Goemans and Williamson (1995) — "Goemans and Williamson Algorithm for MAXCUT"

February 14, 2018; rev. February 18, 2018

Kevan Hollbach, Lily Li

1 SOS SDP Hierarchy Review

Let K be a polytope defined by a set of m linear constraints on n variables, arising as the relaxation of an integer linear program using $\{0,1\}$-valued decision variables. That is,

$$K := \{\mathbf{x} \in \mathbb{R}^n \mid A\mathbf{x} - \mathbf{b} \geq 0\},$$

where the feasible set of the original ILP can be written as $K \cap \{0,1\}^n$.

In general, the relaxation K can be quite a bad approximation of the integral hull, $conv(K \cap \{0,1\}^n)$ — where $conv(\,\cdot\,)$ denotes the convex closure or convex hull of a set of points. If we could derive all linear constraints implied by those of K ($\{\sum_i A_{li}x_i - b_l \geq 0 \mid l \in [m]\}$) and integrality ($\{x_i^2 = x_i \mid i \in [n]\}$), then we could "tighten" K to the integral hull. Unfortunately, integer programming is NP-hard, so we can't expect to derive the integral hull in polynomial time.

For $t \geq 1$, the SOS degree t semidefinite program is a systematic way to tighten the polytope K to a better approximation of the integral hull. It is a "lift and project" method, meaning that additional variables are introduced (i.e. the polytope is "lifted" to a higher dimension), and then after tightening the lifted polytope, we "project" back down onto the original variables, resulting in a tightened polytope in the original dimension. The SOS method is sound: all feasible integral solutions are preserved by the tightening. For degree $t = n$, the SOS method derives exactly the integral hull, albeit in exponential time.

The SOS degree t SDP is the set $\text{SOS}_t(K)$ of points $\mathbf{y} = (y_\varnothing, y_{\{1\}}, \ldots, y_{\{n\}}, \ldots, y_I, \ldots, y_{[n]})$ in $\mathbb{R}^{2^{[n]}}$ that satisfy the following constraints.

$$\begin{aligned}
&(1) \qquad M_t(\mathbf{y}) := (y_{I\cup J})_{|I|,|J|\leq t} \succeq 0 \\
&(2) \qquad M_t^l(\mathbf{y}) := (\textstyle\sum_{i=1}^n A_{li}\, y_{I\cup J\cup\{i\}} - b_l\, y_{I\cup J})_{|I|,|J|\leq t} \succeq 0 \qquad \forall l \in [m] \\
&(3) \qquad y_\varnothing = 1
\end{aligned}$$

A feasible solution to the SOS_t SDP is indexed by subsets of $[n]$ corresponding to subsets of the original variables $x_1, ..., x_n$. The rows and columns of the matrices $M_t(\mathbf{y})$ and $M_t^l(\mathbf{y})$ are also indexed by subsets of the original variables, and the matrices are constrained to be positive semidefinite (psd). $M_t(\mathbf{y})$ is called the moment matrix, and $M_t^l(\mathbf{y})$ is called the moment matrix of slacks corresponding to constraint l.

Notice that although there are 2^n variables in this SDP, only $\binom{n}{\leq 2t+1} = \sum_{k=0}^{2t+1}\binom{n}{k} \leq n^{2t+1}$ variables are mentioned by the SDP constraints, so this SDP can be solved in $m\,n^{O(t)}$ time (modulo technical conditions that we typically ignore). Variables y_I, $|I| > 2t+1$, are entirely unconstrained at level t of the SOS hierarchy, but it is convenient to define the feasible sets of each level to be of the same dimension.

We summarize important properties of the SOS degree t SDP here. Let $\text{SOS}_t^{proj}(K)$ denote the projection of the feasible region $\text{SOS}_t(K)$ on to the original variables; that is, $\text{SOS}_t^{proj}(K) := \{(y_{\{1\}}, \ldots, y_{\{n\}}) \mid \mathbf{y} \in \text{SOS}_t(K)\}$.

