f of G of value α, or it fails.

Algorithm 6.1 $(\varepsilon, 3\sqrt{m/\varepsilon})$-oracle

Input: A weighted connected graph $G = (V, E, w)$ with capacities $u \in \mathbb{R}^E_+$, a value $\alpha \in \mathbb{R}_+$, and $s, t \in V$ distinct.

Output: An $(\varepsilon, 3\sqrt{m/\varepsilon})$-quasi-feasible (s,t)-flow of G if the algorithm does not fail, where $m \coloneqq |E|$.

Set $r_e \leftarrow \frac{1}{u_e^2}\left(w_e + \frac{\varepsilon\|w\|_1}{3m}\right)$ for each $e \in E$

Compute an $(\varepsilon/3)$-approximately electrical (s,t)-flow $\bar{f}$ of value α on (V, E, r) using the algorithm from Theorem 6.3.

if $\mathcal{E}(\bar{f}) \ge (1+\varepsilon)\|w\|_1$ **then return** *fail*

else return $\bar{f}$

Let us analyze the running time of Algorithm 6.1 for a fixed $\varepsilon > 0$. Let $G = (V, E, w)$ be a weighted graph with capacities $u \in \mathbb{R}^E_+$, let $\alpha \in \mathbb{R}_+$, and let $r \in \mathbb{R}^E_{++}$ be defined as in Algorithm 6.1. Define $m \coloneqq |E|$. The computation done by Algorithm 6.1 consists of finding an $(\varepsilon/3)$-approximately electrical (s,t)-flow in $G' \coloneqq (V, E, r)$ by using the algorithm from Theorem 6.3, besides some calculations that can be done in time $O(m)$. By Theorem 6.3, we can compute an $(\varepsilon/3)$-approximately electrical (s,t)-flow in G' in time $\tilde{O}(m \log \omega/\varepsilon)$, where $\omega \coloneqq \kappa(\mathrm{Diag}(r))$. Let $U \coloneqq \kappa(\mathrm{Diag}(u))$, and let $e, f \in E$. By the definition of r, we have

$$\frac{r_e}{r_f} \le U^2 \frac{3mw_e + \varepsilon\|w\|_1}{3mw_f + \varepsilon\|w\|_1} \le U^2 \frac{3m\|w\|_1 + \varepsilon\|w\|_1}{\varepsilon\|w\|_1} \le U^2\left(\frac{3m}{\varepsilon} + 1\right) \le \frac{6mU^2}{\varepsilon}.$$

Hence, $\omega \le 6mU^2/\varepsilon$. Thus, Algorithm 6.1 runs in time $\tilde{O}(m \log U/\varepsilon)$. The next proposition shows that Algorithm 6.3 is an $(\varepsilon, 3\sqrt{m/\varepsilon})$-oracle.

Proposition 6.4. Let $G = (V, E, w)$ be a weighted connected graph with edge capacities $u \in \mathbb{R}^E_{++}$, and let $s, t \in V$ be distinct. Let $\mathrm{OPT} \in \mathbb{R}^E$ be the value of a maximum (s,t)-flow in G, let $\alpha \in \mathbb{R}_{++}$ and let f be an $(\varepsilon/3)$-approximately electrical (s,t)-flow of value α in the graph (V, E, r) for some ε with $0 < \varepsilon \le 1$, where $r \in \mathbb{R}^E_+$ is defined by

$$r_e \coloneqq u_e^{-2}\left(w_e + \frac{\varepsilon\|w\|_1}{3m}\right), \qquad \forall e \in E.$$

If $\mathcal{E}(f) \le (1+\varepsilon)\|w\|_1$, then f is an $(\varepsilon, 3\sqrt{m/\varepsilon})$-quasi-feasible (s,t)-flow in G of value α, where $m \coloneqq |E|$. Moreover, if $|f| \le \mathrm{OPT}$, then $\mathcal{E}(f) \le (1+\varepsilon)\|w\|_1$.

Proof. Let f^* be a maximum (s,t)-flow in G with respect to the capacities u. Since f^* is feasible, we have $\mathrm{cong}_{f^*}(e) \leq 1$ for every $e \in E$. Hence,

$$\begin{aligned}\mathcal{E}(f^*) &= \sum_{e\in E} \frac{1}{u_e^2}\Big(w_e + \frac{\varepsilon\|w\|_1}{3m}\Big)(f_e^*)^2 = \sum_{e\in E}\Big(w_e + \frac{\varepsilon\|w\|_1}{3m}\Big)(\mathrm{cong}_{f^*}(e))^2 \\ &\leq \sum_{e\in E}\Big(w_e + \frac{\varepsilon\|w\|_1}{3m}\Big) = \|w\|_1\Big(1+\frac{\varepsilon}{3}\Big).\end{aligned} \tag{6.17}$$

Suppose that

$$\alpha \leq \mathrm{OPT}\,. \tag{6.18}$$

Let $\tilde{f}$ be an electrical (s,t)-flow in $G' \coloneqq (V,E,r)$ of value α. By Theorem 3.6, $(\mathrm{OPT}/\alpha)\tilde{f}$ is an electrical (s,t)-flow in G' of value OPT. Moreover, by the definition of electrical flow, we have that $\mathcal{E}((\mathrm{OPT}/\alpha)\tilde{f}) \leq \mathcal{E}(f^*)$. Hence,

$$\begin{aligned}\mathcal{E}(f) &\leq \Big(1+\frac{\varepsilon}{3}\Big)\mathcal{E}(\tilde{f}) \overset{(6.18)}{\leq} \Big(1+\frac{\varepsilon}{3}\Big)\Big(\frac{\mathrm{OPT}}{\alpha}\Big)^2\mathcal{E}(\tilde{f}) = \Big(1+\frac{\varepsilon}{3}\Big)\mathcal{E}\Big(\frac{\mathrm{OPT}}{\alpha}\tilde{f}\Big) \leq \Big(1+\frac{\varepsilon}{3}\Big)\mathcal{E}(f^*) \\ &\leq \Big(1+\frac{\varepsilon}{3}\Big)^2\|w\|_1 = \Big(1+\frac{2\varepsilon}{3}+\Big(\frac{\varepsilon}{3}\Big)^2\Big)\|w\|_1 \leq \Big(1+\frac{2\varepsilon}{3}+\frac{\varepsilon}{3}\Big)\|w\|_1 = (1+\varepsilon)\|w\|_1.\end{aligned}$$

