

Combinatorial Geometry of Graph Partitioning - I

Manjish Pal *

Princeton University
mpal@cs.princeton.edu

Abstract

The c -BALANCED SEPARATOR problem is a graph-partitioning problem in which given a graph G , one aims to find a cut of minimum size such that both the sides of the cut have at least cn vertices. In this paper, we present new directions of progress in the c -BALANCED SEPARATOR problem. More specifically, we propose a family of mathematical programs, that depend upon a parameter $p > 0$, and is an extension of the uniform version of the SDPs proposed by Goemans and Linial for this problem. In fact for the case, when $p = 1$, if one can solve this program in polynomial time then simply using the Goemans-Williamson's randomized rounding algorithm for MAX CUT [11] will give an $O(1)$ -factor approximation algorithm for c -BALANCED SEPARATOR improving the best known approximation factor of $O(\sqrt{\log n})$ due to Arora, Rao and Vazirani [4]. This family of programs is not convex but one can transform them into so called **concave programs** in which one optimizes a concave function over a convex feasible set. It is well known that the optima of such programs lie at one of the extreme points of the feasible set [26]. Our main contribution is a combinatorial characterization of some extreme points of the feasible set of the mathematical program, for $p = 1$ case, which to the best of our knowledge is the first of its kind. We further demonstrate how this characterization can be used to solve the program in a restricted setting. Non-convex programs have recently been investigated by Bhaskara and Vijayaraghvan [6] in which they design algorithms for approximating Matrix p -norms although their algorithmic techniques are analytical in nature. It is important to note that the properties of concave programs allows one to apply techniques due to Hoffman [18] or Tuy *et al* [26] to solve such problems with arbitrary accuracy that, for special forms of concave programs, converge in polynomial time.

*A significant portion of this work was done when the author was a B-Tech, M-Tech dual degree student at IIT-Kanpur, India

1 Introduction

Graph partitioning is a problem of fundamental importance both in practice and theory. Many problems belonging to the several areas of computer science namely clustering, PRAM emulation, VLSI layout, packet routing in networks can be modeled as partitioning a graph into two or more parts ensuring that the number of edges in the cut is “small”. The word “small” doesn’t refer to finding the min-cut in the graph as it doesn’t ensure that the number of vertices in both sides of the cut is large. To enforce this balance condition one needs to normalize the cut-size in some sense. For the known notions of normalization like *conductance*, *expansion* and *sparsity*, finding optimal separators is NP-hard for general graphs. Hence, the objective is to look for efficient approximation algorithms. Because of the huge amount of work done to design good approximation algorithm for these problems, graph partitioning has become one of the central objects of study in the theory of geometric embeddings and random walks. Two fundamental problems which we will focus on are SPARSEST CUT and BALANCED SEPARATOR. These graph partitioning problems originally came up in the context of multi-commodity flows in which we are given a graph with capacities on the edges and a set of pairs of vertices (also called source-destination pairs) each having a demand and the aim is to find a cut that minimizes the ratio of capacity of the cut and the total demand through the cut. When the demand and capacities are all unit then the problem is called *uniform* and in case of general demands and capacities the problem is called *non-uniform*.

1.1 Uniform Version

The first approximation algorithm for such graph partitioning problems, came out of the study of Riemannian manifolds in form of the well known Cheeger’s Inequality [8] which says that if $\Phi(G)$ is the conductance of the graph and λ is the second largest eigenvalue of graph Laplacian then $2\Phi(G) \geq \lambda \geq \Phi(G)^2/2$. Because of the quadratic factor in the lower bound, the true approximation is $\frac{1}{\Phi(G)}$ which in worst case can be $\Omega(n)$ in worst case. The first true approximation algorithm for SPARSEST CUT and GRAPH CONDUCTANCE was designed by Leighton and Rao [22] whose approximation factor was $O(\log n)$. This also gave an $O(\log n)$ ***pseudo-approximation algorithm*** for c -BALANCED SEPARATOR. This algorithm is referred to as a pseudo-approximation algorithm because instead of returning a c -balanced cut, it returns a c' -balanced cut for some fixed $c' < c$ whose expansion is at most $O(\log n)$ times the optimum expansion of best c -balanced cut. Their algorithm was based on an LP framework motivated from the idea of Multi-commodity flows. Their main contribution was to derive an approximate max-flow, min-cut theorem corresponding to multi-commodity flow problem and the sparsest cut. Subsequently, a number of results were discovered which showed that good approximation algorithms exist when one is considering extreme cases such as the number of edges in the graphs is either very small or very large. In fact, it is known that for planar graphs one can find balanced cuts which are twice as optimal [10] and for graph with an average degree of $\Omega(n)$, one can design $(1 + \epsilon)$ -factor approximation algorithms where $\epsilon > 0$ with running time polynomial in input size [2] (such an algorithm is called a *Polyno-*

mial Time Approximation Scheme or PTAS). The approximation factor of $O(\log n)$ was improved to $O(\sqrt{\log n})$ in a breakthrough paper by Arora, Rao and Vazirani [4]. Their algorithm is based on semi-definite relaxations of these problems. The techniques and geometric structure theorems proved in their paper have subsequently led to breakthroughs in the field of metric embeddings. The basic philosophy behind these approximation algorithms is to embed the vertices of the input graph in an abstract space and derive a “nice” cut in this space. Recently, following a series of papers graph expansion has been related to the UNIQUE GAMES that ultimately led to sub-exponential time algorithms for UNIQUE GAMES [5].

1.2 Non-uniform Version

The non-uniform version of the cut problems is inextricably linked with low distortion metric embedding. It is easy to see that cut problems can be framed as optimization over l_1 metric which in general is NP-Hard. So the incentive is to embed the points in a space on which one can optimize efficiently for eg. the l_2^2 metric over which can optimize using SDPs. More specifically, using ideas from ARV and the measured descent technique of [14], firstly Lee [16] gave an $O(\log n)$ approximation algorithm for the non-uniform SPARSEST CUT, which was later improved to $O(\log^{3/4} n)$ by Chawla, Gupta and Rache [12]. A major breakthrough came from Arora, Lee and Naor [15] who improved this bound to $O(\sqrt{\log n} \log \log n)$ almost matching an old lower bound due to Enflo [17] which says that there is an n point metric in l_1 which need $\Omega(\sqrt{\log n})$ distortion to be embedded into l_2 .

1.3 Negative Results

Graph partitioning problems like SPARSEST CUT and BALANCED SEPARATOR are considered to among the few NP-hard problems which have resisted various attempts to prove inapproximability results. After the result of ARV, there has been a lot of impetus towards proving lower bounds on approximation factors. It has been shown by Ambuhl et al [1] that SPARSEST CUT can't have a PTAS unless NP-complete problems can be solved in randomized sub-exponential time. Because of the strong connections between semi-definite programming and the UNIQUE GAMES CONJECTURE (UGC) of Khot [20], inapproximability results are also known which assume UGC. More specifically, in a breakthrough result, Khot and Vishnoi [21] showed that UGC implies super-constant lower bounds on the approximation factor for the non-uniform version of the problems. Lee and Naor [24] gave an analytical proof of the result that by exhibiting an n point metric on the Heisenberg Group that is of negative type and needs $\omega(1)$ distortion to be embedded in l_1 . Recently, it has been shown by Cheeger, Kleiner and Naor [13], that the integrality gap of the non-uniform version of the sparsest cut SDP is $\Omega(\log^{O(1)} n)$. Devanur et al [9] showed that the integrality gap of the SDP relaxation of Arora-Rao-Vazirani is $\Omega(\log \log n)$ thereby disproving the original conjecture of ARV that the integrality gap of their SDP relaxation is almost a constant.

