Prof. Dr. Christian Scheideler
Universität Paderborn

Paderborn, November 19, 2018

Advanced Algorithms

WS 2019

Homework Assignment 6

Problem 16:
Let $G = (V, E)$ be a graph with vertex set $V = \{v_1, \ldots, v_n\}$. A *vertex cover* of G is a set $C \subseteq V$ with the property that for every edge $\{v_i, v_j\} \in E$, either v_i or v_j is in C. In the VertexCover problem the task is to find the smallest possible vertex cover.

(a) Formulate the vertex cover problem as an ILP, where variable $x_i \in \{0, 1\}$ indicates whether node v_i is in the vertex cover or not.

(b) Propose a randomized rounding strategy for the optimal solution of the LP relaxation to obtain (with probability at least 1/2) a feasible solution for the original ILP (no proof is needed here). Is there also a simple deterministic rounding strategy?

Problem 17:
Max2SAT is the restriction of the MaxSAT problem to Boolean formulas in CNF that have clauses with at most 2 literals. The decision variant of the Max2SAT problem is known to be NP-hard. Consider the following arithmetization of Max2SAT.
For each Boolean variable x_i we define a variable y_i that can take the value -1 or $+1$. In addition to that we have a variable $y_0 \in \{-1, +1\}$ with the meaning that x_i is True if and only if y_i and y_0 have the same value.
For the arithmetization of a clause C with one literal, we distinguish between two cases:

- $C = x_i$: use $(1 + y_i \cdot y_0)/2$
- $C = \bar{x}_i$: use $(1 - y_i \cdot y_0)/2$

For clauses with two literals we can specify similar formulas.

1. Propose an arithmetization for all 4 possibilities for a clause with 2 literals.

   Hint: given that $a(\phi)$ is the arithmetization of a Boolean expression ϕ, $a(x_i \vee x_j) = 1 - a(\bar{x}_i \wedge \bar{x}_j) = 1 - a(\bar{x}_i) \cdot a(\bar{x}_j)$.

2. Formulate a quadratic program for Max2SAT with the help of these arithmetizations.

3. Formulate a semidefinite program as a relaxation of the quadratic program.

4. Propose an approximation algorithm for Max2SAT. Do you have an idea how to show that its approximation ratio is at most 1.139?

   Hint: use the fact that $\Pr[\text{sgn}(\vec{r}^T \cdot \vec{u}_i) = \text{sgn}(\vec{r}^T \cdot \vec{u}_j)] = 1 - \arccos(\vec{u}_i^T \cdot \vec{u}_j)/\pi$.