It remains to prove that, if $\mathcal{E}(f) \leq (1+\varepsilon)\|w\|_1$, then f is an $(\varepsilon, 3\sqrt{m/\varepsilon})$-quasi-feasible (s,t)-flow in G of value α. Since $\mathcal{E}(f) \leq (1+\varepsilon)\|w\|_1$, we have

$$(1+\varepsilon)\|w\|_1 \geq \mathcal{E}(f) = \sum_{e\in E}\frac{1}{u_e^2}\Big(w_e + \frac{\varepsilon\|w\|_1}{3m}\Big)(f_e)^2 = \sum_{e\in E}\Big(w_e+\frac{\varepsilon\|w\|_1}{3m}\Big)(\mathrm{cong}_f(e))^2. \tag{6.19}$$

By construction, we know that $|f| = \alpha$. Let us prove that

$$w^T\,\mathrm{cong}_f \leq (1+\varepsilon)\|w\|_1. \tag{6.20}$$

From (6.19), we have

$$\sum_{e\in E} w_e\,\mathrm{cong}_f(e)^2 \leq (1+\varepsilon)\|w\|_1.$$

Hence, by the Cauchy-Schwarz inequality,

$$\begin{aligned}(w^T\,\mathrm{cong}_f)^2 &\leq \|w\|_1\Big(\sum_{e\in E} w_e\,\mathrm{cong}_f(e)^2\Big) \leq (1+\varepsilon)\|w\|_1^2 \\ \implies w^T\,\mathrm{cong}_f &\leq \sqrt{1+\varepsilon}\|w\|_1 < (1+\varepsilon)\|w\|_1.\end{aligned}$$

This ends the proof of (6.20). It only remains to prove that

$$\|\mathrm{cong}_f\|_\infty \leq 3\sqrt{\frac{m}{\varepsilon}}.$$

Let $e \in E$. From (6.19), we have

$$\frac{\varepsilon\|w\|_1}{3m}\,\mathrm{cong}_f(e)^2 \leq (1+\varepsilon)\|w\|_1.$$

Hence,

$$\mathrm{cong}_f(e) \leq \sqrt{\frac{(1+\varepsilon)3m}{\varepsilon}} \leq 3\sqrt{\frac{m}{\varepsilon}}. \qquad \square$$

The above proposition proves correctness of Algorithm 6.1. Hence, for any fixed $\varepsilon > 0$, Algorithm 6.1 is an $(\varepsilon, 3\sqrt{m/\varepsilon})$-oracle, where m is the number of edges of the graph received as input by the oracle. The following corollary summarizes what we have just proved.

Algorithm 6.2 Multiplicative Weights Update Method

Input: A connected graph $G = (V, E)$ with capacities $u \in \mathbb{R}^E_+$, vertices $s, t \in V$, an (ε, ρ)-oracle O with $0 < \varepsilon \leq 1/3$, and $\alpha \in \mathbb{R}_+$.

Output: If the algorithm does not fail, it returns a feasible (s,t)-flow of G of value at least $(1 - 3\varepsilon)\alpha$.

$m \leftarrow |E|$

$w^0 \leftarrow \mathbb{1} \in \mathbb{R}^E$ and $N \leftarrow \frac{2\rho \ln m}{\varepsilon^2}$.

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

  Query O with input the graph (V, E, w^{i-1}) with capacities u and target value α.

  if O fails **then return** *fail*

  Let f^i be the (s,t)-flow returned by O.

  for all $e \in E$ **do**

    $w_e^i \leftarrow w_e^{i-1}\Big(1 + \frac{\varepsilon}{\rho}\operatorname{cong}_{f^i}(e)\Big)$

return $\bar{f} \leftarrow \frac{(1-\varepsilon)^2}{(1+\varepsilon)N}\sum_{i=1}^N f_i$

Corollary 6.5. Let $\varepsilon > 0$ be fixed. Then there is an $(\varepsilon, 3\sqrt{m/\varepsilon})$-oracle such that, when given as input a weighted connected graph $G = (V, E, w)$ with capacities $u \in \mathbb{R}^E_{++}$, and distinct vertices $s, t \in V$, it runs in time $\tilde{O}(m \log U/\varepsilon)$, where $m \coloneqq |E|$, and $U \coloneqq \kappa(\mathrm{Diag}(u))$.

In Algorithm 6.2 we present the pseudo-code for the Multiplicative Weights Update routine. At each iteration of Algorithm 6.2, we make a call to the (ε, ρ)-oracle and update the weights of the graph. Since the algorithm executes $N = (2\rho \ln m)\varepsilon^{-2}$ iterations, we conclude that Algorithm 6.2 runs in time $\tilde{O}(\rho\varepsilon^{-2}\max\{m, t_O\})$, where t_O is the running time of the (ε, ρ)-oracle and m is the number of edges of the input graph. Note that we may suppose $t_O \in \Omega(m)$ since the (ε, ρ)-oracle needs $\Omega(m)$ time to read its input. Let us now prove the correctness of the algorithm.