Lemma 1. *Let $K = \{\mathbf{x} \in \mathbb{R}^n | A\mathbf{x} - \mathbf{b} \geq 0\}$, let $t \geq 0$, and let $\mathbf{y} \in SOS_t(K)$. Then the following hold.*

1. $0 \leq y_I \leq 1$, for all $|I| \leq t$
2. $0 \leq y_J \leq y_I \leq 1$, for all $I \subseteq J$ with $|I|, |J| \leq t$
3. $|y_{I\cup J}| \leq \sqrt{y_I\, y_J}$, for all $|I|, |J| \leq t$
4. $\text{SOS}_0(K) \supseteq \cdots \supseteq \text{SOS}_t(K) \supseteq \cdots \supseteq \text{SOS}_n(K)$

5. $K = \mathsf{SOS}_0^{proj}(K)$
6. $\mathsf{SOS}_n^{proj}(K) = conv(K \cap \{0,1\}^n)$

We now state two important lemmas. (For proofs of these results and the above properties, see the notes by Thomas Rothvoss.)

Lemma 2. ***Decomposition Lemma.*** *Let $\mathbf{y} \in SOS_t(K)$, and let $S \subseteq [n]$, $|S| \leq t$. We can write*

$$\mathbf{y} = \sum_{a \in \{0,1\}^S} \lambda_a \mathbf{y}^{(a)},$$

such that $\lambda_a \geq 0$, $\sum \lambda_a = 1$; and for all $\lambda_a \neq 0$, $\mathbf{y}^{(a)} \in SOS_{t-|S|}$ and $y_i^{(a)} = a_i \; \forall i \in S$.

The above lemma states that for a feasible point $\mathbf{y}$ at level t of the SOS hierarchy, and a set S of at most t original variables, $\mathbf{y}$ can be decomposed into a convex combination of feasible points $\mathbf{y}^{(a)}$ from a lower level of the hierarchy, such that all $\mathbf{y}^{(a)}$ are integral on their entries corresponding to the original variables in S.

Lemma 3. ***Locally Consistent Probability Distribution.*** *Let $\mathbf{y} \in SOS_t(K)$, and let $S \subseteq [n]$, $|S| \leq t$. There exists a probability distribution $\mathcal{D}_{\mathbf{y}}(S)$ over $\{0,1\}^S$ such that for all $I \subseteq S$*

$$\Pr_{a \sim \mathcal{D}_{\mathbf{y}}(S)} \left[\bigwedge_{i \in I} (a_i = 1) \right] = y_I.$$

The above lemma states that a feasible $\mathbf{y}$ at level t of the SOS hierarchy defines a probability distribution over integral assignments to the variables in S, where S can be any set of at most t original variables. Moreover, these "local" distributions are consistent with one another: for two sets of variables $S \cap S' \neq \varnothing$, the distributions $\mathcal{D}_{\mathbf{y}}(S)$ and $\mathcal{D}_{\mathbf{y}}(S')$ agree on the probability of events defined in their intersection. In this sense, $\mathbf{y}$ can be though of as a "pseudo-distribution" over integral assignments to the original variables $x_1, \ldots, x_n$, assigning locally-consistent probabilities to each event $\bigwedge_I (x_i = 1) \wedge \bigwedge_J (x_j = 0)$ that mentions at most t variables ($|I \cup J| \leq t$). Events mentioning more than t variables may be assigned negative (pseduo)-probability under $\mathbf{y}$, or may be inconsistent with one another in their marginal probabilities.

2 Goemans Williamson Algorithm (Standard Vector Program)

We now turn to MAXCUT problem, and the Goemans–Williamson 0.878-approximation algorithm. Let $G = (V, E)$ be a weighted undirected graph with $|V| = n$ and where each edge (i,j) has weight $w_{ij} \geq 0$. The goal of MAXCUT is to find a cut S which maximizes the sum of the edge weights of edges crossing $(S, V - S)$.

