Large-Scale Optimization and Applications Course Project Combinatorial optimization: Max-Cut, Min Uncut and Sparsest Cut Problems

M. Danilova, N. Puchkin, A. Shilova

28 March 2017

1 Combinatorial optimization problems

We consider three types of combinatorial optimization problems: Max-Cut, Min UnCut and uniform Sparsest Cut. Detailed formulation of these problems is given below in corresponding subsections. Now we only mention, that all these problems are NP-hard and cannot be solved efficiently if $P \neq NP$.

1.1 Max-Cut problem

The maximal cut problem or Max-Cut can be formulated as follows: given an undirected weighted graph $G = (V, E, W)$, where $V = \{1, \dots, n\}$ is a set of vertices, $E \subseteq V \times V$ is a set of edges and W is a matrix of weights, one wants to find a partition $(S, V \backslash S)$ maximizing the sum of weights of edges in the cut. This problem can be formulated as a combinatorial optimization

$$\begin{cases} \frac{1}{2} \sum\limits_{(i,j)\in E} w_{ij}(1 - x_i x_j) \longrightarrow \max \\ x_i \in \{-1, 1\}, \quad i = \overline{1, n} \end{cases} \tag{1}$$

This problem is indeed Max-Cut, since the partition can be defined by the formula $S = \{i : x_i \geq 0\}$. An edge (i, j) belongs to cut if and only if $x_i x_j = -1$. The Max-Cut problem 1 can be rewritten as follows:

$$\begin{cases} \frac{1}{4} \vec{1}^T W \vec{1} - \frac{1}{4} x^T W x \longrightarrow \max \\ x \in \{-1, 1\}^n, \end{cases} \tag{2}$$

where $\vec{1}$ stands for the vector of ones.

1.2 Min UnCut problem

The minimal uncut problem is a complement of the Max-Cut problem and can be formulated as follows. Given an undirected weighted graph $G = (V, E, W)$, where $V = \{1, \dots, n\}$ is a set of vertices, $E \subseteq V \times V$ is a set of edges and W is a matrix of weights, one wants to find a partition $(S, V \backslash S)$ minimizing the sum of weights of edges in S and $V \backslash S$. In terms of optimization this problem can be formulated as follows:

$$\begin{cases} \frac{1}{4} \vec{1}^T W \vec{1} + \frac{1}{4} x^T W x \longrightarrow \min \\ x \in \{-1, 1\}^n \end{cases} \tag{3}$$

Indeed, the sum of all edges in the graph G is given by the formula $\frac{1}{2}\vec{1}^T W\vec{1}$. Subtracting the sum of weights of edges in the cut $\frac{1}{4}\vec{1}^T W\vec{1} - \frac{1}{4}x^T Wx$, one arrives at the formulation 3.

1.3 Sparsest Cut problem

Finally, we give a formulation of the uniform Sparsest Cut problem. Let $G = (V, E)$, $|V| = n$ stand for a undirected graph and let $S \subseteq V$. Denote $E(S)$ a set of edges in G belonging to the cut $(S, V\backslash S)$. Then the sparsest cut problem can be formulated as follows:

$$\frac{|E(S)|}{|S| \cdot |V\backslash S|} \to \min_S \tag{4}$$

Sparsest Cut is related to the following problem. Denote $\varphi(S) = \frac{|E(S)|}{|S|}$. The number $\varphi(S)$ is called a sparsity of the cut S. The problem of sparsity minimization is formulated as follows:

$$\varphi(S)\frac{|E(S)|}{|S|} \longrightarrow \min_{S:|S| \leq \frac{n}{2}} \tag{5}$$

Since for every cut S, $|S| \leq \frac{n}{2}$, it holds $\frac{n}{2} \leq |V\backslash S| \leq n$, then obviously

$$\frac{\varphi(S)}{n} \leq \frac{|E(S)|}{|S| \cdot |V\backslash S|} \leq \frac{2\varphi(S)}{n}$$

and the problem 5 can be considered as an upper-bound of the problem 4 up to a factor $\frac{2}{n}$.