Lemma 6.6. If $\varepsilon > 0$ and $x \in [0, 1]$, then

$$\exp((1 - \varepsilon)\varepsilon x) \leq 1 + \varepsilon x.$$

Proof. Let $x \in [0, 1]$, and let $\varepsilon > 0$. If $\varepsilon \geq 1$, then $(1 - \varepsilon)\varepsilon x \leq 0$, and hence, $\exp((1 - \varepsilon)\varepsilon x) \leq 1 \leq 1 + \varepsilon x$. Suppose that $\varepsilon < 1$. By definition, we have

$$\exp((1 - \varepsilon)\varepsilon x) = \sum_{i=0}^{\infty}\frac{((1-\varepsilon)\varepsilon x)^i}{i!} = 1 + \varepsilon x - \varepsilon^2 x + \sum_{i=2}^{\infty}\frac{((1-\varepsilon)\varepsilon x)^i}{i!}.$$

Hence,

$$\exp((1 - \varepsilon)\varepsilon x) \leq 1 + \varepsilon x \iff \sum_{i=2}^{\infty}\frac{((1-\varepsilon)\varepsilon x)^i}{i!} \leq \varepsilon^2 x.$$

If $x = 0$, the statement clearly holds. Suppose that $x > 0$. Then

$$\sum_{i=2}^{\infty}\frac{((1-\varepsilon)\varepsilon x)^i}{i!} \leq \varepsilon^2 x \iff \sum_{i=2}^{\infty}\frac{(1-\varepsilon)^i\varepsilon^{i-2}x^{i-1}}{i!} \leq 1$$

Note that

$$\sum_{i=2}^{\infty}\frac{(1-\varepsilon)^i\varepsilon^{i-2}x^{i-1}}{i!} \leq \sum_{i=2}^{\infty}(1-\varepsilon)^i\varepsilon^{i-2}x^{i-1}.$$

Since the right hand side of the above inequality is a geometric series with ratio $(1 - \varepsilon)\varepsilon x < 1$, we have

$$\sum_{i=2}^{\infty}(1-\varepsilon)^i\varepsilon^{i-2}x^{i-1} = \frac{(1-\varepsilon)^2 x}{1 - (1-\varepsilon)\varepsilon x} = \frac{(1-\varepsilon)^2 x}{1 - \varepsilon x + \varepsilon^2 x} \leq \frac{(1-\varepsilon)^2 x}{(1 - \varepsilon + \varepsilon^2)x}$$
$$= \frac{(1-\varepsilon)^2}{(1-\varepsilon+\varepsilon^2)} = \frac{(1-\varepsilon)^2}{(1-\varepsilon)^2 + \varepsilon} < 1.$$

□

Theorem 6.7. Let $G = (V, E)$ be a connected graph with edge capacities $u \in \mathbb{R}^E_{++}$, and let $s, t \in V$ be distinct. Let $\varepsilon > 0$, let $\rho \in \mathbb{R}_{++}$ and define $w^0 \coloneqq \mathbb{1} \in \mathbb{R}^E_{++}$. Let $\alpha \in \mathbb{R}_+$ and, for each $i \in \mathbb{N} \setminus \{0\}$, define the vectors $f^i \in \mathbb{R}^E$ and $w_i \in \mathbb{R}^E_{++}$ in order such that f^i is an (ε, ρ)-quasi-feasible (s,t)-flow of value α in (V, E, w^{i-1}), and w^i is given by the formula

$$w^i_e \coloneqq w^{i-1}_e \Big(1 + \tfrac{\varepsilon}{\rho} \operatorname{cong}_{f^i}(e)\Big), \qquad \forall e \in E.$$

Define

$$\bar{f} = \frac{(1-\varepsilon)^2}{(1+\varepsilon)N} \sum_{i=1}^{N} f^i, \tag{6.21}$$

where $N \coloneqq 2\rho \ln m/\varepsilon^2$, and $m \coloneqq |E|$. Then, $\bar{f}$ is a feasible (s,t)-flow of G with edge capacities u such that $|\bar{f}| \geq (1 - 3\varepsilon)\alpha$.

Proof. Let us first prove that,

$$\|w^{i+1}\|_1 \leq \|w^i\|_1 \exp\Big(\frac{(1+\varepsilon)\varepsilon}{\rho}\Big) \leq m \exp\Big(\frac{i(1+\varepsilon)\varepsilon}{\rho}\Big), \qquad \forall i \in \mathbb{N}. \tag{6.22}$$

For every $i \in \mathbb{N}$, we have

$$\begin{aligned} \|w^{i+1}\|_1 &= \sum_{e\in E} w^{i+1}_e = \sum_{e \in E} w^i_e \Big(1 + \frac{\varepsilon}{\rho} \operatorname{cong}_{f^{i+1}}(e)\Big) = \|w^i\|_1 + \frac{\varepsilon}{\rho} \sum_{e\in E} w^i_e \operatorname{cong}_{f^{i+1}}(e) \\ &\leq \|w^i\|_1 \Big(1 + \frac{(1+\varepsilon)\varepsilon}{\rho}\Big) \overset{\text{Le. 2.2}}{\leq} \|w^i\|_1 \exp\Big(\frac{(1+\varepsilon)\varepsilon}{\rho}\Big), \end{aligned}$$

where in the first inequality we used property (i) from the definition of an (ε, ρ)-quasi-feasible (s,t)-flow. The second inequality from (6.22) follows by induction on $i \in \mathbb{N}$ since $\|w^0\|_1 = \|\mathbb{1}\|_1 = m$. This ends the proof of (6.22).

