**Example 1.1.1.** Let $G = (V, E)$ be a simple graph. We would like to partition the nodes of $G$ into two complementary sets $S$ and $T$ such that the number of edges between $S$ and $T$ is maximized. This problem can be modelled as an LP with binary variables, however, the number of constraints is exponential. To avoid this, we will take the following formulation, where $x_i = 1$ if $i \in S$ and $x_i = -1$ if $i \in T$:

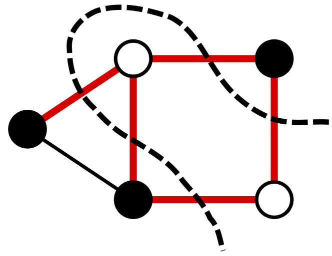

Figure 1.1: An example of a maximal cut

$$\max \sum_{(i,j)\in E} \frac{1 - x_i x_j}{2} \quad \text{s.t} \quad x_v \in \{-1, 1\}, \ \forall v \in V$$

Using the degree sum formula on the first term and rearranging the second sum we can rewrite the objective as

$$\sum_{(i,j)\in E} \frac{1 - x_i x_j}{2} = \frac{1}{2}\Big( \sum_{(i,j)\in E} 1 - \sum_{(i,j)\in E} x_i x_j \Big) = \frac{1}{2}\Big( \frac{1}{2} \sum_{v\in V} \deg(v) - \frac{1}{2} \sum_{i\in V} \sum_{\substack{j\in V \\ (i,j)\in E}} x_i x_j \Big)$$

We can rewrite the second sum using the adjacency matrix cancelling the contribution of non-existing edges:

$$\sum_{\substack{j\in V \\ (i,j)\in E}} x_i x_j = \sum_{j\in V} x_i a_{ij} x_j$$

Finally, observing that $x_i x_i = 1$ and defining $D$ as the diagonal degree matrix, we have

$$\max \sum_{(i,j)\in E} \frac{1 - x_i x_j}{2} = \max \frac{1}{4} x^T (D - A) x = \frac{1}{4} \max x^T L x, \text{ where } L = D - A$$

Using the fact that $x^T L x = x^T (Lx) = \mathrm{tr}(L x x^t)$ we can define a new variable $X = x x^T$ and rewrite the linearized objective as $\mathrm{tr}(LX)$. In order to account for the binary constraints, we notice that they can be expressed as $\mathrm{diag}(X) = e, X \succcurlyeq 0, \mathrm{rank}(X) = 1$. Since the rank constraint is not convex, we drop it and obtain the following SDP relaxation:

$$\max \ \mathrm{tr}(LX) \text{ s.t } \mathrm{diag}(X) = e, X \succcurlyeq 0$$

It has been shown that algorithms based on such SDP relaxations and randomized rounding achieve an approximation ratio of 0.878, meaning that the expected size of the obtained cut is at least 0.878 of the optimal cut. [GW95].