Modify the problem 4. Consider $x \in -1, 1^n$ and define $S = \{i : x_i > 0\}$. Then the number of edges in the cut $|E(S)|$ can be found according to the formula

$$|E(S)| = \frac{1}{4}\vec{1}^T A\vec{1} - \frac{1}{4}x^T Ax,$$

where A is an adjacency matrix of the graph G. Now we find expressions for $|S|$ and $|V\backslash S|$. Note, that

$$\sum_{i=1}^{n} x_i = |S| - |V\backslash S|$$

Then $|S|$ and $|V\backslash S|$ can be found from the system of equations

$$\begin{cases} |S| - |V\backslash S| = \sum\limits_{i=1}^{n} x_i \\ |S| + |V\backslash S| = n \end{cases}$$

Solving this system, one obtains

$$|S| = \frac{1}{2}\left(n + \sum_{i=1}^{n} x_i\right) \tag{6}$$

$$|V| = \frac{1}{2}\left(n - \sum_{i=1}^{n} x_i\right) \tag{7}$$

Then the problem 4 can be formulated as follows:

$$\begin{cases} \frac{\vec{1}^T A\vec{1} - \frac{1}{4}x^T Ax}{n^2 - \left(\sum\limits_{i=1}^{n} x_i\right)^2} \longrightarrow \min \\ x \in \{-1, 1\}^n \end{cases} \tag{8}$$

2 Greedy algorithm

In this section we provide a naive greedy algorithm of combinatorial optimization. Advantage of this method is simple implementation. The pseudocode is given below

Algorithm 1 Naive greedy algorithm

```
Require: $F(\cdot)$ – a functional to minimize
Ensure: $x_{greedy}$, $F(x_{greedy})$ – optimal values
Initialize $x$ randomly
while True do
    Get a neighbourhood $\mathcal{N}(x)$ of the point $x$
    Find $y^* = \arg \min_{y \in \mathcal{N}(x) \setminus \{x\}} F(y)$
    if $F(y^*) \leq F(x)$ then
        $x \leftarrow y^*$
    else
        break
return $x$, $F(x)$
```

Objectives $F(\cdot)$ for Max-Cut, Min UnCut and Sparsest Cut can be taken from problems 2, 3 and 8 respectively with addition a factor -1, if necessary. It remains to describe, how we choose a neighbourhood $\mathcal{N}(x)$ of a point x. For arbitrary $x \in \{-1, 1\}^n$ define $\mathcal{N}(x) = \{y : \|y - x\|_1 \leq 2\}$. In other words, we call y a neighbour of x, if it differs from x no more, than in one position. Since initial point is generated randomly, one can run the greedy algorithm several times and choose the best value of the objective.

It is easy to observe, that in case of Max-Cut the greedy algorithm on each step chooses a vertex v, such that the number of edges from v in the cut is less, than a number of edges from v outside the cut. Similarly in case of Min UnCut the greedy algorithm on each step chooses a vertex v, such that the number of edges from v outside the cut is less, than a number of edges from v in the cut. Thus, in these cases the greedy algorithm gives a 2-approximation of optimal solution.

3 SDP relaxations

3.1 SDP relaxation of Max - Cut problem

As it was already written earlier the Max-Cut can be formulated as the problem 2. In such formulation it is not convex as x can only be equal to -1 or to 1. Let's notice that $\forall x \hookrightarrow x^T W x = \text{Tr}(W x x^T)$, therefore if we introduce $X = xx^T$ and $\mathbb{I}$ a matrix containing only ones as elements, then the whole optimisation problem can be rewritten in the following way

$$\begin{cases} \frac{1}{4}\left(\text{Tr}(W\mathbb{I}) - \text{Tr}(WX)\right) \longrightarrow \max, \\ \text{diag}(X) = 1, \\ X \succeq 0, \\ \text{rank}(X) = 1, \\ X_{i,j} \in \{-1, 1\} & \forall i, j. \end{cases}$$

It is still not a convex problem, but we will obtain one if the last two constraints are thrown away.

Hence, the SDP relaxation of Max-Cut looks like

$$\begin{cases} \frac{1}{4}\left(\mathrm{Tr}(W\mathbb{I}) - \mathrm{Tr}(WX)\right) \longrightarrow \max \\ \mathrm{diag}(X) = 1, \\ X \succeq 0. \end{cases} \tag{9}$$

3.2 Min-UnCut relaxation

For Min-UnCut SDP relaxation we used the idea described in [1]. In the lecture it was suggested to use new graph $G'(V', E')$. In order to construct it, new vertices $\{-1, \dots, -n\}$ should be added to the initial graph. We connect two vertices i and j with an edge in G' if and only if i and $-j$ or $-i$ and j are connected with an edge in G. Then the initial problem of finding the cut such that the sum of weights of edges not in the cut was minimal is equivalent to the problem of finding the minimal symmetric cut in G', i.e. $(S', T' = -S')$ and $S' \cup T' = V'$, where $S' \equiv \{-i| \ i \in S'\}$. That is true if every cut (S, T) in the graph G corresponds to the cut $(S \cup (-T), (-S) \cup T)$ in G' and on the other hand if we have a cut (S', T') in G', then one can easily get the cut in G, where $S = \{i| \ i \geq 0, \ i \in S'\}$ and $T = \{i| \ i \geq 0, \ i \in T'\}$. We can exploit it and then our problem is equivalent to

$$\begin{cases} \frac{1}{8}\left(\mathrm{Tr}(\tilde{W}\mathbb{I}) - \mathrm{Tr}(\tilde{W}X)\right) \longrightarrow \min, \\ \mathrm{diag}(X) = 1, \\ X_{i,-i} = -1, \\ X \succeq 0, \\ \mathrm{rank}(X) = 1, \\ X_{i,j} \in \{-1, 1\} \qquad\qquad \forall i, j. \end{cases}$$

