Prof. Dr. Christian Scheideler
Universität Paderborn

Paderborn, June 1, 2018

Randomized Algorithms

SS 2018

Homework Assignment 8

Problem 22:

Consider the standard ILP with $c \in \mathbb{R}^n_{\geq 0}$, $b \in \mathbb{R}^m_{\geq 0}$, and $A \in M(m, n; \mathbb{R}_{\geq 0})$ (i.e., all coefficients in b, c, and A are non-negative):

$$\begin{aligned} \text{maximize} \quad & c^T \cdot x \\ \text{subject to} \quad & A \cdot x \leq b \\ & x_i \in \{0,1\} \text{ for all } i \in \{1, \dots, n\} \end{aligned}$$

Suppose that we want to find out about the existence of a solution that is at least a $(1+\epsilon)$-approximate solution to the optimal fractional result (i.e., we are at most a $1+\epsilon$ factor away from the optimal result). Under which conditions (on the dependency between the constraints and the probability of violating any of them) can we show via the symmetric LLL that this is possible?

Hint: Use the random experiment following the standard randomized rounding approach to obtain an integral solution out of an optimal solution for the linear relaxation and use the Chernoff bounds to bound the deviation from the expected value.

Problem 23:

Consider the following parallel variant of the FastLLL algorithm:

> **Algorithm FastParLLL($\mathcal{P}, \mathcal{A}$):**
>   **for all** $P \in \mathcal{P}$ **do in parallel**
> $v_P :=$ a random outcome of P
>   **while** $\exists A \in \mathcal{A}$: A is true under $(P = v_P)_{P \in \mathcal{P}}$ **do**
> $S :=$ maximal independent set of true events, constructed in parallel
>   **for all** $P \in \bigcup_{A \in S} vbl(A)$ **do in parallel**
> $v_P :=$ a new random outcome of P
>   **return** $(v_P)_{P \in \mathcal{P}}$

Our goal is to show that as long as for all $A \in \mathcal{A}$,

$$\Pr[A] \leq (1-\epsilon) \cdot x(A) \prod_{B \in \Gamma_{\mathcal{A}}(A)} (1 - x(B))$$

then the algorithm finds a solution with a low expected runtime so that all $A \in \mathcal{A}$ are false.

Recall the construction of witness trees out of the execution log. Let S_j be the j-th independent set chosen by the algorithm and let $Q(j)$ be the probability that the parallel algorithm runs for at least j rounds.

(a) Show that if $t \in S_j$, then the depth of any $\tau(t)$ is $j - 1$.

(b) Based on (a), show that $Q(j) \leq (1 - \epsilon)^j \sum_{A \in \mathcal{A}} \frac{x(A)}{1 - x(A)}$.

(c) Use (b) to bound the expected runtime of the algorithm.