We can formulate the problem as the following quadratic program:

$$\text{Maximize:} \quad \sum_{(i,j) \in E} \frac{w_{ij}(1 - x_i x_j)}{2} \tag{1}$$

$$\text{Subject to:} \quad x_i \in \{-1, +1\}, \text{ for } i \in [n] \tag{2}$$

where x_i is associated with vertex v_i and $x_i x_j = 1$ if and only if v_i and v_j are placed in the same set. Let OPT denote the optimum solution to this quadratic program.

Next we introduce the vector programming relaxation of the above quadratic program:

$$\text{Maximize:} \quad \sum_{(i,j) \in E} \frac{w_{ij}(1 - \mathbf{u_i} \cdot \mathbf{u_j})}{2} \tag{3}$$

$$\text{Subject to:} \quad \|\mathbf{u}_i\|^2 = 1 \text{ and } \mathbf{u}_i \in \mathbb{R}^n, \text{ for } i \in [n]. \tag{4}$$

To see that this is indeed a relaxation, take $\mathbf{u}_i = (x_i, 0, ..., 0)$ for each $i \in [n]$. These $\mathbf{u}_i$'s satisfy the constraints ($\|\mathbf{u}_i\|^2 = 1$ and $\mathbf{u}_i \in \mathbb{R}^n$) and $\mathbf{u_i} \cdot \mathbf{u_j} = x_i x_j$. Thus, if OPT_{VP} denotes the optimum solution to the vector program, then $OPT_{VP} \geq OPT$.

The above vector program is equivalent to the following semidefinite program:

$$\text{Maximize:} \quad \sum_{(i,j)\in E} \frac{w_{ij}(1 - X_{ij})}{2} \tag{5}$$

$$\text{Subject to:} \quad X_{ii} = 1 \text{ for } i \in [n] \text{ and } X \succeq 0 \tag{6}$$

where X has entries X_{ij}; to see that these two forms are equivalent remark that $X \succeq 0$ if and only if $X = U^T U$. If we take the columns of U to be the set of vectors $\{\mathbf{u}_i\}$ of the vector program, then feasible solutions of SDP corresponds to feasible solutions of the vector program and vice versa.

We can solve this SDP in polynomial time and obtain an optimal solution X^*. Cholesky factorize X^* into $(U)^T U$ and let the columns of U, $\mathbf{u}_i \in \mathbb{R}^n$, be the solutions to the vector program. We want to round each $\mathbf{u}_i$ to $x_i \in \{-1, +1\}$. Then the set $\{x_i\}_{i=1}^n$ will be a solution to our original quadratic program. Apply randomized rounding as follows: pick $\mathbf{r} = (r_1, ..., r_n)$ by drawing each r_i independently from the distribution $\mathcal{N}(0, 1)$. Then let

$$x_i = \begin{cases} 1 & \mathbf{u}_i \cdot \mathbf{r} \geq 0 \\ -1 & \text{otherwise} \end{cases}.$$

It is helpful to have the geometric picture in mind: each $\mathbf{u}_i$ is a vector which lies on the $(n-1)$-dimensional unit sphere. The hyper-plane with normal $\mathbf{r}$ splits the sphere in-half. All vectors $\mathbf{u}_i$ in the same half of the sphere gets mapped to the same value $c \in \{-1, 1\}$ and all vectors $\mathbf{u}_j$ in the other half gets mapped to $-c$.

To show the constant of approximation, we consider the probability that an edge (i, j) gets cut. This is equivalent to the probability that $\mathbf{u}_i$ and $\mathbf{u}_j$ fall in different halves of the sphere cut by the hyper-plane. Consider the projecting of the normalized vector $\mathbf{r}$ onto the span of $\{\mathbf{u}_i, \mathbf{u}_j\}$. See Figure 1.

