Complexity Theory, WS 13/14: Solution Hints 9.

Markus Bläser, Thatchaphol Saranurak

Exercise 9.1 *EquiSLP* $\leq$ ACIT Easy. EquiSLPis just a special case of ACITwithout any variables.

ACIT $\leq$ *EquiSLP* If we have given our transformed circuit P_C than we construct our SLP as $P_{C_1}^2 + \cdots + P_{C_n}^2$ where P_{C_i} is the SLP from our circuit C where we set $X_{i\neq j} = 1$ and $X_i = 2c^{2^s}$. Let us consider the polynomial of P_{C_I} in the form $p_k(X_{j\neq i})X_i^k + \cdots + p_0(X_{j\neq i})X_i^0$. This gives us the following summation of terms

$$\begin{aligned}\sum_{a=0}^{k-1} |p_a(X_{j\neq i})| X_i^a \leq & c^{2^s} \sum_{a=0}^{k-1} X_i^a \\ = & c^{2^s}(X_i^k - 1)/(X_i - 1) \\ < & c^{2^s} X_i^k/(X_i/2) \\ = & 2c^{2^s} X_i^{k-1} \\ \leq & X_i^k.\end{aligned}$$

This shows that the absolute value of the first term is larger than the summation of the other terms. Hence we keep our zero property.

Exercise 9.2 We show that we can solve {0,1}-Perm with BitSLP. For this we look at the product

$$\prod_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x^{2^{j-1}}.$$

Note that the coefficient of x^{2^n-1} is the permanent. To show this let us look at one summand of the known permanent form $\sum_{\sigma\in\text{Permutations}} \prod_{i=1}^{n} a_{i\sigma(i)}$. If you look at x^{2^n-1} we see that we need to pick up $X = \{x^1, x^2, x^4, \ldots, x^{2^{n-1}}\}$ to get x^{2^n-1}. This is exactly one permutation σ. Because of our unique construction only a permutation of X fulfills the requirement. Notice also that it does not help us to get two times the same "j".

To gather the bits of our permanent we take a large enough x such that the gap between our monomials is large enough. 2^{n^2} does the job. The reason for this is easy to see. We now that a coefficient is at most $n^n = 2^{n\log n}$ large and we need enough space from to distinguish this, meaning $x^{2^n-1} > 2^{n\log n} + x^{2^n-2}$. This works for 2^{n^2} as $(2^{n^2})^{2^n-1} > 2^{n\log n} + (2^{n^2})^{2^n-2}$ is fulfilled.

Exercise 9.3 (Csanky's algorithm) Instead of $s_{A,i}$, we just write s_i in this solution.

a) $s_0 = 1$ because, to compute determinant, the only permutation which contributes to X^n is the identity permutation and also the variable X, in $X \cdot I - A$, has no coefficient in front.

The roots of characteristic polynomial $c_A(X)$ of a matrix A are the eigenvalues $\lambda_1, \ldots, \lambda_n$. We have

$$c(X) = \prod_{k=1}^{n}(X - \lambda_k) = \sum_{k=0}^{n} s_{n-k}X^k.$$

Let e_k be the k-th elementary symmetric polynomial. Observe that $s_k = (-1)^k e_k(\lambda_1, \ldots, \lambda_n)$. Newton's identity says that we can recursively write e_k using the power sum polynomial $p_k(\lambda_1, \ldots, \lambda_n) = \sum_{j=1}^{n} \lambda_j^k$:

$$e_k = \frac{1}{k}\sum_{i=1}^{k}(-1)^{i-1}e_{k-i}p_i.$$

With further observation that $\mathrm{trace}(A^i) = p_i(\lambda_1, \ldots, \lambda_n)$, we have that

$$\begin{aligned} s_k &= (-1)^k \frac{1}{k}\sum_{i=1}^{k}(-1)^{i-1}((-1)^{k-i}s_{k-i})p_i \\ &= -\frac{1}{k}\sum_{i=1}^{k} s_{k-i}\mathrm{trace}(A^i). \end{aligned}$$

b) $s_n = (-1)^n e_n(\lambda_1, \ldots, \lambda_n) = (-1)^n \prod_{i=1}^{n} \lambda_n = (-1)^n \det(A)$.

c) Our formular says $s_k = -\frac{1}{k}\sum_{i=1}^{k} s_{k-i}\,\mathrm{trace}(A^i) = \sum_{i=0}^{k-1} \frac{-1}{k}\,\mathrm{trace}(A^{k-i}) \cdot s_i$. Let S be an $n+1$-dimensional vector where $S_k = s_k$. Let T be $(n+1) \times (n+1)$-matrix where

$$T_{k,i} = \begin{cases} -\frac{1}{k}\mathrm{trace}(A^{k-i}) & \text{if } i \le k-1 \\ 0 & \text{o.w.} \end{cases}$$

By the definition of $T_{k,i}$ and S_k, we write $S_k = \sum_{i=0}^{n} T_{k,i} \cdot S_i$, except that $S_0 = 1$. Let e_1 be zero vector except that the first element is 1. Thus, $S = TS + e$. Hence, $(I - T)S = e$.

$T_{k,i}$ can be computed using polylog depth circuit by iterated square of matrix. Finally, we can solve the linear equation $(I - T)S = e$ using polylog depth circuit because $(I - T)$ is a lower triangular matrix, and hence we can divide and conquer.