# Lecture 11

*In which we introduce the Arora-Rao-Vazirani relaxation of sparsest cut, and discuss why it is solvable in polynomial time.*

## 1 The Arora-Rao-Vazirani Relaxation

Recall that the sparsest cut $\phi(G)$ of a graph $G = (V, E)$ with adjacency matrix $A$ is defined as

$$\phi(G) = \min_{S \subseteq V} \frac{\frac{1}{2|E|} \sum_{u,v} A_{u,v} |1_S(u) - 1_S(v)|}{\frac{1}{|V|^2} \sum_{u,v} |1_S(u) - 1_S(v)|}$$

and the Leighton-Rao relaxation is obtained by noting that if we define $d(u, v) := |1_S(u) - 1_S(v)|$ then $d(\cdot, \cdot)$ is a semimetric over $V$, so that the following quantity is a relaxation of $\phi(G)$:

$$LR(G) = \min_{\substack{d : V \times V \to R \\ d \text{ semimetric}}} \frac{\frac{1}{2|E|} \sum_{u,v} A_{u,v} d(u, v)}{\frac{1}{|V|^2} \sum_{u,v} d(u, v)}$$

If $G$ is $d$-regular, and we call $M := \frac{1}{d} \cdot A$ the normalized adjacency matrix of $A$, and we let $\lambda_1 = 1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ be the eigenvalues of $M$ with multiplicities, then we proved in a past lecture that

$$1 - \lambda_2 = \min_{x: V \to \mathbb{R}} \frac{\frac{1}{2|E|} \sum_{u,v} A_{u,v} |x(u) - x(v)|^2}{\frac{1}{|V|^2} \sum_{u,v} |x(u) - x(v)|^2} \tag{1}$$

which is also a relaxation of $\phi(G)$, because, for every $S$, every $u$ and every $v$, $|1_S(u) - 1_S(v)| = |1_S(u) - 1_S(v)|^2$.

We note that if we further relax (1) by allowing $V$ to be mapped into a higher dimension space $\mathbb{R}^m$ instead of $\mathbb{R}$, and we replace $| \cdot - \cdot |$ by $|| \cdot - \cdot ||^2$, the optimum remains the same.

**Fact 1**

$$1-\lambda_2 = \min_{m,\mathbf{x}:V\to\mathbb{R}^m} \frac{\frac{1}{2|E|}\sum_{u,v} A_{u,v}||\mathbf{x}(u)-\mathbf{x}(v)||^2}{\frac{1}{|V|^2}\sum_{u,v}||\mathbf{x}(u)-\mathbf{x}(v)||^2}$$

PROOF: For a mapping $\mathbf{x}: V \to \mathbb{R}^m$, define

$$\delta(\mathbf{x}) := \frac{\frac{1}{2|E|}\sum_{u,v} A_{u,v}||\mathbf{x}(u)-\mathbf{x}(v)||^2}{\frac{1}{|V|^2}\sum_{u,v}||\mathbf{x}(u)-\mathbf{x}(v)||^2}$$

It is enough to show that, for every $\mathbf{x}$, $1-\lambda_2 \leq \delta(\mathbf{x})$. Let $x_i(v)$ be the $i$-th coordinate of $\mathbf{x}(v)$. Then

$$\begin{aligned}\delta(\mathbf{x}) &= \frac{\frac{1}{2|E|}\sum_i\sum_{u,v} A_{u,v}|x_i(u)-x_i(v)|^2}{\frac{1}{|V|^2}\sum_i\sum_{u,v}|x_i(u)-x_i(v)|^2}\\ &\geq \min_i \frac{\frac{1}{2|E|}\sum_{u,v} A_{u,v}|x_i(u)-x_i(v)|^2}{\frac{1}{|V|^2}\sum_{u,v}|x_i(u)-x_i(v)|^2}\\ &\geq 1-\lambda_2\end{aligned}$$

where the second-to-last inequality follows from the fact, which we have already used before, that for nonnegative $a_1,\ldots,a_m$ and positive $b_1,\ldots,b_m$ we have

$$\frac{a_1+\cdots a_m}{b_1+\cdots+b_m} \geq \min_i \frac{a_i}{b_i}$$

□

The above observations give the following comparison between the Leighton-Rao relaxation and the spectral relaxation: both are obtained by replacing $|1_S(u)-1_S(v)|$ with a "distance function" $d(u,v)$; in the Leighton-Rao relaxation, $d(u,v)$ is constrained to satisfy the triangle inequality; in the spectral relaxation, $d(u,v)$ is constrained to be the square of the Euclidean distance between $\mathbf{x}(u)$ and $\mathbf{x}(v)$ for some mapping $\mathbf{x}: V \to \mathbb{R}^m$.