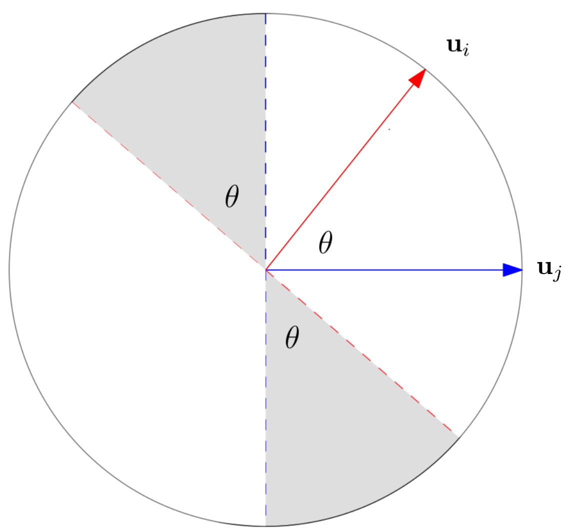

Figure 1: If the normalized $\mathbf{r}$ lies in the shaded region then $\mathbf{u}_i \cdot \mathbf{r}$ and $\mathbf{u}_i \cdot \mathbf{r}$ have different sign.

Thus the probability that $\mathbf{r} \cdot \mathbf{u}_i$ and $\mathbf{r} \cdot \mathbf{u}_j$ have different sign is $\frac{2\theta}{2\pi} = \frac{\theta}{\pi}$. Since $\theta = \arccos(\mathbf{u}_i \cdot \mathbf{u}_j)$,

$$\Pr[(i, j) \text{ is in the cut}] = \frac{\arccos(\mathbf{u}_i \cdot \mathbf{u}_j)}{\pi}. \tag{7}$$

We state without proof that

$$\frac{\arccos(x)}{\pi} \geq 0.878 \left(\frac{1-x}{2}\right) \tag{8}$$

for $x \in [-1, 1]$ — it helps to observe that the constant 0.878 approximately minimizes $f(x) = \frac{2\arccos(x)}{\pi(1-x)}$. Thus the expected sum of weights obtained by the algorithm is

$$\begin{aligned}
\mathbb{E}[W] &= \sum_{(i,j)\in E} w_{ij} \Pr[(i,j) \text{ is in the cut}] \\
&= \sum_{(i,j)\in E} w_{ij} \frac{\arccos(\mathbf{u}_i \cdot \mathbf{u}_j)}{\pi} && \text{by 7} \\
&\geq 0.878 \cdot \left(\sum_{(i,j)\in E} w_{ij} \frac{1 - \mathbf{u}_i \cdot \mathbf{u}_j}{2} \right) && \text{by 8} \\
&= 0.878 \cdot OPT_{VP}
\end{aligned}$$

Since the vector program is a relaxation of the original quadratic program, it is the case that $\mathbb{E}[W] \geq 0.878 \cdot OPT_{VP} \geq 0.878 \cdot OPT$. Further, since this algorithm is constructive, the cut found can have value at most OPT so $OPT \geq \mathbb{E}[W] \geq 0.878 \cdot OPT_{VP}$.

3 Vector Representation

Before examining how the SOS hierarchy behaves for the problem of MAXCUT, we see an equivalent way to represent solutions to the SOS SDP.

Let $\mathbf{y}$ be a feasible solution in $\mathrm{SOS}_t(K)$. We can equivalently represent $\mathbf{y}$ as vectors $\{v_I\}$, $|I| \leq t$, such that $y_{I\cup J} = \langle v_I, v_J \rangle$ for all $|I|, |J| \leq t$. This representation arises from the Cholesky decomposition of the moment matrix, $M_t(\mathbf{y}) \succeq 0$, into matrices V^T and V; the columns of V become the vectors v_I. Notice that $\|v_I\|^2 = y_I$ for $|I| \leq t$, and in particular, $\|v_\varnothing\|^2 = 1$.

