CSCI 2510 - Problem set 6

Shay Mozes
shay@cs.brown.edu

Problem 16.4

We Give a derandomized algorithm for the randomized 3/4-factor algorithm of section 16.4. Let the random variable W^f denote the total weight of satisfied clauses of a formula f, and let W_c^f denote the weight contributed by clause c of f to W. Consider the self reducibility tree T of the formula f. As discussed in class and in chapter 16, it suffices to show that we can compute the conditional expectation of any node of T. Consider the node $x_1 = a_1 \dots x_i = a_i$. Let W_i be the weight of clauses of f that are satisfied by the partial assignment. W_i can be easily computed in a single scan of all clauses. Let ϕ be the formula obtained (efficiently) by deleting these causes and restricting the remaining clauses of f to variables $x_{i+1} \dots n$. Clearly,

$$E[W|x_1 = a_1 \dots x_i = a_i] = W_i + E[W^\phi|x_1 = a_1 \dots x_i = a_i].$$

It remains to show how to compute $E[W^\phi|x_1 = a_1 \dots x_i = a_i]$ efficiently. Recall that the randomized algorithm tosses a coin b to decide whether to run the randomized 1/2-approximation or the randomized $(1-1/e)$-approximation. Since $b=0, b=1$ are exhaustive mutually disjoint events,

$$\begin{aligned} E[W^\phi|x_1 = a_1 \dots x_i = a_i] &= \\ E[W^\phi|x_1 = a_1 \dots x_i = a_i, b=0]P(b=0|x_1 = a_1 \dots x_i = a_i) &+ \\ E[W^\phi|x_1 = a_1 \dots x_i = a_i, b=1]P(b=1|x_1 = a_1 \dots x_i = a_i). \end{aligned}$$

We now show how to efficiently compute each of these quantities.

- $E[W^\phi|x_1 = a_1 \dots x_i = a_i, b=0]$: Since $b=0$ the algorithm chooses all x_is independently at random. Hence $E[W^\phi|x_1 = a_1 \dots x_i = a_i, b=0] = \sum_{c \text{ clause of } \phi} w_c(1-2^{-k_c})$, where k_c is the number of literals in c.

- $E[W^\phi|x_1 = a_1 \dots x_i = a_i, b=1]$: Since $b=1$ the algorithm chooses x_i to be true with probability $y_i^\star$. Hence,

$$\begin{aligned} E[W^\phi|x_1 = a_1 \dots x_i = a_i, b=1] &= \\ \sum_{c \text{ clause of } \phi} E[W_c^\phi|x_1 = a_1 \dots x_i = a_i, b=1] &= \\ \sum_{c \text{ clause of } \phi} w_c \left(1 - \prod_{i \in c^+}(1-y_i^\star) \prod_{i \in c^-} y_i^\star\right), \end{aligned}$$

  where $c^+(c^-)$ is the set of variables (negated variables) appearing in clause c. In the last equality we used the definition of expectation.

- $P(b = 0 | x_1 = a_1 \ldots x_i = a_i)$: This is computed using the conditional probability rule:

$$P(b = 0 | x_1 = a_1 \ldots x_i = a_i) = \frac{P(b = 0, x_1 = a_1 \ldots x_i = a_i)}{P(x_1 = a_1 \ldots x_i = a_i)}.$$

  $P(b = 0, x_1 = a_1 \ldots x_i = a_i) = 2^{-1} \cdot 2^{-i} = 2^{-(i+1)}$. Since the probability that $b = 0$ is $1/2$ and then all variables are chosen uniformaly at random. We decompose the denominator into two disjoint events and sum their probability:

$$\begin{aligned} P(x_1 = a_1 \ldots x_i = a_i) &= P(b = 0, x_1 = a_1 \ldots x_i = a_i) + P(b = 1, x_1 = a_1 \ldots x_i = a_i) \\ &= 2^{-(i+1)} + 2^{-1} \cdot \prod_{a_i \text{ is true}} y_i^\star \prod_{a_i \text{ is flase}} (1 - y_i^\star), \end{aligned}$$

  where we have used the fact that b=1 with probability 1/2 and then each x_i is set to be true independently at random with probability $y_i^\star$.

- Finally, we compute $P(b = 1 | x_1 = a_1 \ldots x_i = a_i)$ as $1 - P(b = 0 | x_1 = a_1 \ldots x_i = a_i)$.

We have thus established that the conditional expectations at each node of T may be computed efficiently. At each node we compute the two conditional expectations at each of the children of that node and set x_i according to the one with larger expectation. Since we know that the expectation at the root of the tree is at least 3/4·OPT, and since the expectation at each node is a convex combination of the conditional expectations at the children, we are guaranteed that the larger of the two conditional expectations is at least 3/4·OPT as well.