The Arora-Rao-Vazirani relaxation is obtained by *enforcing both conditions*, that is, by considering distance functions $d(u,v)$ that satisfy the triangle inequality *and* can be realized of $||\mathbf{x}(u)-\mathbf{x}(v)||^2$ for some mapping $\mathbf{x}: V \to \mathbb{R}^m$.

**Definition 2** *A semimetric $d: V \to V \to \mathbb{R}$ is called of negative type if there is a dimension $m$ and a mapping $\mathbf{x}: V \to \mathbb{R}^m$ such that $d(u,v) = ||\mathbf{x}(u)-\mathbf{x}(v)||^2$ for every $u,v \in V$.*

With the above definition, we can formulate the Arora-Rao-Vazirani relaxation as

$$ARV(G) := \min_{\substack{d : V \times V \to R \\ d \text{ semimetric of negative type}}} \frac{\frac{1}{2|E|}\sum_{u,v} A_{u,v} d(u,v)}{\frac{1}{|V|^2}\sum_{u,v} d(u,v)} \tag{2}$$

**Remark 3** *The relaxation (2) was first proposed by Goemans and Linial. Arora, Rao and Vazirani were the first to prove that it achieves an approximation guarantee which is better than the approximation guarantee of the Leighton-Rao relaxation.*

We have, by definition,

$$\phi(G) \leq ARV(G) \leq \min\{LR(G), 1 - \lambda_2(G)\}$$

and so the approximation results that we have proved for $1 - \lambda_2$ and $LR$ apply to $ARV$. For every graph $G = (V, E)$

$$ARV(G) \leq O(\log |V|) \cdot \phi(G)$$

and for every regular graph

$$ARV(G) \leq \sqrt{8 \cdot \phi(G)}$$

Interestingly, the examples that we have given of graphs for which $LR$ and $1 - \lambda_2$ give poor approximation are complementary. If $G$ is a cycle, then $1 - \lambda_2$ is a poor approximation of $\phi(G)$, but $LR(G)$ is a good approximation of $\phi(G)$; if $G$ is a constant-degree expander then $LR(G)$ is a poor approximation of $\phi(G)$, but $1 - \lambda_2$ is a good approximation.

When Goemans and Linial (separately) proposed to study the relaxation (2), they conjectured that it would always provide a constant-factor approximation of $\phi(G)$. Unfortunately, the conjecture turned out to be false, but Arora, Rao and Vazirani were able to prove that (2) does provide a strictly better approximation than the Leighton-Rao relaxation. In the next lectures, we will present parts of the proof of the following results.

**Theorem 4** *There is a universal constant $c$ such that, for every graph $G = (V, E)$,*

$$ARV(G) \leq c \cdot \sqrt{\log |V|} \cdot \phi(G)$$

**Theorem 5** *There is an absolute constant $c$ and an infinite family of graphs $G_n = (V_n, E_n)$ such that*

$$ARV(G) \geq c \cdot \log\log |V_n| \cdot \phi(G)$$

In the rest of this lecture we discuss the polynomial time solvability of (2).

## 2 The Ellipsoid Algorithm and Semidefinite Programming

**Definition 6** *If $C \subseteq \mathbb{R}^m$ is a set, then a separation oracle for $C$ is a procedure that, on input $\mathbf{x} \in R^m$,*

- *If $\mathbf{x} \in C$, outputs "yes"*
- *If $\mathbf{x} \notin C$, outputs coefficients $a_1, \ldots, a_m, b$ such that*

$$\sum_i x_i a_i < b$$

*but, for every $\mathbf{z} \in C$,*

$$\sum_i z_i a_i \geq b$$

Note that a set can have a separation oracle only if it is convex. Under certain additional mild conditions, if $C$ has a polynomial time computable separation oracle, then the optimization problem

$$\begin{array}{ll} \text{minimize} & \sum_i \mathbf{c}^T \mathbf{x} \\ \text{subject to} & \\ & A\mathbf{x} \geq b \\ & \mathbf{x} \in C \end{array}$$

is solvable in polynomial time using the Ellipsoid Algorithm.