This alternate representation associates a vector of dimension $\binom{n}{\leq t}$ to each event $|I| \leq t$, and allows us to form crucial geometric intuition about the SOS solution $\mathbf{y}$. In particular, we note that each vector v_I lies on the sphere defined by the endpoints of $v_\varnothing$, with radius $\frac{1}{2}$ and center $\frac{1}{2}v_\varnothing$. This can be seen by the following calculation: $\|v_I - \frac{1}{2}v_\varnothing\|^2 = \|v_i\|^2 - 2(\frac{1}{2}v_I \cdot v_\varnothing) + \frac{1}{4}\|v_\varnothing\|^2 = \frac{1}{4}$. See Figure 2.

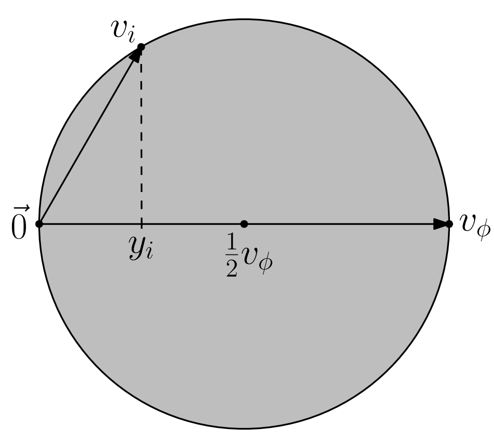

Figure 2: Visualization of the vector representation of an SOS solution

More generally, for events $J \subseteq I$ with $|I|, |J| \le t$, we see that v_I lies on the sphere defined by the endpoints of v_J: the sphere with radius $\frac{1}{2}\|v_J\| = \frac{1}{2}\sqrt{y_J}$ and center $\frac{1}{2}v_J$. (Again, because $\|v_I - \frac{1}{2}v_J\|^2 = \|v_i\|^2 - 2(\frac{1}{2}v_I \cdot v_J) + \frac{1}{4}\|v_J\|^2 = \frac{1}{4}y_J$.) Recall that v_I lies on the sphere defined by the endpoints of $v_\varnothing$ as mentioned before; thus v_I lies on the intersection of the two spheres. (The nontrivial intersection of two m-dimensional spheres is an $m-1$-dimensional sphere. In 3 dimensions, the intersection of two spheres is a circle; see Figure 3.)

Figure 3: Intersection of two 3D spheres: note the circular rim of the Death Star's superlaser

If two events are "disjoint", that is $y_{I\cup J} = 0$, then v_I and v_J will be at an angle of $\frac{\pi}{2}$, and thus v_I and v_J will be antipodal on the sphere centered at $\frac{1}{2}v_\varnothing$. Also, for any $|I|, |J| \le \frac{t}{2}$, the angle between v_I and v_J will be at most $\frac{\pi}{2}$, by the nonnegativity of $y_{\{I\cup J\}}$.

4 Goemans Williamson Algorithm via SOS ($t = 5$)

We are now ready to present the GW algorithm through the lens of SOS. Let our graph $G = (V, E)$ be as the above with $|V| = n$ and where each edge (i, j) has weight $w_{ij} \ge 0$. Formulate MAXCUT as the following integer linear program:

$$\begin{aligned} \text{Maximize:} \quad & \sum_{(i,j)\in E} w_{ij} z_{ij} \\ \text{Subject to:} \quad & \max(x_i - x_j, x_j - x_i) \le z_{ij} \le \min(x_i + x_j, 2 - x_i - x_j) \text{ for } (i,j) \in E, \\ & x_i, z_{ij} \in \{0,1\} \text{ for } i \in [n] \text{ and } (i,j) \in E \end{aligned}$$

where x_i is the indicator variable for a vertex chosen to be in set S of the partition and z_{ij} is the indicator variable for an edge crossing the cut $(S, V - S)$. Observe that $z_{ij} = (x_i - x_j)^2$.