Let us now prove that, if $e \in E$, then

$$w^i_e \geq \exp\Big(\frac{(1-\varepsilon)\varepsilon}{\rho} \sum_{j=1}^{i+1} \operatorname{cong}_{f^i}(e)\Big), \qquad \forall i \in \mathbb{N}. \tag{6.23}$$

Let $e \in E$. It is easy to verify by induction on $i \in \mathbb{N}$ that

$$w^{i+1}_e = \prod_{j=1}^{i+1} \Big(1 + \tfrac{\varepsilon}{\rho} \operatorname{cong}_{f^j}(e)\Big).$$

By Lemma 6.6, for each $i \in \mathbb{N}$,

$$w^{i+1}_e = \prod_{j=1}^{i+1} \Big(1 + \tfrac{\varepsilon}{\rho} \operatorname{cong}_{f^j}(e)\Big) \geq \exp\Big(\frac{(1-\varepsilon)\varepsilon}{\rho} \sum_{j=1}^{i+1} \operatorname{cong}_{f^i}(e)\Big).$$

This ends the proof of (6.23). We are now in position to prove (6.21).

Let $e \in E$. By (6.23) and (6.22), we have

$$\begin{aligned} m \exp\Big(\frac{N(1+\varepsilon)\varepsilon}{\rho}\Big) &\geq \|w^{N+1}\|_1 \geq w^{N+1}_e \geq w^N_e \geq \exp\Big(\frac{(1-\varepsilon)\varepsilon}{\rho} \sum_{j=1}^{N} \operatorname{cong}_{f^j}(e)\Big) \\ &\geq \exp\Big(\frac{N(1+\varepsilon)\varepsilon}{(1-\varepsilon)\rho} \operatorname{cong}_{\bar{f}}(e)\Big). \end{aligned}$$

Hence,

$$\operatorname{cong}_{\bar{f}}(e) \leq 1 - \varepsilon + \frac{(1-\varepsilon)\rho \ln m}{(1+\varepsilon)\varepsilon N} = 1 - \varepsilon + \frac{\varepsilon(1-\varepsilon)}{2(1+\varepsilon)} \leq 1.$$

Therefore, $\bar{f}$ is feasible in G with edge capacities u. Moreover, since $|f^i| = \alpha$ for every $i \in \mathbb{N}$, we have

$$|\bar{f}| = \frac{(1-\varepsilon)^2}{(1+\varepsilon)N} \sum_{i=1}^{N} |f^i| = \frac{(1-\varepsilon)^2}{1+\varepsilon}\alpha = \frac{1+\varepsilon-3\varepsilon+\varepsilon^2}{1+\varepsilon}\alpha > \frac{1+\varepsilon-3\varepsilon}{1+\varepsilon}\alpha \geq (1-3\varepsilon)\alpha. \qquad \square$$

The above Theorem proves the correctness of Algorithm 6.2. Moreover, as already discussed, the running time of the algorithm is $\tilde{O}(\rho\varepsilon^{-2}t_O)$, where t_O is the running time of the (ε,ρ)-oracle given as input to Algorithm 6.2. The following corollary summarizes what we have proved.

Corollary 6.8. There is an algorithm that takes as input:

- a connected graph $G = (V, E)$ with edge capacities $u \in \mathbb{R}^E$,
- distinct vertices $s, t \in V$,
- a target flow value $\alpha \geq 0$,
- a value $0 < \varepsilon \leq 1/3$,
- an (ε, ρ)-oracle with running time $O(t_O)$,

and, if $\alpha \leq \mathrm{OPT}$, it computes as output a feasible (s,t)-flow in G with respect to the capacities u of value at least $(1-3\varepsilon)\alpha$, where OPT is the value of a maximum (s,t)-flow in G. Otherwise, it either outputs an feasible (s,t)-flow in G with respect to the capacities u of value at least $(1-3\varepsilon)\alpha$, or it fails. Moreover, this algorithm runs in time $\tilde{O}(\rho\varepsilon^{-2}t_O)$, where $m := |E|$.

6.3 Calculating an Approximately Maximum Flow

Let $G = (V, E)$ be a graph with capacities $u \in \mathbb{R}^E_{++}$, let $s, t \in V$ be distinct, and let f be an (s,t)-flow in G. The flow f is **maximum** (with respect to u) if it is feasible and has maximum value. If $\delta \in \mathbb{R}_+$, then f is **δ-approximately maximum** (with respect to u) if it is feasible and $|f| \geq \delta\,\mathrm{OPT}$, where $\mathrm{OPT} \in \mathbb{R}_+$ is the value of a maximum (s,t)-flow in G.

It only remains to show how to use the algorithm from the previous section to find, for a given $\varepsilon > 0$, a $(1-\varepsilon)$-approximately maximum flow in a graph. The idea is to use binary search in the range of possible flow values, using as search condition for each target flow value the success or failure of the algorithm from Section 6.2. This will approximate the value of the maximum flow since the latter algorithm only fails for values greater than the optimum value, and when the algorithm succeeds, it yields a feasible flow of value not far from the one supplied. A problem that arises is that, if the sizes of the capacities are not bounded by a polynomial in the input size, neither is the maximum value of a feasible flow in G. In this section, we will show how to modify the input in a way which will not affect the value of a maximum flow by much so the graph has capacities bounded by a polynomial on the number of edges in the graph and on the inverse of the error tolerance supplied. Then, we describe our application of binary search to the approximately maximum flow problem.