It remains to see how to put the Arora-Rao-Vazirani relaxation into the above form.

Recall that a matrix $X \in \mathbb{R}^{n \times n}$ is *positive semidefinite* if all its eigenvalues are nonnegative. We will use the set of all $n \times n$ positive semidefinite matrices as our set $C$ (thinking of them as $n^2$-dimensional vectors). If we think of two matrices $M, M' \in \mathbb{R}^{n \times n}$ as $n^2$-dimensional vectors, then their "inner product" is

$$M \bullet M' := \sum_{i,j} M_{i,j} \cdot M'_{i,j}$$

**Lemma 7** *The set of $n \times n$ positive semidefinite matrices has a separation oracle computable in time polynomial in $n$.*

Proof: Given a symmetric matrix $X$, its smallest eigenvalue is

$$\min_{\mathbf{z}\in\mathbb{R}^n,\ ||\mathbf{z}||=1} \mathbf{z}^T X \mathbf{z}$$

the vector achieving the minimum is a corresponding eigenvector, and both the smallest eigenvalue and the corresponding eigenvector can be computed in polynomial time.

If we find that the smallest eigenvalue of $X$ is non-negative, then we answer "yes." Otherwise, if $\mathbf{z}$ is an eigenvector of the smallest eigenvalue we output the matrix $A = \mathbf{z}^T\mathbf{z}$. We see that we have

$$A \bullet X = \mathbf{z}^T X \mathbf{z} < 0$$

but that, for every positive semidefinite matrix $M$, we have

$$A \bullet M = \mathbf{z}^T M \mathbf{z} \geq 0$$

□

This implies that any optimization problem of the following form can be solved in polynomial time

$$\begin{array}{ll} \text{minimize} & C \bullet X \\ \text{subject to} & \\ & A^1 \bullet X \geq b_1 \\ & \cdots \\ & A^m \bullet X \geq b_m \\ & X \succeq 0 \end{array} \tag{3}$$

where $C, A^1, \ldots, A^m$ are square matrices of coefficients, $b_1, \ldots, b_m$ are scalars, and $X$ is a square matrix of variables. An optimization problem like the one above is called a *semidefinite program.*

It remains to see how to cast the Arora-Rao-Vazirani relaxation as a semidefinite program.

**Lemma 8** *For a symmetric matrix $M \in \mathbb{R}^{n\times n}$, the following properties are equivalent:*

1. *$M$ is positive semidefinite;*
2. *there are vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$ such that, for all $i, j$, $M_{i,j} = \langle \mathbf{x}_i, \mathbf{x}_j \rangle$;*
3. *for every vector $\mathbf{z} \in \mathbb{R}^n$, $\mathbf{z}^T M \mathbf{z} \geq 0$*

Proof: That (1) and (3) are equivalent follows from the characterization of the smallest eigenvalue of $M$ as the minimum of $\mathbf{z}^T M \mathbf{z}$ over all unit vectors $\mathbf{z}$.

To see that (2) $\Rightarrow$ (3), suppose that vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n$ exist as asserted in (2), take any vector $\mathbf{z}$, and see that

$$\mathbf{z}^T M \mathbf{z} = \sum_{i,j} z(i) M_{i,j} z(j)$$

$$= \sum_{i,j,k} z(i) x_i(k) x_j(k) z(j) = \sum_k \left( \sum_i z(i) x_i(k) \right)^2 \geq 0$$

Finally, to see that (1) $\Rightarrow$ (2), let $\lambda_1, \ldots, \lambda_n$ be the eigenvalues of $M$ with multiplicities, and let $\mathbf{v}_1, \ldots, \mathbf{v}_n$ be a corresponding orthonormal set of eigenvectors. Then

$$M = \sum_i \lambda_k \mathbf{v}_k \mathbf{v}_k^T$$

that is,

$$M_{i,j} = \sum_k \lambda_k v_k(i) v_k(j) = \langle \mathbf{x}_i, \mathbf{x}_j \rangle$$

if we define $\mathbf{x}_1, \ldots, \mathbf{x}_n$ as the vectors such that $x_i(k) := \sqrt{\lambda_k} v_k(i)$. $\square$