1.4 Non-Convex Programming

In this paper we work with a form of non-convex programs called *Concave Programming*. In order to define concave programming one first needs to define a concave function. A function $f : \mathbb{R}^d \rightarrow \mathbb{R}$ with domain $\mathbf{dom} f$ is said to be concave if $\mathbf{dom} f$ is convex and for all $x, y \in \mathbf{dom} f$, $f(\lambda x + (1 - \lambda)y) \geq \lambda f(x) + (1 - \lambda)f(y)$ for all $\lambda \in [0 - 1]$. Therefore, f is concave iff $-f$ is a convex function. Based on this definition one defines concave programming as a form of mathematical programming in which one optimizes a concave function over a convex feasible set. More formally, a concave programming problem can be written as $[\min_{x \in C} f(x)]$ where C is a convex set in \mathbb{R}^d and f is a concave function. The following is well known result for concave programming [19].

Fact 1. *For every concave programming problem there is an extreme point of the convex feasible set C which globally minimizes the optimization problem.*

The first algorithm for concave programming was designed by Tuy [27] in a restricted scenario when the feasible set is a polytope. A more general case, when the feasible set is convex but not necessarily polyhedral, was solved by Horst [19] and subsequently by Hoffman [18], Tuy and Thai [28]. General concave programming is NP-hard as $\{0, 1\}$ -integer programming can be cast as a concave program. There has been work towards designing efficient algorithms for some special class of concave programming. A comprehensive list of works done in concave programming can be found in Vaserstein’s homepage [31]. Recently, using analytical techniques Bhaskara and Vijayaraghavan [6] have successfully used non-convex programs to design algorithms for approximating matrix p -norms.

1.5 Our Contributions

Our main contribution is to initiate the study of combinatorial geometric properties of a non-convex relaxation for the c -Balanced Separator problem. We show that an efficient solution to our proposed program will imply improved an approximation algorithm for this problem. In section 2, we formally introduce the notions of sparsity and balanced cuts and sketch the Semi-Definite relaxation for c -BALANCED SEPARATOR of ARV. We then start section 4 by introducing a family of relaxations for c -BALANCED SEPARATOR which is generated by a parameter $p > 0$ and show that one can use its solution to design an $O(1)$ -factor approximation algorithm for the problem. Our result, although conditional, proposes new directions of progress on this problem and also a family of optimization problems which are more powerful than semi-definite programs in the context of approximation algorithms. Section 5 and 6 are devoted to find interesting properties on the geometry of the feasible region of our program and in section 7 we show how these properties can be used to design an efficient algorithm to search over a subset of extreme point called *vertices*. We end the paper with Section 8 in which we present conclusions and future directions.

2 Problem Definition

We now formally define the versions of balanced graph partitioning problem that we focus on, in this paper. c -BALANCED SEPARATOR¹

Given a graph $G = (V, E)$ with $|V| = n, |E| = m$, the c -BALANCED SEPARATOR problem is to find $\alpha_c(G)$ where $\alpha_c(G) = \min_{S \subset V, cn < |S| < (1-c)n} E(S, \bar{S})$.

Although our techniques can potentially be generalized to SPARSEST CUT² and other balanced graph partitioning problems.

2.1 SDP Relaxation for c -BALANCED SEPARATOR

Unifying the spectral and the metric based (linear programming) approaches, ARV used the following SDP relaxation to get an improved (pseudo)-approximation algorithm for the c -BALANCED SEPARATOR. Let us call this program SDP_{BS} ,

$$\begin{aligned} \min \frac{1}{4} \sum_{i,j \in E} \|v_i - v_j\|^2 \\ \|v_i\|^2 = 1 \quad \forall i \\ \|v_i - v_j\|^2 + \|v_j - v_k\|^2 \geq \|v_i - v_k\|^2 \quad \forall i, j, k \\ \sum_{i < j} \|v_i - v_j\|^2 \geq 4c(1-c)n^2 \end{aligned}$$

It is easy to see that this indeed is a *vector program* (and hence an SDP) and is a relaxation for the c -BALANCED SEPARATOR problem. To show that this is a relaxation we have to show that for every cut we can get an assignment of vectors such that all the constraints are satisfied and the value of the objective function is the size of the cut. Given a cut (S, \bar{S}) if one maps all the vertices in S to a unit vector \mathbf{n} and the vertices in \bar{S} to $-\mathbf{n}$ then the value of the function is indeed the cardinality of $E(S, \bar{S})$. The main idea behind their algorithm is to show that for any set of vectors which satisfy the constraints of the SDP there always exist two disjoint subsets of “large” size such that for any two points belonging to different subsets the squared Euclidean distance between them is at least $\Omega\left(\frac{1}{\sqrt{\log n}}\right)$. The same idea is also used to get an improved approximation algorithm for SPARSEST CUT in [4]. Subsequently, this key idea has crucially been used in various other SDP based approximation algorithms and in solving problems related to metric embeddings.

¹In [4] c -BALANCED SEPARATOR is defined as the minimum sparsity of c -balanced cuts, we will be working with a definition which upto constant factors is equivalent to their definition

²Given a graph $G = (V, E)$ with $|V| = n, |E| = m$, for each cut (S, \bar{S}) define *sparsity* of the cut to be the quantity $A(S) = \frac{|E(S, \bar{S})|}{|S||\bar{S}|}$. The uniform sparsest cut problem is to find $\alpha(G)$ where $\alpha(G) = \min_{S \subset V} A(S)$.

3 Non-Convex Relaxation for c -BALANCED SEPARATOR

Consider the following family of optimization problems which depend on a parameter $p \geq 0$. This family is essentially an extension of the semi-definite program proposed by ARV. Throughout the paper we will use $\|\cdot\|$ to represent the l_2 norm. Let us call this family of programs F_{BS}^p .

$$\begin{aligned} \min \frac{1}{2^p} \sum_{i,j \in E} \|v_i - v_j\|^p \\ \|v_i\|^2 = 1 \quad \forall i \\ \|v_i - v_j\|^p + \|v_j - v_k\|^p \geq \|v_i - v_k\|^p \quad \forall i, j, k \\ \sum_{i,j \in E} \|v_i - v_j\|^2 \geq 4c(1-c)n^2 \end{aligned}$$

Note that for $p = 2$ this is the SDP relaxation used by ARV. For $p = 1$, we are mapping the points onto a unit sphere, therefore we do not have to force the additional triangle inequality constraint of l_2 metric. The same mapping described for SDP_{BS} of the vertices of the graph onto the unit sphere allows us to conclude that each program in this family is also a relaxation for c -BALANCED SEPARATOR. In most part of the paper we will be working with the case $p = 1$. Now it is easy to see that if we can solve this program for $p = 1$, then simply using the randomized rounding algorithm of Goemans and Williamson [11] will give an $O(1)$ -approximation algorithm for the problem. This is because of the fact that the last constraint ensures that a random hyperplane will find two sets of large size on both sides with constant probability [4]. Another way to look at it is that in this program we are actually embedding the points in an l_2 metric which in turn is in l_1 metric. Therefore we have the following theorem:

