Max-cut problem

Source: Matoušek semidefinite programming

Input: Graph $G = (V, E)$
Output: $S \subset V$ such that $|\delta(S)| = |\{uv \in E : u \in S \text{ and } v \notin S\}|$ is maximized.

Max-cut problem is NP-complete

α-approximation algorithm is algorithm that provides α multiplicative good solution.

- x be instance of problem
- $OPT(x)$ be value of optimal solution to x
- $f(x)$ be value computed by algorithm f
- f is α-approximation if

$$\forall x \begin{cases} OPT(x) \leq f(x) \leq \alpha \cdot OPT(x) & \text{if } \alpha > 1 \\ \alpha \cdot OPT(x) \leq f(x) \leq OPT(x) & \text{if } \alpha < 1 \end{cases}$$

Example Randomized 0.5-approximation algorithm for max-cut.
For every $v \in V$ pick with probability 0.5 to insert v to S.

1: Show that in expectation, the algorithm is 0.5-approximation algorithm.

Solution: Every edge is chosen with probability 0.5. So linearity of expectation gives we take half of the edges and maxcut has at most all the edges. There exists a deterministic way of doing it base on degrees.

0.878-Approximation Algorithm by Goemans-Williamson (1995) Outline:

- create a program (P) with integer variables
- find a relaxation (P') of (P) without integer variables
- express (P') as a semidefinite program and solve it
- use a smart way of rounding the solution of (SDP)

2: Show that the following program (P) is solving max-cut for a graph $G = (V, E)$:

$$(P) \begin{cases} \text{maximize} & \sum_{ij \in E} \frac{1 - x_i x_j}{2} \\ \text{subject to} & x_i \in \{-1, 1\} \quad \text{for } i = 1, \ldots, |V| = n \end{cases}$$

Solution: We define $x_i \in S$ if $x_i = 1$ and $x_i \notin S$ if $x_i = -1$. Notice that the objective function is indeed counting every edge with exactly one endpoints in S once and no other edges.

Now we would like to do a relaxation. An obvious one is to have $x \in [-1, 1]$. But this does not work well with rounding.

Instead, we map x_i to a unit n-dimensional vector.

$$x_i \to u_i \in S^{n-1} = \{u \in \mathbb{R}^n : ||u|| = 1\}$$

Notice that $\{-1, 1\}$ could be seen as a sphere in $\mathbb{R}^1$.

3: Try to formulate (P'), which is the relaxation of (P) to a vector program.

Solution:

$$(P')\begin{cases}\text{maximize} & \sum_{ij\in E}\frac{1-\mathbf{u}_i^T\mathbf{u}_j}{2}\\ \text{subject to} & u_i\in S^{n-1}\end{cases}$$

4: Show that $OPT(P)\leq OPT(P')$.

Solution: If we have a feasible solution of (P), we can always map x_i to vector $u_i=(x_i,0,0,\ldots,0)$. So it is indeed a relaxation.

Now our goal is to write (P') as a semidefinite program.

Let

$$y_{i,j}=\mathbf{u}_i^T\mathbf{u}_j$$

Now notice that if we put $y_{i,j}$ into a matrix Y and and u_i's form a column of matrix U, we get

$$Y=U^TU.$$

Recall from linear algebra that Y is positive semidefinite iff exists U such that $Y=U^TU$. From positive semidefinite Y, one can obtain U by Cholesky factorization.

5: Show that the following (SDP) is solving the same problem as (P'):

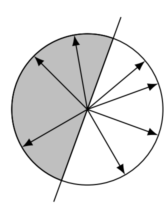
$$(SDP)\begin{cases}\text{maximize} & \sum_{ij\in E}(1-y_{i,j})/2\\ \text{subject to} & y_{i,i}=1\text{ for all } i\\ & Y\succeq 0\end{cases}$$

Solution: obvious, we just did it

Now we solve (SDP) with ε error.

Rounding to $\{-1,1\}$

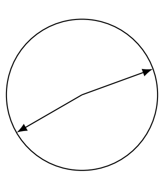
Idea: Randomly pick a halfplane going trough origin an cut the sphere into two halves. One goes to $+1$ and the other one to -1.



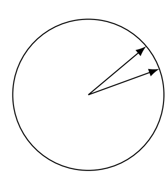
Formally, pick $\mathbf{p}\in S^{n-1}$ randomly and map

$$\mathbf{u}\to\begin{cases}1 & \text{if } \mathbf{p}^T\mathbf{u}\geq 0\\ -1 & \text{otherwise}\end{cases}$$

Is there a chance that the rounding is good?



$\begin{cases}\text{large contribution to objective}\\ \text{likely to be separated}\end{cases}$



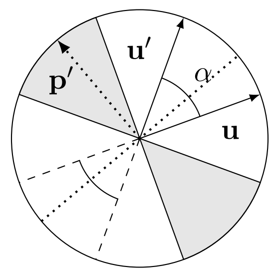
$\begin{cases}\text{small contribution to objective}\\ \text{not likely to be separated}\end{cases}$

Lemma: Let $\mathbf{u}, \mathbf{u}' \in S^{n-1}$. The probability that $\mathbf{u}$ and $\mathbf{u}'$ are mapped into different values is

$$\frac{1}{\pi} \arccos \mathbf{u}^T \mathbf{u}'.$$

6: Prove the lemma

Solution: $\cos\alpha = \mathbf{u}^T \mathbf{u}'$ hence $\alpha = \arccos \mathbf{u}^T \mathbf{u}$. Now chance of separation by a hyperplane given by a random vector $\mathbf{p}$ is $\frac{2\alpha}{2\pi}$. We can always look in the 2D situation when we project $\mathbf{p}$ to the plane given by $\mathbf{u}$ and $\mathbf{u}'$.



We want to estimate $\mathbb{E}\left(\sum \frac{1}{\pi} \arccos \mathbf{u}_i^T \mathbf{u}_j\right)$ but we only know $\sum (1 - \mathbf{u}_i^T \mathbf{u}_j)/2$

Lemma

$$\frac{1}{\pi} \arccos z \geq 0.8785(1-z)/2$$

for $z \in [-1, 1]$.

Conclusion:

$$\sum_{i,j \in E} \frac{1}{\pi} \arccos \mathbf{u}_i^T \mathbf{u}_j \geq 0.8785 \sum (1 - \mathbf{u}_i^T \mathbf{u}_j)/2 \geq 0.8785 \cdot (OPT(P) - \varepsilon) \geq 0.878 \cdot OPT(P)$$

Note: The approximation is best possible if Unique Games Conjecture holds.