Let K be the feasible region of the LP relaxation of the above integer program. For any graph we can set $x_i = \frac{1}{2}$ and $z_{i,j} = 1$ in the LP and obtain a value of $\sum_{(i,j)\in E} w_{ij}$. In particular, since the max cut of a complete graph on n-vertices with unit weight edges is at most $\frac{|E|}{2} + O(\sqrt{|E|})$ while the output to the relaxation is $|E|$, the integrality gap of this relaxation approaches 2 for large n.

Consider instead $\mathbf{y} \in SOS_5(K)$. The elements of note in $\mathbf{y}$ are y_{x_i}, y_{x_i,x_j}, and $y_{z_{ij}}$ for all $(i,j) \in E$.

Lemma 4. *For any edge* $(i, j) \in E$*,*

$$y_{z_{ij}} = y_{x_i} + y_{x_j} - 2y_{x_i,x_j}. \tag{9}$$

Proof. Consider edge $(i, j) \in E$, and let $S = \{x_i, z_{ij}, x_i x_j\}$. By the Decomposition Lemma (Lemma 2), $\mathbf{y} = \sum \lambda\tilde{\mathbf{y}}$ where $\tilde{\mathbf{y}} \in SOS_2(K)$ and the entry $\tilde{y}_s \in \{0,1\}$ for each $s \in S$. Since equation (9) is linear, if we can show that the lemma holds for each $\tilde{\mathbf{y}}$, then we can take a linear combination of the elements of $\tilde{\mathbf{y}}$ and show that the lemma holds for the elements of $\mathbf{y}$.

Since $\tilde{y}_{x_i}, \tilde{y}_{x_j} \in \{0,1\}$, by Locally Consistent Probability Distributions (Lemma 3), $\tilde{y}_i\tilde{y}_j = \Pr[x_i = 1] \cdot \Pr[x_j = 1] = \Pr[x_i = 1, x_j = 1] = \tilde{y}_{x_i x_j}$. Thus

$$\tilde{y}_{x_i} + \tilde{y}_{x_j} - 2\tilde{y}_{x_i,x_j} = \tilde{y}_{x_i} + \tilde{y}_{x_j} - 2\tilde{y}_{x_i}\tilde{y}_{x_j} = \tilde{y}_{x_i}^2 + \tilde{y}_{x_j}^2 - 2\tilde{y}_{x_i}\tilde{y}_{x_j} = (\tilde{y}_{x_i} - \tilde{y}_{x_j})^2$$

since $\tilde{y}_{x_i}^2 = \tilde{y}_{x_i}$ and $\tilde{y}_{x_j}^2 = \tilde{y}_{x_j}$ as $\tilde{y}_{x_i}, \tilde{y}_{x_j} \in \{0,1\}$. Since $\tilde{\mathbf{y}}$ is a feasible solution to the original LP and $\tilde{y}_{z_{ij}}, \tilde{y}_{x_i}, \tilde{y}_{x_j} \in \{0,1\}$, $(\tilde{y}_{x_i} - \tilde{y}_{x_j})^2 = \tilde{y}_{z_{ij}}$. Thus

$$y_{z_{ij}} = \sum \lambda \tilde{y}_{z_{ij}} = \sum \left(\tilde{y}_{x_i} + \tilde{y}_j - 2\tilde{y}_{x_i,x_j}\right) = y_{x_i} + y_{x_j} - 2y_{x_i,x_j}$$

for all edges (i,j) as claimed. □

We solve the SOS SDP to obtain an optimum moment matrix $M_5(\mathbf{y})$. Using the vector representation from section 3, $M_5(\mathbf{y}) = V^T V$ where the columns $\mathbf{v}_i$ of V satisfy: $\langle \mathbf{v}_i, \mathbf{v}_j \rangle = y_{x_i,x_j}$ for all $i,j \in [n]$. Recall however that the angle between any two vectors $\mathbf{v}_i$ and $\mathbf{v}_j$ is between 0 and $\frac{\pi}{2}$ so applying hyper-plane rounding on the $\mathbf{v}_i$'s would be sub-optimal; we want the vectors to be between 0 and π so that a random hyper-plane through the origin would be more likely to separate a pair of vectors belonging to different sets.