Theorem 1. *An efficient algorithm for solving F_{BS}^p for $p = 1$ implies an $O(1)$ -factor approximation algorithm for c -BALANCED SEPARATOR.*

4 A Concave Programming Formulation

In this section, we consider the family of optimization problems F_{BS}^p proposed above and transform it into a concave program. This formulation allows us to use the algorithms which have been developed to solve a concave program with arbitrary accuracy. We now write F_{BS}^p as a program with variables as matrix entries and not as d -dimensional vectors. The variables in the new program are of the form $x_{ij} = \langle v_i, v_j \rangle$. Since all v_i 's are unit vectors we can write $\|v_i - v_j\|$ as $\sqrt{2 - 2\langle v_i, v_j \rangle}$. If we consider the matrix X with ij^{th} entry as x_{ij} use the transformation $z_{ij} = (1 - x_{ij})$, the new optimization problem becomes:

$$\min \frac{1}{2^{p/2}} \sum_{i,j \in E} z_{ij}^{p/2}$$

$$\begin{aligned}
z_{ij}^{p/2} + z_{jk}^{p/2} &\geq z_{ik}^{p/2} && \forall i, j, k \\
\sum_{i < j} z_{ij} &\geq c(1-c)n^2 \\
z_{ii} &= 0 && \forall i \\
\mathbf{1} - Z &\succeq 0
\end{aligned}$$

where $\mathbf{1}$ is the matrix with all entries as 1.

Let us call the above program \tilde{F}_{BS}^p . This formulation allows us to prove the following lemma:

Theorem 2. \tilde{F}_{BS}^p is a concave program for $0 < p < 2$.

Proof. See Appendix. □

5 Case $p = 1$

In this case the our feasibility problem now looks like the following:

$$\begin{aligned}
\min \frac{1}{\sqrt{2}} \sum_{i, j \in E} \sqrt{z_{ij}} \\
\sqrt{z_{ij}} + \sqrt{z_{jk}} &\geq \sqrt{z_{ik}} && \forall i, j, k \\
\sum_{i < j} z_{ij} &\geq c(1-c)n^2 \\
z_{ii} &= 0 && \forall i \\
\mathbf{1} - Z &\succeq 0
\end{aligned}$$

where $\mathbf{1}$ is the matrix with all entries as 1. Since $-1 \leq x_{ij} \leq 1$, $0 \leq z_{ij} \leq 2$. Let us denote the region in \mathbb{R}^d , which satisfies the last two constraints as \mathcal{P} , the triangle inequality constraints as \mathcal{T} and the “well-separated” constraint as \mathcal{H} . We will denote by \mathcal{F} the feasible region. Also if C is an inequality constraint, then C^* be the equality constraint corresponding to it. The following is an easy observation which follows essentially from the definition.

Observation 1. $\mathcal{P} \subseteq \mathcal{T}$.

5.1 The 3-Dimensional Intuition

If we just focus our attention to 3-variables and look at the feasible set (with out the positive semi-definite constraint), then because of the nature of the triangle inequality constraints the geometry of the feasible set looks as shown in the Figure 5.1. Notice that the feasible set is not polyhedral but it has one dimensional line segments on its boundary. The shaded region enclosed by the points $p_1, p_2, p_3, p_4, p_5, p_6$ and p_7 , depicts the feasible set. The line segments which are on the boundary of this object are $p_1p_6, p_3p_5, p_2p_4, p_6p_7, p_7p_4$ and p_5p_7 . Also p_1p_2, p_2p_3 and p_3p_1 are non-linear arcs which are on the boundary of the feasible set. Given this description the following is easy to show.

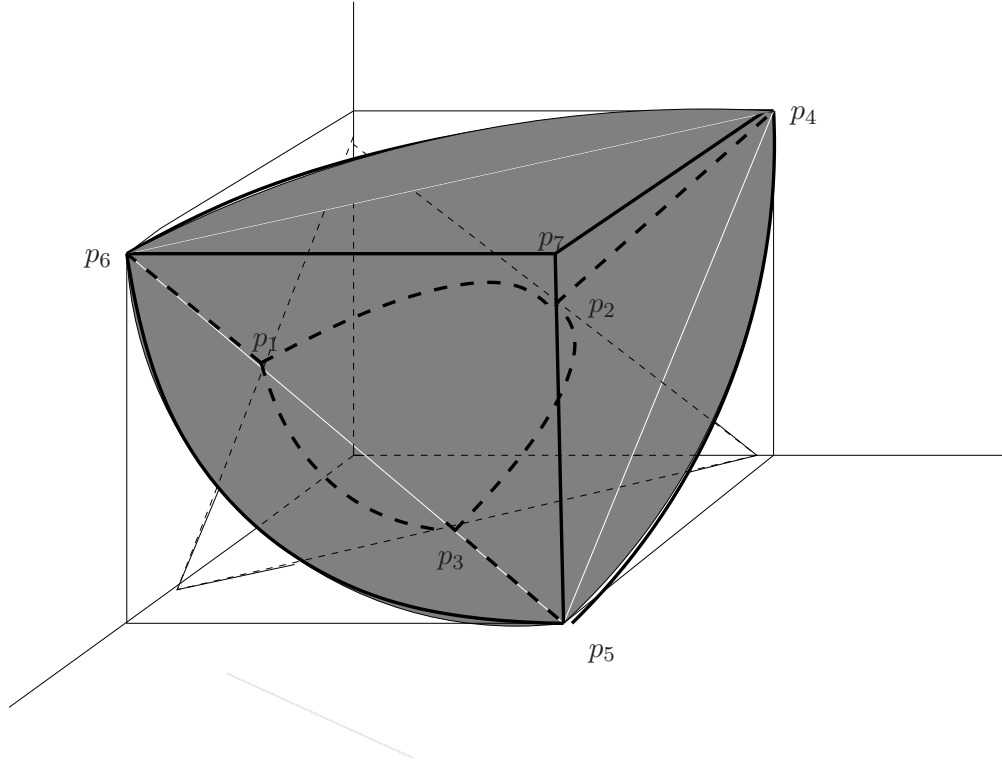


Figure 1: The picture in 3-D

Lemma 1. *Let $f = x_1^p + x_2^p + x_3^p$ where $p < 1/2$ be the objective function to be minimized over \mathcal{F} , then the optimum is achieved at one of extreme points p_1, p_2 or p_3 .*

6 Combinatorial Geometry of the Feasible Set

In this section we separately consider the constraints and develop tools to understand the geometry of the feasible set which can potentially help us in getting an efficient algorithm to solve the feasibility problem.

Our aim in the sections to follow is to give a tight characterization of the “vertices” of the proposed program which are defined as follows:

Definition 1 (Vertex). *A point $p \in \mathbb{R}^d$ is called a vertex-set of the feasible set \mathcal{F} if $p \in \mathcal{F}$ and there exists equality constraints $C_1^*, C_2^*, \dots, C_r^*$ such that $p = \bigcap_k^r C_k^*$.*

Definition 2 (Arc). *An arc a of \mathcal{F} is a closed one-dimensional curve joining two vertices of \mathcal{F}*

such that there exists equality constraints $C_1^*, C_2^*, \dots, C_r^*$ such that $a = \bigcap_i^r C_i^*$

Notice that we can analogously define vertices and arcs corresponding to the regions \mathcal{P}, \mathcal{H} and \mathcal{T} . From the previous section it is clear that some arcs are line segments while others are not. The arcs which are line segments we will call them *edges*. In the subsequent sections we will consider the constraints separately.