Here we took into account the fact that in the extended graph the number of edges doubles, therefore the coefficient $\frac{1}{8}$ is present. X_{ij} has the similar meaning as for Max-Cut problem, but $X_{ij} = -1$ if and only if vertices i and j are separated by the cut in G'. Moreover, in this statement

$$\tilde{W} = \begin{pmatrix} O & W \\ W & O \end{pmatrix}$$

where W is the matrix of weights for G. At last, the third constraint binds our cut to be symmetric or to satisfy $(S', T' = -S')$.

As in the previous case two last constraints prevent this problem from being convex, hence, getting rid of them, we will have the SDP relaxation for Min-UnCut problem.

$$\begin{cases} \frac{1}{8}\left(\mathrm{Tr}(\tilde{W}\mathbb{I}) - \mathrm{Tr}(\tilde{W}X)\right) \longrightarrow \min, \\ \mathrm{diag}(X) = 1, \\ X_{i,-i} = -1, \\ X \succeq 0. \end{cases} \tag{10}$$

3.3 Sparsest Cut relaxation

Basing on (8),

3.4 Relaxations with addition of triangle constraints

If we introduce the metric for our partitioning tasks:

$$d(i,j) = \begin{cases} 0 & \text{if } i \text{ and } j \text{ are vertices in the same part of the graph,} \\ 1 & \text{otherwise,} \end{cases}$$

which satisfies all the metric conditions, then the variables in the problems mentioned above are connected with the metric in the following way

$$X_{ij} = 1 - 2d(i,j).$$

Taking into account this connection, one can obtain the triangle constraints with our variables, writing those for the metric itself:

$$d(i,j) + d(j,k) \geq d(i,k) \Leftrightarrow X_{ij} + X_{jk} - 1 \leq X_{ik}.$$

Those expressions should be true for any i, j and k. Let's denote for any j $E_j = (\mathbf{e}_j, \dots, \mathbf{e}_j)$ and $\mathbf{e}_j$ is a unit vector with 1 on the j-th place. From this follows that the constraint above can be rewritten as following:

$$XE_j + E_j^T X - 1 \leq X \qquad \forall j.$$

Beside all the mentioned above, one could notice that for the metric $d(i,j)$ it also holds the fact that

$$d(i,j) + d(j,k) + d(i,k) \leq 2 \qquad \forall i, j, k.$$

Therefore, applying the link between the metric and X, we have

$$X_{ij} + X_{jk} + X_{ik} \geq -1 \qquad \forall i, j, k.$$

Rewriting it in the matrix way, we will derive the last triangle constraint:

$$XE_j + E_j^T X + X \geq -1 \qquad \forall j.$$

3.5 Primal-dual central path tracing interior point method

Consider an SDP problem

$$\begin{cases} \langle C, X \rangle \longrightarrow \min \\ X \in \mathcal{L} - B \\ X \succeq 0, \end{cases} \tag{11}$$

where $\langle C, X \rangle = \operatorname{Tr}(C^T X)$. The dual problem can be formulated as follows

$$\begin{cases} \langle B, S \rangle \longrightarrow \max \\ S \in \mathcal{L}^\perp + C \\ S \succeq 0 \end{cases} \tag{12}$$

If problems 11 and 12 are strictly feasible, then the strong duality takes place. We describe an interior point method based on primal-dual central path tracing. Consider a pair of modified primal and dual problems

$$X_\mu = \arg\min_{X \in \mathcal{L}} \langle C, X \rangle - \mu \log\det X \tag{13}$$

$$S_\mu = \arg\max_{S \in \mathcal{L}^\perp} \langle B, X\rangle + \mu \log\det S \tag{14}$$