Let $G = (V, E)$ be a graph with capacities $u \in \mathbb{R}^E_{++}$. The **bottleneck** of a path P in G (with respect to the capacities u) is $\min\{\, u_e : e \in E(P)\}$.

Proposition 6.9. Let $D = (V, A)$ be a connected digraph with capacities $u \in \mathbb{R}^A_{++}$, and let $s, t \in V$ be distinct. If there is a feasible (s,t)-flow of value $\alpha \in \mathbb{R}_+$ in D, then there is a feasible (s,t)-flow f in D, a collection $\mathcal{P}$ of (s,t)-paths in D with $|\mathcal{P}| \leq |A|$, and a vector $c \in \mathbb{R}^{\mathcal{P}}_+$ with $\|c\|_1 = \alpha$ such that

$$f = \sum_{P \in \mathcal{P}} c(P)\mathbb{1}_{E(P)}. \tag{6.24}$$

Moreover, let $G = (V, E)$ be a connected graph with capacities $u \in \mathbb{R}^E_{++}$, and let $s, t \in V$ be distinct. If there is a feasible (s,t)-flow of value $\alpha \in \mathbb{R}_+$ in G, then there is a feasible (s,t)-flow f in G, a collection $\mathcal{P}$ of (s,t)-paths in G with $|\mathcal{P}| \leq |E|$, and a vector $c \in \mathbb{R}^{\mathcal{P}}_+$ with $\|c\|_1 = \alpha$ such that

$$f = \mathrm{Diag}(\mathrm{sgn}(f)) \sum_{P \in \mathcal{P}} c(P)\mathbb{1}_{E(P)}. \tag{6.25}$$

In particular, if $\beta \in \mathbb{R}_{++}$ is the value of the maximum bottleneck of an (s,t)-path in G, then $\|c\|_1 \leq m\beta$.

Proof. Let f be a feasible (s,t)-flow of value α in G, and let $D = (V, A)$ be the induced orientation of G with respect to f. Define $g \coloneqq \text{Diag}(\text{sgn}(f))f$. By the definition of induced orientation, g is an (s,t)-flow in D of value α. Hence, proving the directed case of the statement yields the undirected case directly.

By Proposition 3.4, there is a collection of directed circuits $\mathcal{C}$ in D, a collection of directed (s,t)-paths $\mathcal{P}$ in D with $|\mathcal{C}| + |\mathcal{D}| \leq |A|$, vectors $b \in \mathbb{R}_+^{\mathcal{C}}$ and $d \in \mathbb{R}_+^{\mathcal{P}}$ with $\|d\|_1 = \alpha$ such that

$$g = \sum_{C \in \mathcal{C}} b(C)\mathbb{1}_{A(C)} + \sum_{P \in \mathcal{P}} d(P)\mathbb{1}_{A(P)}.$$

Define

$$g' \coloneqq \sum_{P \in \mathcal{P}} d(P)\mathbb{1}_{A(P)}.$$

By Proposition 3.4, g' is an (s,t)-flow in D of value $\|d\|_1 = \alpha$. Hence, $\text{Diag}(\text{sgn}(f))g'$ is an (s,t)-flow in G of value α.

Let $\beta \in \mathbb{R}_{++}$ be the maximum bottleneck of an (s,t)-path in G. By the definition of maximum bottleneck, and since every directed (s,t)-path in D is an (s,t)-path in G, for every $P \in \mathcal{P}$, there is $a \in A(P)$ such that $u_a \leq \beta$. Hence, if there is $P' \in \mathcal{P}$ such that $c(P) > \beta$, then $g'_a > \beta$ for each $a \in A(P)$, which is a contradiction since g' is a feasible flow. Therefore, $\|d\|_1 \leq m\beta$. ☐

Proposition 6.10. Let $G = (V, E)$ be a connected graph with capacities $u \in \mathbb{R}_{++}^E$, let $0 < \varepsilon \leq 1$, and let $s, t \in V$ be distinct. Then there is a spanning connected subgraph $G' = (V, E')$ of G and capacities $u' \in \mathbb{R}^{E'}$ such that

(i) $1 \leq u'_e \leq 2m^2/\varepsilon$ for each $e \in E'$, where $m \coloneqq |E|$,

(ii) if $f' \in \mathbb{R}^{E'}$ is an $(1-\varepsilon/2)$-approximately maximum (s,t)-flow of G', then f is an $(1-\varepsilon/2)$-approximately maximum (s,t)-flow of G, where $f \in \mathbb{R}^E$ is defined by $f_e \coloneqq [e \in E']f'_e$ for every $e \in E$.

Moreover, the graph G' and the capacities u' can be computed in time $O(m + n\log n)$, where $n \coloneqq |V|$.

Proof. Let OPT be the value of a maximum (s,t)-flow in G and let $\beta \in \mathbb{R}$ be the maximum bottleneck of an (s,t)-path in G with respect to the capacities u. The value of β can be computed in time $O(m\log n)$ by computing a spanning tree T of maximum weight in G, and then computing the bottleneck of the unique (s,t)-path P in T. By Proposition 3.4, we have that $\beta\mathbb{1}_{A(P)}$ is an (s,t)-flow in G of value β. Moreover, by Proposition 6.9, there is a maximum (s,t)-flow f in G such that $\text{OPT} = |f| \leq m\beta$ and

$$\|f\|_\infty \leq m\beta. \tag{6.26}$$

Thus, we have

$$\beta \leq \text{OPT} \leq m\beta. \tag{6.27}$$

Define $u' \in \mathbb{R}^E$ by

$$u'_e \coloneqq \begin{cases} m\beta & \text{if } u_e > m\beta, \\ u_e & \text{otherwise,} \end{cases} \qquad \forall e \in E.$$