6.1 The Triangle Inequality Constraints

We will now look at the geometric structure posed by the Triangle Inequality Constraints inside the $[0 - 2]^d$ hypercube and prove some structural results on those.

Definition 3. Let \mathcal{R} denote the region inside $[0 - 2]^d$ that is formed by the intersections of the constraints $z_{ij} + z_{jk} \geq z_{kl}$, for all i, j, k .

Observation 2. The 0 vector is a vertex of \mathcal{T} .

Observation 3. $e(p)$ is an edge(vertex) of \mathcal{T} iff it is an edge(vertex) of \mathcal{R} .

We now take a deviation and first characterize all the symmetric $n \times n$ matrices with 0, 1 entries and main diagonal as 0 which satisfy the triangle inequality constraints. Since such matrices represent the adjacency matrix of some graph, we essentially need to characterize all graphs whose corresponding matrices satisfy the triangle inequality constraints. As we will see later this characterization will be helpful in analyzing the geometry of the triangle inequality constraints. But before that, we first need to define the following class of graphs:

Definition 4. A graph $G(V, E)$ is called **partial-clique** if there exists pairwise disjoint sets $S_1, S_2, \dots, S_r \subseteq V$, such that $G = K_V \setminus \{\bigcup_{i=1}^r K_{S_i}\}$ where K_S denotes edges in the complete graph on $S \subseteq V$.

We also define the following which will be of our interest later and subsequently prove a series of combinatorial results based on these.

Definition 5. A partial-clique $G = K_V \setminus \{\bigcup_{i=1}^r K_{S_i}\}$ over a vertex set V is called a **multi-clique** if $\bigcup_{i=1}^r S_i = V$.

(In the literature these graphs are popularly known as multi-partite graphs.) For the sake of brevity we will assume that the empty graph ($V = \emptyset$) is a multi-clique. We will also allow, again for the sake of simplification, that one can take S_i 's of cardinality 1.

Theorem 3. A graph satisfies the triangle inequality constraints if and only if it is a partial-clique.

Proof. It is easy to see that if the graph is a partial clique then it satisfies the triangle inequality constraints. Let S_1, S_2, \dots, S_r be the corresponding set of subsets. Consider any three vertices v_i, v_j, v_k , the following cases may arise: (i) none of them lies in a subset S , (ii) all three lie in distinct $S_{i_1}, S_{i_2}, S_{i_3}$ (iii) two of them lie in a set S_{i_1} and one lie in S_{i_2} (iv) only one of them lies in a subset S_{i_1} (v) all lie in the same subset S_i . In all these cases one can easily verify that the triangle inequality holds. Hence a partial-clique satisfies the triangle inequality constraints.

For the converse part consider a graph that satisfies the triangle inequality constraints and assume that it is not a partial-clique. Clearly the graph can't be disconnected because in that case one can easily find three vertices violating the corresponding triangle inequality. The triangle inequality essentially says that for all i, j, k , if two edges $v_i v_j$ and $v_j v_k$ are not present in the graph then the edge $v_i v_k$ should also not be present. Now assume that the complement of the graph has m connected components. Consider any one component say H . Let $S_1 \subseteq V$ be the set of vertices in the component H with an edge $v_i v_j \in H$. If H has just these two vertices v_{i_1} and v_{i_2} then it is already a clique. Let v_{i_3} be a vertex which share an edge with v_{i_1} or v_{i_2} . W.l.g let it be v_{i_1} . Since $v_{i_1} v_{i_2}$ and $v_{i_1} v_{i_3}$ is in the complement, $v_{i_3} v_{i_2}$ also have to be in the complement, which forms a 3-clique. We can repeat the same argument for the next vertex v_{i_4} , which will share an edge with at least one vertex in $\{v_{i_1}, v_{i_2}, v_{i_3}\}$, to show that it forms a 4-clique. Repeating this argument for all the vertices of the component we can show that H is a clique. The same holds for all the components. \square

Lemma 2. *The edges of \mathcal{R} are of the form λB where B is a bi-clique on $V = \{1, 2, \dots, n\}$.*

Proof. One way is easy to verify. For the other side, Notice that if the intersection of a set of equalities actually results into a line then there will be a set of variables x_{ij} such that all of them are equal and rest are all zeros (hence there is just one variable). Therefore the line will actually be a vector with some entries as λ and rest as 0. Since the 0/1 vectors which satisfy the triangle inequalities are the partial-cliques such a vector will pass through a partial clique G_v . Consider the graph $G_\lambda v$ which represents a weighted partial-clique with all edges with weight λ . Now if such a partial clique is not a bi-clique then either one of the two cases are possible: 1. One can find an edge $\{i, j\}$ such that weight of $\{i, j\}$ is λ and there is no pair of the form $\{j, k\}$ or $\{i, k'\}$ such that $w_{\{j, k\}} = 0$ or $w_{\{i, k'\}} = 0$.

2. G_λ is a multi-clique with λ as edge weights.

If it is the first case let $\{i, j\}$ and $\{i_1, j_1\}$ be the two pairs which have weight λ and since both have value λ , the intersection of the planes which we have chosen implies $z_{ij} = z_{i_1 j_1}$. This means there must be some k such that one of $z_{ij} + z_{jk} = z_{ik}$, $z_{ij} + z_{ik} = z_{jk}$ and $z_{ik} + z_{jk} = z_{ij}$ is chosen and $z_{jk} = z_{i_1 j_1}$ or $z_{ik} = z_{i_1 j_1}$ is implied by the rest of the planes chosen. But in both these cases, one of z_{ik} or z_{jk} is zero which is a contradiction.

For the second case, let i, j, k be three vertices lying in sets V_1, V_2 and V_3 , whose cliques are removed, respectively. Now due to the way equality of two variables is implied by a set of constraints, for

the edge $\{i, j\}$ to have the same value as $\{i, k\}$ there must exist some $j' \in V_2$ and $k' \in V_3$ such that the plane containing variable $x_{ij'}$ and $x_{ik'}$ is chosen, but this implies $x_{j'k'} = 0$ which is a contradiction. \square

Lemma 3. *Let T be the set of all 0/1 vectors in the hypercube which satisfy the triangle inequality constraints, then T is exactly the set of all 0-dimensional faces of \mathcal{R} .*

Proof. It is easy to see that the vertices of the cube are n -dimensional 0/1 vectors and edges are formed by joining those vertices which have hamming distance 1. Since \mathcal{R} is essentially the intersection of the Hamming cube with the unbounded polytope corresponding to the triangle inequalities, all the vertices of the cube which satisfy the triangle inequality constraints will also be the vertices of \mathcal{R} . We only need to show that there is no other vertices of \mathcal{R} . Since vertices form the boundary of the edges, the vertices of \mathcal{R} are formed as a result of the intersection of edges of the cube \mathcal{H} with \mathcal{P} or edges of \mathcal{P} with \mathcal{H} . But we can show that every edge of \mathcal{H} intersects the boundary of \mathcal{P} only at its end points and also that every edge of \mathcal{P} intersects the supporting planes of \mathcal{H} only at the vertices of \mathcal{H} hence the vertices of \mathcal{R} can only be the vertices of \mathcal{H} . To show the first claim let $(a_{12}, a_{13}, \dots, \lambda, \dots, a_{(n-1)n}, a_{nn})$ be an edge of \mathcal{H} in which all a_{ij} 's except one are fixed to either 0 or 1 and only one coordinate is varying as $\lambda \in [0 - 1]$. Now consider any plane corresponding to the triangle inequality constraints of the form $x_{ij} + x_{jk} = x_{ik}$. Since there is only one co-ordinate in the line this equality can't be satisfied for any $0 < \lambda < 1$, and hence intersection is only possible when λ is either 0 or 1. Based on Lemma ?? it is easy to see that the edges intersect the supporting planes of \mathcal{H} only at vertices of \mathcal{H} . \square