This means that the generic semidefinite program (4) can be rewritten as an optimization problem in which the variables are the vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n$ as in part (2) of the above lemma.

$$\begin{array}{ll} \text{minimize} & \sum_{i,j} C_{i,j} \langle \mathbf{x}_i, \mathbf{x}_j \rangle \\ \text{subject to} & \\ & \sum_{i,j} A^1_{i,j} \langle \mathbf{x}_i, \mathbf{x}_j \rangle \geq b_1 \\ & \cdots \\ & \sum_{i,j} A^m_{i,j} \langle \mathbf{x}_i, \mathbf{x}_j \rangle \geq b_m \\ & \mathbf{x}_i \in \mathbb{R}^d \qquad\qquad \forall i \in \{1, \ldots, n\} \end{array} \tag{4}$$

where the dimension $d$ is itself a variable (although one could fix it, without loss of generality, to be equal to $n$). In this view, a semidefinite program is an optimization problem in which we wish to select $n$ vectors such that their pairwise inner products satisfy certain linear inequalities, while optimizing a cost function that is linear in their pairwise inner product.

The square of the Euclidean distance between two vectors is a linear function of inner products

$$||\mathbf{x}-\mathbf{y}||^2 = \langle \mathbf{x}-\mathbf{y}, \mathbf{x}-\mathbf{y}\rangle = \langle \mathbf{x},\mathbf{x}\rangle - 2\langle \mathbf{x},\mathbf{y}\rangle + \langle \mathbf{y},\mathbf{y}\rangle$$

and so, in a semidefinite program, we can include expressions that are linear in the pairwise squared distances (or squared norms) of the vectors. The ARV relaxation can be written as follows

$$\begin{array}{lll}
\text{minimize} & \sum_{u,v} A_{u,v} ||\mathbf{x}_u - \mathbf{x}_v||^2 & \\
\text{subject to} & & \\
& \sum_{u,v} ||\mathbf{x}_u - \mathbf{x}_v||^2 = \frac{|V|^2}{2|E|} & \\
& ||\mathbf{x}_u - \mathbf{x}_v||^2 \leq ||\mathbf{x}_u - \mathbf{x}_w||^2 + ||\mathbf{x}_w - \mathbf{x}_v||^2 & \forall u,v,w \in V \\
& \mathbf{x}_u \in \mathbb{R}^d & \forall u \in V
\end{array}$$

and so it is a semidefinite program, and it can be solved in polynomial time.

**Remark 9** *Our discussion of polynomial time solvability glossed over important issues about numerical precision. To run the Ellipsoid Algorithm one needs, besides the separation oracle, to be given a ball that is entirely contained in the set of feasible solutions and a ball that entirely contains the set of feasible solutions, and the running time of the algorithm is polynomial in the size of the input, polylogarithmic in the ratio of the volumes of the two balls, and polylogarithmic in the desired amount of precision. At the end, one doesn't get an optimal solution, which might not have a finite-precision exact representation, but an approximation within the desired precision. The algorithm is able to tolerate a bounded amount of imprecision in the separation oracle, which is an important feature because we do not have exact algorithms to compute eigenvalues and eigenvectors (the entries in the eigenvector might not have a finite-precision representation).*

The Ellipsoid algorithm is typically not a practical algorithm. Algorithms based on the *interior point* method have been adapted to semidefinite programming, and run both in worst-case polynomial time and in reasonable time in practice.

Arora and Kale have developed an $\tilde{O}((|V|+|E|)^2/\epsilon^{O(1)})$ time algorithm to solve the ARV relaxation within a multiplicative error $(1+\epsilon)$. The dependency on the error is worse than that of generic algorithms, which achieve polylogarithmic dependency, but this is not a problem in this application, because we are going to lose an $O(\sqrt{\log |V|})$ factor in the rounding, so an extra constant factor coming from an approximate solution of the relaxation is a low-order consideration.