Lecture 3.4

Friday, 8 September 2023 16:02

In the weighted vertex-cover problem, each of the vertices is assigned a number (weight/cost). The aim is to select a subset of vertices that covers all edges such that this subset has a minimum total weight.

The 2-approximation algorithm for the unweighted vertex-cover problem does not work for the weighted version; as a counterexample, consider a graph consisting of one edge, where one node has weight 1 and the other has weight > 2.

To attempt to find an algorithm for weighted vertex cover, we first describe it as a linear program. The approach has the following elements:

- For every vertex v, a decision variable x_v is introduced. This variable is 1 if v is in the cover and 0 if v is not in the cover.
- The weighted vertex cover problem is formulated using a linear cost function and linear constraints on the decision variables.
  We want to minimize the total weight of the vertices in C subject to the constraint that the vertices in C form a cover.
  - We have that $total\ weight = \sum_{v \in V} weight(v) \cdot x_v$, which should be minimized.
  - We have that, for every edge (u, v), the constraint $x_u + x_v \geq 1$ is added.
  - A problem here is that the variables are only meaningful when their value is either 0 or 1. This can be addressed using so-called 0/1-constraints. However, such constraints are not linear. Therefore, we are solving what is known as a 0/1-LP, the solving of which is NP-hard.
    - The next lecture will show LP relaxation, which can use a 0/1-LP to build an approximation algorithm.