Theorem 4. *The line segment joining two vertices u and v of \mathcal{R} is an edge of \mathcal{R} if and only if the subgraphs of $G_{\lambda u + (1-\lambda)v}$ corresponding to the edges with weights λ and $1 - \lambda$ respectively are both bi-cliques.*

Proof. (if part) Let u and v be the vertices of \mathcal{R} and $G_u = K_V \setminus \{\bigcup_{i=1}^2 K_{S_i}\}$ $G_v = K_V \setminus \{\bigcup_{i=1}^2 K_{R_i}\}$ (since both are bi-cliques). Recall that by definition for any two vertices u_i and v_j in $S_i \cap R_j$ the edge between them is not present. Also the set of edges with weight λ will be those which are present in G_u and not in G_v vice versa for edges with weights $1 - \lambda$. Let H_λ and $H_{1-\lambda}$ be the subgraphs comprising of edges with weights λ and $1 - \lambda$ respectively. Consider H_λ . Since this graph is given to be a bi-clique we can assume it to be $H_\lambda = K_{V'} \setminus \{K_{T_1} \cup K_{T_2}\}$. We now choose hyperplanes such that their intersection gives us $G_{\lambda u + (1-\lambda)v}$. For every $i, j, k \in V'$ such that i, j is in some T_{i_1} and k is in some T_{i_2} , $i_1, i_2 \in \{1, 2\}$ with $T_{i_1} \neq T_{i_2}$ choose the hyperplanes $z_{ij} + z_{jk} = z_{ik}$ and $z_{ij} + z_{ik} = z_{jk}$ among the set of planes. Note that this implies that $z_{ik} = z_{jk}$ and $z_{ij} = 0$. As a result of selecting these hyperplanes we will get all the variables z_{ij} where $\{i, j\}$ is an edge in H_λ to be equal. Take this equal value as λ . Also for all $\{i, j\}$ which are not edge set of H_λ will have weight 0. Repeat the same exercise of choosing hyperplanes for the subgraph $H_{1-\lambda}$ but this time instead of taking the equal value as λ take the value as $1 - \lambda$. For all the rest of the edges $\{i, j\}$ with weight 1 choose the hyperplane $z_{ij} = 1$. To link these values we need to choose some other planes.

For all i, j, k such that $\{i, j\} \in H_\lambda$ and $\{j, k\} \in H_{1-\lambda}$ choose the plane $z_{ij} + z_{jk} = z_{ik}$. It is now easy to verify that the intersection of all these planes indeed gives the line segment $\lambda u + (1 - \lambda)v$.

(only if part) Let u and v be vertices of \mathcal{R} and the graph $G_{\lambda u + (1-\lambda)v}$ doesn't satisfy above mentioned condition. Similar to Lemma ??, it can be verified that in this case it is always the case that at least one the following two scenarios will arise:

1. The subgraph H_λ has more than one edge and one can find a pair $\{i, j\}$ such that $w_{\{i, j\}}$ is λ (or $1 - \lambda$) and there is no pair of the form $\{j, k\}$ or $\{i, k'\}$ such that $w_{\{j, k\}} = 0$ or $w_{\{i, k'\}} = 0$.
2. The subgraph H_λ (or $H_{1-\lambda}$) is a collection of disconnected multi-cliques.

If it is the first case let $\{i, j\}$ and $\{i_1, j_1\}$ be the two pairs which have weight λ (w.l.g. assume it is λ) and since both have value λ , the intersection of the planes which we have chosen implies $z_{ij} = z_{i_1 j_1}$. This means there must be some k such that one of $z_{ij} + z_{jk} = z_{ik}$, $z_{ij} + z_{ik} = z_{jk}$ and $z_{ik} + z_{jk} = z_{ij}$ is chosen and $z_{jk} = z_{i_1 j_1}$ or $z_{ik} = z_{i_1 j_1}$ is implied by the rest of the planes chosen. But in both these cases, one of z_{ik} or z_{jk} is zero which is a contradiction.

For the second case, let $\{i_1, j_1\}$ and $\{i_2, j_2\}$ be two pairs which are in different multi-cliques but $w_{\{i_1, j_1\}} = w_{\{i_2, j_2\}}$. Therefore, $z_{i_1 j_1} = z_{i_2 j_2}$ must be implied by the chosen set of hyperplanes. But from the discussion presented before, such a scenario implies that both $i_1 j_1$ and $i_2 j_2$ have to be in a connected graph which has to be a biclique. \square

6.2 Positive Semi-Definite Constraint

We now investigate the surface defined by the positive semi-definite constraint $\mathbf{1} - Z \succeq 0$. From Observation 1, the region defined by this constraint is enclosed inside the region defined by the triangle inequalities. In this direction we will prove certain interesting results again relating the graphs which some of these matrices correspond to.

Given a symmetric $n \times n$ matrix A with ± 1 entries define a new matrix \tilde{A} such that $\tilde{A}_{ij} = 1$ if $A_{ij} = -1$ and $\tilde{A}_{ij} = 0$ if $A_{ij} = 1$. The matrix \tilde{A} can be treated as the adjacency matrix of a graph on vertices $\{v_1, v_2, \dots, v_n\}$. We now prove the following lemma which will be of interest in the further discussion.

Lemma 4. *Given a symmetric matrix $A = [a_{ij}]$ with ± 1 entries, the expression $E(x_1, x_2, \dots, x_n) = \sum_i^n a_{ii} x_i^2 + 2 \sum_{i < j}^n a_{ij} x_i x_j$ is non-negative for all x_i 's $\in \mathbb{R}$, iff there exist $b_1, b_2, \dots, b_n \in \{1, -1\}$ such that E can be expressed as $(b_1 x_1 + b_2 x_2 + \dots + b_n x_n)^2$.*

Proof. Clearly one way is trivial, i.e. if E is of the above form then it must be non-negative. For the converse part we have to show that for all expressions E which are not of this form we can find some values of x_i 's $i = 1, 2, \dots, n$, for which the value of expression these choice of x_i 's becomes negative. We will denote $\mathbf{x} = (x_1, x_2, \dots, x_n)$.

Firstly, it is easy to see that any E that is non-negative for all x_i 's must have the values of a'_{ii} s as

1 because if any $a_{ii} = -1$ then the expression will be negative for the vector \mathbf{x} which is a at the i^{th} position and 0 otherwise where a is a non-zero number. Now we would show that if E is not of the form $(b_1x_1 + b_2x_2 + \dots + b_nx_n)^2$ then there always exists a triple i, j, k all three distinct such that among a_{ij}, a_{jk}, a_{ik} either all are -1 or two are 1 and one is -1. It is easy to see that under this assumption we will be done as for both these cases we can find an \mathbf{x} such that $E(\mathbf{x}) < 0$. If it is the first case i.e. all are -1's then take \mathbf{x} as the vector with a at the positions i, j, k and 0 otherwise. The value of E at this x will be $-3a^2 < 0$. If it is the other case then w.l.g assume that $a_{ij} = a_{jk} = 1$ and $a_{ik} = -1$. In this case we can choose \mathbf{x} which has a at positions i and k and $-a$ at position j . Again the value of the expression will be $-3a^2 < 0$.