It is easy to note that the value of a maximum flow in G with capacities u' is no greater then OPT. In the other hand, by (6.26) the flow f is feasible in G with capacities u'. Hence the value of a maximum flow in G with capacities u' is OPT. Define $F \coloneqq \{\, e \in E : u_e < \frac{\varepsilon\beta}{2m} \}$ and define $E' \coloneqq E \setminus F$. Moreover, define the graph $G' \coloneqq (V, E')$ and let $u'' \in \mathbb{R}_{++}^{E'}$ be the restriction of u' to E'. Let us show that

$$\text{the value of a maximum flow in } G' \text{ with capacities } u'' \text{ is at least OPT} - \varepsilon\beta/2. \tag{6.28}$$

To see this, let $\mathcal{C}$ and $\mathcal{P}$ be collections, and let $b \in \mathbb{R}^{\mathcal{C}}$ and $d \in \mathbb{R}_+^{\mathcal{P}}$ be vectors as in Proposition 3.4 such that

$$f = \text{Diag}(\text{sgn}(f)) \sum_{C \in \mathcal{C}} b(C)\mathbb{1}_C + \sum_{P \in \mathcal{P}} d(P)\mathbb{1}_P.$$

Note that, for each $e \in E$, we have $|f_e| \leq d(P)$ for every $P \in \mathcal{P}$ with $E(P) \ni e$. Hence, since f is feasible,

$$d(P) \leq \frac{\varepsilon\beta}{2m}, \qquad \forall P \in \mathcal{P} \text{ with } E(P) \cap F \neq \emptyset. \tag{6.29}$$

Define $\mathcal{P}' := \{ P \in \mathcal{P} : E(P) \cap F = \emptyset\}$ and $\mathcal{C}' := \{ C \in \mathcal{C} : E(C) \cap F = \emptyset\}$. Hence, by (6.29),

$$\sum_{P \in \mathcal{P}'} d(P)' \geq \sum_{P \in \mathcal{P}} d(P) - \frac{\varepsilon\beta}{2} = \|d\|_1 - \frac{\varepsilon\beta}{2} = \mathrm{OPT} - \frac{\varepsilon\beta}{2}.$$

Hence, by Proposition 3.4, we have that

$$f' := \mathrm{Diag}(\mathrm{sgn}(f)) \sum_{C \in \mathcal{C}'} b(C) \mathbb{1}_C + \sum_{P \in \mathcal{P}'} d(P) \mathbb{1}_P$$

is a feasible flow (with respect to u') of value at least $\mathrm{OPT} - \varepsilon\beta/2$. By construction, we have $f'_e = 0$ for each $e \in F$. Hence, restricting f' to E' yields a feasible flow of value at least $\mathrm{OPT} - \varepsilon\beta/2$ in G' with capacities u''. This ends the proof of (6.28).

Let OPT' be the value of a maximum flow in G' with capacities u''. By (6.28),

$$\alpha' \geq \mathrm{OPT} - \frac{\varepsilon\beta}{2} \overset{(6.27)}{\geq} \left(1 - \frac{\varepsilon}{2}\right) \mathrm{OPT} \geq 0,$$

where in the lest inequality we used that $\varepsilon \leq 1$ and that $\mathrm{OPT} \geq 0$. Let f' be a $(1 - \varepsilon/2)$-approximately maximum (s,t)-flow in G' with capacities u''. Note that we may extend f' to a feasible (s,t)-flow in G with capacities u by setting to 0 the flow on the edges of $E \setminus E'$. Moreover,

$$|f'| \geq \left(1 - \frac{\varepsilon}{2}\right) \mathrm{OPT}' \overset{(6.3)}{\geq} \left(1 - \frac{\varepsilon}{2}\right)^2 \mathrm{OPT} \geq (1 - \varepsilon) \mathrm{OPT}.$$

Hence, if we have a $(1 - \varepsilon/2)$-approximately maximum (s,t)-flow in G' with capacities u'', we can extend this flow to a $(1 - \varepsilon)$-approximately maximum (s,t)-flow in G with capacities u. It only remains now to construct capacities that obey the bound from (i).

Let $u''_{\min} := \min\{ u''_e : e \in E'\}$. It is easy to see that a flow f is feasible in G' with capacities u'' if and only if $(u''_{\min})^{-1} f$ is feasible in G' with capacities $(u''_{\min})^{-1} u''$. Since $\max\{ u''_e : e \in E'\} \leq m\beta$ by construction, we have

$$1 \leq u''_{\min} u''_e \leq m\beta, \qquad \forall e \in E'.$$

Hence, the graph G' equipped with capacities $(u''_{\min})^{-1} u''$ satisfies the properties (i) and (ii) from the statement. ☐

In Algorithm 6.3 we present the pseudocode to compute a $(1 - \varepsilon/2)$-approximately maximum (s,t)-flow in a graph computed by Proposition 6.10. Let us analyze its running time. Let the graph $G = (V, E)$ with capacities u, vertices $s, t \in V$ and error tolerance $\varepsilon > 0$ be the input to Algorithm 6.3. Define $\delta := \varepsilon/12$. The $(\delta, 3\sqrt{m/\delta})$-oracle used in Algorithm 6.3 runs in time $\tilde{O}(m \log(1/\varepsilon))$ by Corollary 6.5 since $1 \leq u_e \leq 2m^2/\varepsilon$ for every $e \in E$. Hence, by Corollary 6.8, the algorithm O used in Algorithm 6.3 runs in time $\tilde{O}(m^{3/2}\varepsilon^{-5/2})$. Moreover, since Algorithm 6.3 makes $O(\log m/\varepsilon)$ calls to O, we conclude that it runs in time $\tilde{O}(m^{3/2}\varepsilon^{-5/2})$.