It is known, that $X_\mu S_\mu = S_\mu X_\mu = I$, and duality gap will be equal to $\langle X_\mu, S_\mu\rangle = \mu n$. The set $\{(X_\mu, S_\mu) : \mu > 0\}$ defines a so-called primal-dual central path. If one can move along the primal-dual central path, then one can easily solve optimization problems 11, 12 with arbitrary predefined accuracy tending μ to zero. Unfortunately, we cannot stay on the primal-dual central path, but we can try to move in its vicinity. The following algorithm allows to trace the primal-dual central path.

```
Algorithm 2 Primal-dual central path tracing IPM
Require: Matrices $B$ and $C$ as in problems 11, 12; projectors $\pi_{\mathcal{L}}(\cdot, Q)$, $\pi_{\mathcal{L}^\perp}(\cdot, Q)$ on linear spaces $Q\mathcal{L}Q$
    and $Q^{-1}\mathcal{L}^\perp Q^{-1}$; parameters $0 \le \chi \le \kappa \le 0.1$; initial values $X \in \mathcal{L} - B$, $S \in \mathcal{L}^\perp + C$, $X, S \succ 0$ and $t$,
    which fulfil
        $\|tX^{\frac{1}{2}}SX^{\frac{1}{2}} - I\|_2 \le \kappa;$
    tolerance $\varepsilon$ and (or) number of iterations $N_{\max}$
Ensure: $\varepsilon$-optimal solution $X$
Compute duality gap $\langle X, S\rangle$
while (Duality gap $> \varepsilon$) or (number of iterations $< N_{\max}$) do
    $t \leftarrow \left(1 - \frac{\chi}{\sqrt{k}}\right)^{-1} t$
    Choose a Nesterov-Todd scaling matrix
        $Q = P^{\frac{1}{2}}$    (15)
        $P = X^{-\frac{1}{2}}\left(X^{-\frac{1}{2}}S^{-1}X^{-\frac{1}{2}}\right)^{\frac{1}{2}} X^{\frac{1}{2}}S$    (16)
    $\hat{X} \leftarrow QXQ$
    Solve a Lyapunov matrix equation
        $\hat{X}Z + Z\hat{X} = \frac{2}{t}I - 2\hat{X}^2$    (17)
    $\Delta\hat{X} \leftarrow \pi_{\mathcal{L}}(Z, Q)$, $\Delta\hat{S} \leftarrow \pi_{\mathcal{L}^\perp}(Z, Q)$
    $\Delta X \leftarrow Q^{-1}\Delta\hat{X}Q^{-1}$, $\Delta S \leftarrow Q\Delta\hat{S}Q$
    Update $X \leftarrow X + \Delta X$, $S \leftarrow S + \Delta S$, $t \leftarrow \frac{k}{\langle X,S\rangle}$
    Compute duality gap $\langle X, S\rangle = \frac{k}{t}$
return $X$
```

The Nesterov-Todd scaling matrix defined by 15 – 16 ensures, that matrices $\hat{X} = QXQ$ and $\hat{S} = Q^{-1}SQ^{-1}$ commute and moreover $\hat{X} = \hat{S}$. The equation 17 can be considered as a linearization of the equation

$$\tilde{X}\tilde{S} + \tilde{S}\tilde{X} = \frac{2}{t}I$$

at the point $(\hat{X}, \hat{S}) = (QXQ, Q^{-1}SQ^{-1})$ w.r.t. X and S, and $Z = Q\Delta XQ + Q^{-1}\Delta SQ^{-1}$. Note, that the algorithm guarantees, that $\Delta X \in \mathcal{L}$ and $\Delta S \in \mathcal{L}^\perp$, so the output X is feasible. We refer to lectures of Ben-Tal and Nemirovski [2] for more details.

The next theorem guarantees consistency of the procedure 2.

Theorem 1 *4.5.2 in [2] Let the parameters χ, κ of 2 satisfy*

$$0 < \chi \leq \kappa \leq 0.1$$

Let (X, S) be a pair of strictly feasible primal and dual solutions to 11, 12, such that the triple (X, S, t) satisfies

$$\|tX^{\frac{1}{2}}SX^{\frac{1}{2}} - I\|_2 \leq \kappa; \tag{18}$$

Then the updated pair $(X+\Delta X, S+\Delta S)$ is well-defined (i. e. equation 17 has a unique solution), $X+\Delta X$, $S+\Delta S$ are strictly feasible solutions to 11, 12 respectively, and the triple $\left(X+\Delta X, S+\Delta S, t = \frac{k}{\langle X,S\rangle}\right)$ satisfies 18.

References

[1] Sanjeev Arora, *Lecture 8: Approximating Min UnCut and Min-2CNF Deletion*, (2005)
`http://www.cs.princeton.edu/courses/archive/spr05/cos598B/lecture8.pdf`

[2] A. Ben-Tal, A. Nemirovski, *Lectures on Modern Convex Optimization*, (2013)
`http://www2.isye.gatech.edu/~nemirovs/Lect_ModConvOpt.pdf`