We now have to prove that our assumption is always true. We will prove this by induction on n

Base: Can easily be verified for $n = 4$.

Induction: Assuming the above statement holds for $k = n$, we have to show it for $k = n + 1$. The above statement can be interpreted in terms of a graph. Given a matrix $A := a_{ij}$, consider a weighted clique on n vertices in which weight of an edge is 1 or -1. Therefore every expression E represents a clique. If it is of the form $(b_1x_1 + b_2x_2 + \dots + b_nx_n)^2$, then we can partition the vertex set of the corresponding graphs into two sets S^+ and $S^- = V \setminus S^+$ such that weights of all edges in $E(S^+, S^-)$ will be -1 and all other edges will be 1. Suppose the statement doesn't hold for $k = n + 1$ i.e. there exists an expression which is not of the form $(b_1x_1 + b_2x_2 + \dots + b_nx_n)^2$ but no triplet exists which satisfies our condition, i.e. all triples are either 1 or two are 1 and one is -1. In such a case, remove one vertex from the set and this property still holds for all triples and hence by induction we can assume this new graph can be decomposed into two sets S^+ and S^- as above. Now, if we put the removed vertex back then it is easy to verify that we cannot preserve the initial property. \square

Theorem 5. *An $n \times n$ symmetric matrix A with ± 1 entries is positive semidefinite if and only if the graph corresponding to \tilde{A} is a complete bipartite graph on vertices $\{v_1, v_2, \dots, v_n\}$.*

Proof. If the given matrix A is a positive semi-definite matrix then for all vectors $\mathbf{x} = (x_1, x_2, \dots, x_n)$, $\mathbf{x}^T A \mathbf{x} \geq 0$. Now in general for a symmetric matrix A , $\mathbf{x}^T A \mathbf{x}$ can be expanded as

$$\mathbf{x}^T A \mathbf{x} = \sum_i a_{ii} x_i^2 + 2 \sum_{i < j} a_{ij} x_i x_j$$

In our case, each $a_{ij} = \pm 1$. Now, we can appeal to Lemma 4 to conclude that the above expression will be non-negative iff it is of the form $(b_1x_1 + b_2x_2 + \dots + b_nx_n)^2$ for some b_i 's $\in \{1, -1\}$. From the proof of Lemma 4 the weighted graph can be partitioned into two sets S^+ and S^- such that edges of $E(S^+, S^-)$ are of weight -1 and rest have weight 1. As per the definition of \tilde{A} edges with weight 1 are removed and rest have weight 1, which makes the graph corresponding to \tilde{A} , a complete bipartite graph. \square

Lemma 5. *All points Z of the form λA where $A := [a_{ij}]$ corresponds to a bi-clique, are in \mathcal{P} for $\lambda \in [0 - 2]$.*

Proof. $X := [x_{ij}]$ be the matrix such that $X_{ij} = 1 - (\lambda a_{ij})^2$. Now λA will be in \mathcal{S} iff the matrix X is positive semi-definite. Since A corresponds to the adjacency matrix of a bi-clique, there will exist two disjoint non-empty subsets of $V = 1, 2, \dots, n$, S_1 and S_2 such that $S_1 = A \setminus S_2$ and for all $i \in S_1$ and $j \in S_2$, $a_{ij} = 1$. Also for all $i, j \in S_1$, $a_{ij} = 0$. and $i, j \in S_2$, $a_{ij} = 0$. Since it is the adjacency matrix of a graph $a_{ii} = 0$. This implies that $x_{ij} = 1 - \lambda$ for all $i \in S_1$ and $j \in S_2$ and $x_{ij} = 1$ for all $i, j \in S_1$ and $i, j \in S_2$. Also $x_{ii} = 1$. Now we know that the matrix X will be a PSD matrix iff there exists some n , n -dimensional vectors u_1, u_2, \dots, u_n such that $x_{ij} = \langle u_i, u_j \rangle$. Since $x_{ii} = 1$ all of these have to be unit vectors. Since $1 - \lambda$ takes values in the range $[-1, 1]$. There will always be two vectors n_1 and n_2 on the unit sphere such that $\langle n_1, n_2 \rangle = 1 - \lambda$. Therefore, we can choose the vectors u_1, \dots, u_n as for all $i \in S_1$ take $u_i = n_1$ and for all $i \in S_2$ take $u_i = n_2$. It is easy to verify that for all values of i, j , x_{ij} is indeed $\langle u_i, u_j \rangle$. \square

The following also is provable similar to Lemma 5

Lemma 6. *Let B be a partial clique from is obtained by removing k cliques form K_n , then all vectors corresponding to λB belong to \mathcal{P} for $\lambda \in [0 - \lambda_k]$ where $\lambda_k \in [0 - 2]$.*

Lemma 7. *An edge e of the cube $[0 - 2]^d$ is completely contained inside \mathcal{P} or completely outside it.*

7 Optimizing Over the Vertices is Easy

In this section we give a characterization of the vertices and arcs of \mathcal{F} and show how the objective function can easily be optimized over the vertices. Let Γ be the the points of the intersection of the hyperplane supporting H (denoted by \mathcal{H}^*) the segments of the form λB where B is a partial clique. The vertices Δ , of \mathcal{F} , can be divided into types,

Type 1: Points formed by the intersection of of \mathcal{H}^* with the edges of \mathcal{P} which we call Δ_1 and

Type 2: Vertices of the hypercube which satisfy \mathcal{H} and \mathcal{P} . Let this set be denoted by Δ_2 .

Notice that $\Delta_1 \subset \Gamma$.

We now prove an interesting result showing that we can infact optimize this objective function very efficiently over the set of vertices. The reason is that once we find out the intersection points of \mathcal{H}^* with the with the edges of \mathcal{T} and subsequently find the points among these which minimizes the objective function $\sum_{i,j \in G} \sqrt{z_{ij}}$. But things become simpler because there aren't many edges which intersect with the levels of the objective function.

More specifically, our result essentially answers the following question: given a connected graph $G = (V, E)$ is it possible to efficiently find a partial clique G' on the vertex set V such that if an edge $\{i, j\}$ is present in G then it is also present in G' and if it is not present in G then it is also not present in G' . The following theorem says given a connected graph $G = (V, E)$ and a subset of

edges E' one can easily decide whether there exists a partial clique $G^* = (V, E^*)$ such that $E' \subseteq E^*$ and $E \setminus E' \subseteq \bar{E}^*$, where \bar{E}^* is the set of edges in the complement of G^* . Before that we state the following observation that directly follows from Theorem 3.