For the correctness of the algorithm, let us show that, at the beginning of each iteration, the following invariant holds:

> Let O be the algorithm from Corollary 6.8. Then when O receives as input the graph G with capacities u, distinct vertices $s, t \in V$, error tolerance $\delta \in \mathbb{R}_+$, and target flow value $\alpha \in \mathbb{R}_+$, it fails if $\alpha = r$, and it succeeds if $\alpha = l$. (6.30)

Let $\mathrm{OPT} \in \mathbb{R}_+$ be the value of a maximum (s,t)-flow in G. In the first iteration $l = 0$, and hence O trivially succeeds in this case. Moreover, since $u_e \leq 2m^2/\varepsilon$ for each $e \in E$, by the flow decomposition result from Proposition 6.9, we have $\mathrm{OPT} \leq 2m^3/\varepsilon$. Moreover, suppose O succeeds when given as input α_f. By Corollary 6.8, O returns a feasible (s,t)-flow of value at least $(1 - 3\delta)r$. However, notice that in the first iteration we have

$$r > \frac{2m^3}{\varepsilon(1 - 3\delta)} \implies (1 - 3\delta) r > \frac{2m^3}{\varepsilon} \geq \mathrm{OPT},$$

Algorithm 6.3 Binary Search with MWU

Input: An error tolerance $\varepsilon > 0$, a connected graph $G = (V, E)$ with capacities $u \in \mathbb{R}^E_{++}$ such that $1 \leq u_e \leq 2m^2/\varepsilon$, and distinct vertices $s, t \in V$, where $m := |E|$.

Output: An approximately $(1 - \varepsilon/2)$-approximately maximum (s, t)-flow of G with capacities u.

$m \leftarrow |E|$
$l \leftarrow 0$ and $r \leftarrow \frac{2m^3}{\varepsilon(1-3\delta)} + 1$
$\delta \leftarrow \varepsilon/12$
Let O be the algorithm from Corollary 6.8 using a $(\delta, 3\sqrt{m/\delta})$-oracle from Corollary 6.5.
repeat
  $\alpha \leftarrow (l + r)/2$
  Query O with the graph G, capacities u, error tolerance δ and target flow value value α.
  if O fails **then** $r \leftarrow \alpha$
  else
    Let f be the resulting (s, t)-flow returned by O
    $l \leftarrow \alpha$
until $r - l \leq \varepsilon/4$
return f

which is a contradiction. Hence, O fails in this case. It is easy to see that (6.30) still holds in the following iterations of Algorithm 6.3. The following lemma completes the proof of the correctness of the algorithm.

Lemma 6.11. Let $G = (V, E)$ be a connected graph with capacities $u \in \mathbb{R}^E_{++}$ with $u \geq \mathbb{1}$, let $s, t \in V$ be distinct, and let $\varepsilon > 0$. Define $\delta := \varepsilon/12$ and let O be the algorithm from Corollary 6.8 using a $(\delta, 3\sqrt{m/\delta})$-oracle from Corollary 6.5. Let $l, r \in \mathbb{R}_+$ such that $l \leq r$ and that $r - l \leq \varepsilon/4$. We have that f is an $(1 - \varepsilon/2)$-approximately maximum (s, t)-flow in G if the following property holds:

> when O receives as input the graph G with capacities u, vertices s, t, error tolerance δ and target flow value $\alpha \in \mathbb{R}_+$, it fails if $\alpha = r$, and it succeeds if $\alpha = l$, returning an (s, t)-flow f in the latter case. (6.31)

Proof. Let $\mathrm{OPT} \in \mathbb{R}_+$ be the value of a maximum (s, t)-flow in G. By Corollary 6.8, we know that $r \geq \mathrm{OPT}$ since O fails when receives as input the graph G with capacities u, vertices s, t, error value δ and target flow value r. Moreover, since $u \geq \mathbb{1}$, and since G is connected, by Proposition 3.4 there is a feasible flow of value 1. Thus, $\mathrm{OPT} \geq 1$. Hence,

$$\frac{\varepsilon}{4} \geq r - l \geq \mathrm{OPT} - l \implies l \geq \mathrm{OPT} - \frac{\varepsilon}{4} \geq \mathrm{OPT}\Big(1 - \frac{\varepsilon}{4}\Big).$$

Moreover, by Corollary 6.8, we have $|f| \geq (1 - 3\delta)l = (1 - \varepsilon/4)l$. Therefore,

$$|f| \geq \Big(1 - \frac{\varepsilon}{4}\Big)l \geq \Big(1 - \frac{\varepsilon}{4}\Big)^2 \mathrm{OPT} \geq \Big(1 - \frac{\varepsilon}{2}\Big)\mathrm{OPT}\,.$$ □

Theorem 6.12. There is an algorithm that takes as input

- a connected graph $G = (V, E)$,
- edge capacities $u \in \mathbb{R}^E$,
- distinct vertices $s, t \in V$,
- a value $0 < \varepsilon \leq 2/3$,

and computes as output a $(1 - \varepsilon)$-approximately maximum (s, t)-flow in G with respect to the capacities u. Moreover, this algorithm runs in time $\tilde{O}(m^{3/2}\varepsilon^{-5/2})$.