Perform the vector transformation $\mathbf{u}_i = 2\mathbf{v}_i - \mathbf{v}_\emptyset$. Observe that $\mathbf{u}_i$ is a unit vector on the sphere centered at the origin. See Figure 4. In essence this transformation takes vectors $\mathbf{v}_i \in [0,1]^n$ to vectors $\mathbf{u}_i \in [-1,1]^n$ before rounding $\mathbf{u}_i$ to $\{0,1\}$.

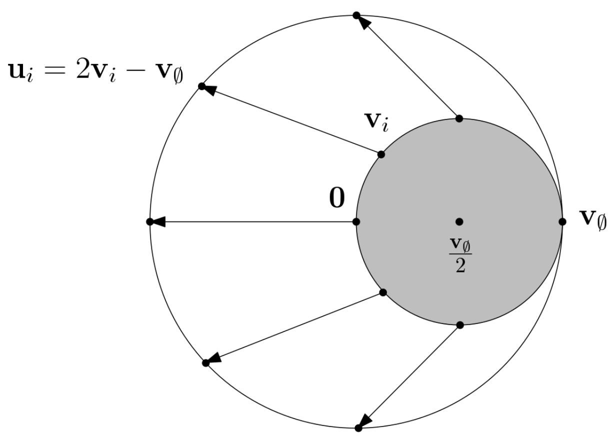

Figure 4: Vector transformation from $\mathbf{v}_i$ to $\mathbf{u}_i = 2\mathbf{v}_i - \mathbf{v}_\emptyset$.

Lemma 5. $\{\mathbf{u}_i\}$ *forms a solution to the vector program (equation 3 above) and* $y_{z_{ij}} = \frac{1 - \mathbf{u}_i \cdot \mathbf{u}_j}{2}$.

Proof. We need to show that $\mathbf{u}_i$ is a unit vector and that $y_{z_{ij}} = \frac{1 - \mathbf{u}_i \cdot \mathbf{u}_j}{2}$. Observe that

$$\mathbf{u}_i^2 = (2\mathbf{v}_i - \mathbf{v}_\emptyset)^2 = 4\mathbf{v}_i^2 - 4\mathbf{v}_i\mathbf{v}_\emptyset + \mathbf{v}_\emptyset^2 = 1$$

since $\langle \mathbf{v}_i, \mathbf{v}_i \rangle = y_{x_i} = \langle \mathbf{v}_i, \mathbf{v}_\emptyset \rangle$ and $\mathbf{v}_\emptyset^2 = y_\emptyset = 1$. Further

$$\frac{1 - \mathbf{u}_i \cdot \mathbf{u}_j}{2} = \frac{1 - (2\mathbf{v}_i - \mathbf{v}_\emptyset) \cdot (2\mathbf{v}_j - \mathbf{v}_\emptyset)}{2} = \mathbf{v}_i \cdot \mathbf{v}_\emptyset + \mathbf{v}_j \cdot \mathbf{v}_\emptyset - 2\mathbf{v}_j \cdot \mathbf{v}_i = y_{x_i} + y_{x_j} - 2y_{x_i,x_j} = y_{z_{ij}}.$$

by Lemma 4. Thus the $\mathbf{u}_i$ vectors form a solution to the vector program. □

It remains to round each $\mathbf{u}_i$ to $\{0,1\}$ to obtain a solution to our original ILP. The rounding algorithm and analysis is identical to that of the standard vector program formulation. Thus we again obtain a cut with expected weight $\mathbb{E}[W]$ bounded by $OPT \geq \mathbb{E}[W] \geq 0.878 \cdot OPT$.

Reference

Goemans, M. X. and Williamson, D. P. (1995). Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM (JACM)*, 42(6):1115–1145.