Observation 4. *If $G = (V, E)$ is a partial-clique on n vertices that does not contain a fixed set of edges E' which forms a connected component then it does not contain the clique defined by the vertices induced over E' .*

Theorem 6. *Given a connected graph $G = (V, E)$ and a subset of edges E' there exists a unique (if any) partial clique $G^* = (V, E^*)$ such that $E' \subseteq E^*$ and $E \setminus E' \subseteq \bar{E}^*$, where \bar{E}^* is the set of edges in the complement of G^* and that partial-clique can be found efficiently.*

Proof. If such a partial-clique exists then it will be of the form $K_V \setminus \{\bigcup_{i=0}^r K_{S_i}\}$ for some subsets $S_1, S_2, \dots, S_r \subset V$ for some r . Let C_1, C_2, \dots, C_k be the connected components of $G' = (V, E \setminus E')$ with the corresponding vertex sets as V_1, V_2, \dots, V_k . Clearly all these vertex sets are pairwise disjoint. Using Observation 4 we can infer that each K_{V_i} is not present in the graph. Now we have to show that some other clique or a clique that contains some of these cliques is not missing. The first possibility is easily ruled out as G is a connected graph and hence any other clique will contain at least one edge in E which will violate the condition that $E' \subseteq E^*$. Similarly, for the other case as well if some other $K_{V'}$ is removed such that $V_s \subset V'$ for some s , then also the same condition will be violated. As evident from the proof such a partial-clique (if exists) can be computed efficiently. \square

8 Conclusion

In this paper, we propose a well-structured family of programs called concave programming and investigate the combinatorial geometric structure of the feasible set of the program and show how to possibly use them in the context of graph partitioning problems like C-BALANCED SEPARATOR. This is a major paradigmatic shift to attack these problems. It will of immense use to see whether or not such techniques can give us improved approximation factor for other problems. This also gives us hope that for many of the problems for which optimal approximation factors are not known one can possibly rely upon some “nice” programs which are although not convex but can be potential candidates for polynomial time solvability because of their geometric structure. Since this family is a new form of mathematical programming that is being used in an approximation algorithm, progress both in the direction of hardness and algorithms will provide more insights into the nature of these concave programs and can potentially lead us to optimal inapproximability results for various graph-partitioning problems Another tempting direction inspired from the recent results on the UNIQUE GAMES [?] is to exploit these combinatorial geometric ideas to design sub-exponential time $O(1)$ -approximation algorithms for the problem.

9 Acknowledgments

The author would like to thank Sanjeev Arora for discussing the prospects of mathematical programming paradigms beyond SDPs. Thanks to Purushottam Kar for going through an earlier draft of the paper and sending his comments.

References

- [1] C. Ambuhl, M. Mastrolilli and O. Svensson, *Inapproximability Results for Sparsest Cut, Optimal Linear Arrangement, and Precedence Constrained Scheduling* FOCS 2007, pp. 329-337.
- [2] S. Arora, D. Karger and M. Karpinski, *Polynomial Time Approximation Schemes for Dense Instances of NP-hard Problems*, Proceedings of the 27th ACM Symposium on Theory Of Computing , pp. 87-92, 1995.
- [3] S. Arora, S. Khot, A. Kolla, D. Steurer, M. Tulsiani and N. Vishnoi, *Unique Games on Expanding Constraint Graphs are Easy*, STOC 2008, pp. 21-28
- [4] S. Arora, S. Rao and U. Vazirani, *Expander Flows, Geometric Embeddings and Graph Partitioning*, JACM 56, 2009, pp. 1-37 (Preliminary version appeared in ACM STOC, 2004, pp. 222-231.)
- [5] S. Arora, B. Barak and D. Steurer, *Subexponential Algorithms for Unique Games and Related Problems*, FOCS, 2010 (to appear).
- [6] A. Bhaskara and A. Vijayaraghvan, *Approximating Matrix p -norms*. To appear in SODA 2011.
- [7] K. Ball, *An elementary introduction to modern convex geometry*, in *Flavors of Geometry*, S. Levy (ed.), Cambridge University Press, 1997.
- [8] J. Cheeger, *A lower bound for the smallest eigenvalue of the Laplacian*, Problem in Analysis, 195-199, Princeton Univ. Press, 1970.
- [9] N. R. Devanur, S. Khot, R. Saket and N. K. Vishnoi, *Integrality gaps for sparsest cut and minimum linear arrangement problems*, STOC 2006, pp. 537-546
- [10] N. Garg, H. Saran, V. V. Vazirani, *Finding separator cuts in planar graphs within twice the optimal*, FOCS 1994, pp. 14-23
- [11] M.X. Goemans and D. Williamson, *Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming*, JACM, 42(6) 1995, pp. 1115-1145.
- [12] S. Chawla, A. Gupta and H. Racke, *Approximations for Generalized Sparsest Cut and Embeddings of L_2 into L_1* , ACM Transactions on Algorithms, 4(2), 2008

- [13] J. Cheeger, B. Kleiner and A. Naor, *A $\log n^{\Omega(1)}$ integrality gap for the Sparsest Cut SDP*, FOCS, 4(2), 2009.
- [14] R. Krauthgamer, M. Mendel, J. Lee and A. Naor, *Measured descent: A new embedding method for finite metrics*, Geometric and Functional Analysis (GAFA) 15(4): 839-858, 2005.
- [15] S. Arora, J. Lee and A. Naor, *Euclidean distortion and the Sparsest Cut*, Journal of the American Mathematical Society, 21(1): 1-21, 2008
- [16] J. Lee, *Distance scales, embeddings, and metrics of negative type*, SODA 2005
- [17] P. Enflo, *On the non-existence of uniform homeomorphism between L spaces*. Arkiv. Mat., 8:103105, 1969.
- [18] K. L. Hoffman, *A Method for globally minimizing concave functions over convex sets*, Mathematical Programming (20), 1981, pp. 22-32.
- [19] R. Horst, *An Algorithm for Non-Convex Programming Problem*, Mathematical Programming(10)-3, 1985, pp. 498-514.
- [20] S. Khot, *On the power of Unique 2-prover 1-round Games* STOC 2002, pp. 767-775.
- [21] S. Khot and N. K. Vishnoi, *The Unique Games Conjecture, Integrality Gap for Cut Problems and Embeddability of Negative Type Metrics into l_1* , FOCS 2005, pp. 53-62.
- [22] T. Leighton and S. Rao *Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms*, JACM 46 1999, pp. 787-832. Prelim. version in ACM STOC 1988.
- [23] N. Linial, E. London and U. Rabinovich, *The Geometry of graphs and some of its algorithmic applications*, Combinatoria (15) 2 1995, pp 215-245.
- [24] J. Lee and A. Naor, *L_p metrics on the Heisenberg Group and the Goemans-Linial conjecture*, FOCS 2008
- [25] J. Matousek. *Lectures on Discrete Geometry*, Springer Verlag, 2002.
- [26] H. Tuy, T. V. Theiu, and Ng. Q. Thai *A Conical Algorithm for Globally Minimizing a Concave Function over a Closed Convex Set*, Mathematics of Operation Research(10)-3, 1985, pp. 498-514.
- [27] H. Tuy, *Concave Programming under Linear Constraints*, Dokl. Akad. Nauk (159), 1964, pp. 32-35. Translated Soviet Math. (5), pp. 1437-1440.
- [28] H. Tuy and Ng. Q. Thai, *Minimizing a Concave Function over a Compact Convex Set*, Proc. Conf. on Optimization Vitte/Hiddensee, May, 1981.