Proof. By Proposition 6.10, to compute an $(1-\varepsilon)$-approximately maximum (s,t)-flow in G with capacities u, it suffices to compute an $(1-\varepsilon/2)$-approximately maximum (s,t)-flow in a subgraph $G' = (V, E')$ with edge capacities $u' \in \mathbb{R}_{++}^{E'}$ such that

$$\max\left\{ \frac{u_e}{u_f} : e, f \in E' \right\} \leq \frac{2m^2}{\varepsilon}. \tag{6.32}$$

Moreover, we can compute G' and the capacities u' in time $O(m+n\log n)$. To compute an $(1-\varepsilon/2)$-approximately maximum (s,t)-flow in the graph G' with capacities u', we can use Algorithm 6.3, which is correct by Lemma 6.11, and runs in time $\tilde{O}(m^{3/2}\varepsilon^{-5/2})$.

□

References

[1] I. Abraham, Y. Bartal, and O. Neiman. "Nearly Tight Low Stretch Spanning Trees". In: abs/0808.2017 (August 2008). arXiv: 0808.2017 [cs.DS]. URL: http://arxiv.org/abs/0808.2017 (cited on page 46).

[2] S. Arora, E. Hazan, and S. Kale. "The multiplicative weights update method: a meta-algorithm and applications". In: *Theory Comput.* 8 (2012), pages 121–164 (cited on pages 49, 53).

[3] P. Christiano, J. A. Kelner, A. Mądry, D. A. Spielman, and S.-H. Teng. *Electrical Flows, Laplacian Systems, and Faster Approximation of Maximum Flow in Undirected Graphs.* October 2010. arXiv: 1010.2921 [cs.DS]. URL: http://arxiv.org/abs/1010.2921 (cited on pages 1, 49). Complete version of "Electrical flows, Laplacian systems, and faster approximation of maximum flow in undirected graphs". In: *STOC'11—Proceedings of the 43rd ACM Symposium on Theory of Computing.* ACM, 2011, pages 273–281. STOC Best Paper Award.

[4] F. Chung and O. Simpson. "Solving linear systems with boundary conditions using heat kernel pagerank". In: *Algorithms and models for the web graph.* Volume 8305. Lecture Notes in Comput. Sci. Springer, 2013, pages 203–219 (cited on pages 1, 43).

[5] A. Cobham. "The intrinsic computational difficulty of functions". In: *Logic, Methodology and Philos. Sci. (Proc. 1964 Internat. Congr.)* North-Holland, Amsterdam, 1965, pages 24–30 (cited on page 1).

[6] J. Edmonds. "Paths, trees, and flowers". In: *Canadian Journal of Mathematics* 17 (1965), pages 449–467 (cited on page 1).

[7] A. Goldberg. *Andrew Goldberg guest blog on new max flow result.* November 8, 2012. URL: http://blog.computationalcomplexity.org/2012/11/andrew-goldberg-guest-blog-on-new-max.html (visited on 11/23/2015) (cited on page 1).

[8] J. A. Kelner, L. Orecchia, A. Sidford, and Z. A. Zhu. "A simple, combinatorial algorithm for solving SDD systems in nearly-linear time". In: *STOC'13—Proceedings of the 2013 ACM Symposium on Theory of Computing.* ACM, 2013, pages 911–920 (cited on pages 1, 43).

[9] I. Koutis, G. L. Miller, and R. Peng. "Approaching optimality for solving SDD linear systems". In: *IEEE 51st Annual Symposium on Foundations of Computer Science—FOCS 2010.* IEEE Computer Soc., 2010, pages 235–244 (cited on pages 1, 43).

[10] R. Kyng and S. Sachdeva. "Approximate Gaussian Elimination for Laplacians: Fast, Sparse, and Simple". In: *CoRR* abs/1605.02353 (May 2016). arXiv: 1605.00235 [cs.DS]. URL: http://arxiv.org/abs/1605.02353 (cited on pages 1, 43).

[11] D. A. Spielman and S.-H. Teng. "A local clustering algorithm for massive graphs and its application to nearly linear time graph partitioning". In: *SIAM J. Comput.* 42.1 (2013), pages 1–26. URL: http://dx.doi.org/10.1137/080744888 (cited on page 43).

[12] D. A. Spielman and S.-H. Teng. "Nearly linear time algorithms for preconditioning and solving symmetric, diagonally dominant linear systems". In: *SIAM J. Matrix Anal. Appl.* 35.3 (2014), pages 835–885. URL: http://dx.doi.org/10.1137/090771430 (cited on page 43).

[13] D. A. Spielman and S.-H. Teng. "Nearly-linear Time Algorithms for Graph Partitioning, Graph Sparsification, and Solving Linear Systems". In: *Proceedings of the Thirty-sixth Annual ACM Symposium on Theory of Computing.* STOC '04. New York, NY, USA: ACM, 2004, pages 81–90 (cited on pages 1, 43).

[14] D. A. Spielman and S.-H. Teng. "Spectral sparsification of graphs". In: *SIAM J. Comput.* 40.4 (2011), pages 981–1025. URL: http://dx.doi.org/10.1137/08074489X (cited on page 43).

[15] S.-H. Teng. "The Laplacian paradigm: emerging algorithms for massive graphs". In: *Theory and applications of models of computation*. Volume 6108. Lecture Notes in Comput. Sci. Springer, 2010, pages 2–14 (cited on pages 1, 43).

[16] N. K. Vishnoi. "$Lx = b$ Laplacian solvers and their algorithmic applications". In: *Found. Trends Theor. Comput. Sci.* 8.1-2 (2012), pages 1–141 (cited on page 29).

[17] D. Wagner. *Combinatorics of Electrical Networks*. 2009. URL: http://www.math.uwaterloo.ca/~dgwagner/Networks.pdf (visited on 11/05/2015) (cited on page 24).