- [29] H. Tuy, S. Ghannadan, A. Migdalas and P. Vabrand, *A strongly polynomial algorithm for a concave production-transportation problem with a fixed number of nonlinear variables*, *Mathematical Programming* (72), 1996, pp. 229-258
- [30] V. Vazirani, *Approximation algorithms*, Springer Verlag, 2002.
- [31] *Concave Programming*, <http://www.math.psu.edu/vstein/concave.html>

A Appendix

A.1 Proof of Theorem 2

Proof. Since $z^{p/2}$ is concave for $p < 2$ for $z > 0$, and the sum of concave functions is also concave, the objective function is clearly concave. For the constraints defining the feasible set, $\sum_{i < j} z_{ij} \geq c(1-c)n^2$ and $z_{ii} = 0$ are convex. The constraint $\mathbf{1} - Z \succeq 0$ can be shown to be convex as follows: Let Z_1 and Z_2 be two matrices corresponding to the variables z_{ij} 's which lie in the feasible set. Therefore, they satisfy $\mathbf{1} - Z_1 \succeq 0$ and $\mathbf{1} - Z_2 \succeq 0$. Now, consider the line segment for $\lambda \in [0 - 1]$ $\lambda Z_1 + (1 - \lambda)Z_2$ and the matrix $\mathbf{1} - (\lambda Z_1 + (1 - \lambda)Z_2)$. This is positive semidefinite as it can be rewritten as $\lambda(\mathbf{1} - Z_1) + (1 - \lambda)(\mathbf{1} - Z_2)$ which is a sum of two PSD matrices.

The only type of constraint left are the triangle inequality constraints. Consider an inequality of this type say $z_{ij}^{p/2} + z_{jk}^{p/2} \geq z_{ik}^{p/2}$. In general, let us look at the region $x^r + y^r \geq z^r$ for $0 < r < 1$. If $r = 1/q$ for $q > 1$ then this region is same as $(x^{1/q} + y^{1/q})^q \geq z$. Let $p_1 = (x_1, y_1, z_1)$ and $p_2 = (x_2, y_2, z_2)$ be two points which lie in this region, i.e. $(x_1^{1/q} + y_1^{1/q})^q \geq z_1$ and $(x_2^{1/q} + y_2^{1/q})^q \geq z_2$. To prove the convexity of the region we need to show that for any $\lambda \in [0 - 1]$, $(\lambda x_1 + (1 - \lambda)x_2, \lambda y_1 + (1 - \lambda)y_2, \lambda z_1 + (1 - \lambda)z_2)$ also lies inside the region for all such points p_1 and p_2 . Therefore, we have to show $\lambda z_1 + (1 - \lambda)z_2 \leq \left((\lambda x_1 + (1 - \lambda)x_2)^{\frac{1}{q}} + (\lambda y_1 + (1 - \lambda)y_2)^{\frac{1}{q}} \right)^q$. Thus we will be done if we show $\lambda(x_1^{1/q} + y_1^{1/q})^q + (1 - \lambda)(x_2^{1/q} + y_2^{1/q})^q \leq \left((\lambda x_1 + (1 - \lambda)x_2)^{\frac{1}{q}} + (\lambda y_1 + (1 - \lambda)y_2)^{\frac{1}{q}} \right)^q$ which is equivalent to proving that the function $f(x, y) = \left(x^{\frac{1}{q}} + y^{\frac{1}{q}} \right)^q$ is concave. We will prove this by showing that the Hessian of this function is negative-definite for all x, y . We now compute the entries of the Hessian matrix. The following calculations are easy to verify,

$$\begin{aligned} \frac{\partial f}{\partial x} &= \left(1 + \frac{y^{\frac{1}{q}}}{x^{\frac{1}{q}}} \right)^{q-1} ; \frac{\partial f}{\partial y} = \left(1 + \frac{x^{\frac{1}{q}}}{y^{\frac{1}{q}}} \right)^{q-1} ; \frac{\partial^2 f}{\partial x^2} = - \left(\frac{q-1}{q} \right) \left(1 + \frac{x^{\frac{1}{q}}}{y^{\frac{1}{q}}} \right)^{q-2} \frac{y^{\frac{1}{q}}}{x^{\frac{q+1}{q}}} ; \\ \frac{\partial^2 f}{\partial y^2} &= - \left(\frac{q-1}{q} \right) \left(1 + \frac{y^{\frac{1}{q}}}{x^{\frac{1}{q}}} \right)^{q-2} \frac{x^{\frac{1}{q}}}{y^{\frac{q+1}{q}}} ; \frac{\partial^2 f}{\partial x \partial y} = \left(\frac{q-1}{q} \right) \left(\frac{1}{x^{\frac{1}{q}}} + \frac{1}{y^{\frac{1}{q}}} \right)^{q-2} \frac{1}{y^{\frac{1}{q}} x^{\frac{1}{q}}} = \frac{\partial^2 f}{\partial y \partial x} \end{aligned}$$

In order to show that the Hessian is negative-definite we have to show that for any $\alpha, \beta \in \mathbb{R}$, the following expression is always non-positive for all $x, y > 0$ (for x, y as 0 the derivatives do not

exist):

$$\begin{aligned}
& \alpha^2 \frac{\partial^2 f}{\partial x^2} + \beta^2 \frac{\partial^2 f}{\partial y^2} + 2\alpha\beta \frac{\partial^2 f}{\partial x \partial y} \\
&= -\left(\frac{q-1}{q}\right) \left[\alpha^2 \left(1 + \frac{x^{\frac{1}{q}}}{y^{\frac{1}{q}}}\right)^{q-2} \frac{y^{\frac{1}{q}}}{x^{\frac{q+1}{q}}} + \beta^2 \left(1 + \frac{y^{\frac{1}{q}}}{x^{\frac{1}{q}}}\right)^{q-2} \frac{x^{\frac{1}{q}}}{y^{\frac{q+1}{q}}} - \left(\frac{1}{x^{\frac{1}{q}}} + \frac{1}{y^{\frac{1}{q}}}\right)^{q-2} \cdot \frac{2\alpha\beta}{y^{\frac{1}{q}} x^{\frac{1}{q}}} \right] \\
&= -\left(\frac{q-1}{q}\right) \left(x^{\frac{1}{q}} + y^{\frac{1}{q}}\right)^q \left[\frac{\alpha^2 y^{\frac{1}{q}}}{x^{\frac{2q-1}{q}}} + \frac{\beta^2 x^{\frac{1}{q}}}{y^{\frac{2q-1}{q}}} - \frac{2\alpha\beta}{x^{\frac{q-1}{q}} y^{\frac{q-1}{q}}} \right] \\
&= -\left(\frac{q-1}{q}\right) \left(x^{\frac{1}{q}} + y^{\frac{1}{q}}\right)^q \left[\frac{\alpha^2 y^2 + \beta^2 x^2 - 2\alpha\beta xy}{x^{\frac{2q-1}{q}} y^{\frac{2q-1}{q}}} \right] = -\left(\frac{q-1}{q}\right) \left(x^{\frac{1}{q}} + y^{\frac{1}{q}}\right)^q \left[\frac{(\alpha y - \beta x)^2}{x^{\frac{2q-1}{q}} y^{\frac{2q-1}{q}}} \right]
\end{aligned}$$

which is non-positive for all α, β . This proves that the region $x^{p/2} + y^{p/2} \geq z^{p/2}$ is a convex set for all $0 < p < 2$. Hence the intersection of all the triangle inequality constraints is also convex. \square