An Interactive Proof System for TQBF

This material was presented in class on March 26, 2015.

We wish to revise the interactive proof system for coSAT that was given in class on March 24, 2015, so that it works for TQBF. The existence of such a proof system implies that PSPACE is contained in IP.

We follow the argument on pages 161 and 162 of your textbook. That argument is clear until it gets to the last displayed formula on page 161. Because the displayed expression should be the fully arithmetized and linearized version of the TQBF instance ψ, it should be

$$\Pi_{X_1} L_1 \Sigma_{X_2} L_1 L_2 \cdots \Sigma_{X_n} L_1 L_2 \cdots L_n P_\phi(X_1, X_2, \ldots . X_n). \tag{1}$$

Here and throughout this lecture, the products and sums are computed over $X_i \in \{0, 1\}$, where 0 and 1 are elements of the field $\mathbb{Z}_p$, and p is a suitably large prime. (The size of p will be addressed below.) Assume without loss of generality that ϕ is a 3CNF formula on n boolean variables with m clauses.

The proof system for TQBF needs $i+1$ *segments* of interaction, say $(i.1)$ through $(i.i+1)$, between Arthur and Merlin in order to handle X_i, for $1 \leq i \leq n$, and each segment requires $O(1)$ rounds. So the entire execution requires $O(n^2)$ rounds of interaction. For X_i, the first segment handles the operator Π_{X_i}, if i is odd, and it handles the operator Σ_{X_i}, if i is even. Subsequent segments of interaction for X_i handle the operators L_1 through L_i.

Merlin's original claim is that the formula ψ is true, which is equivalent to

$$\Pi_{X_1} L_1 \Sigma_{X_2} L_1 L_2 \cdots \Sigma_{X_n} L_1 L_2 \cdots L_n P_\phi(X_1, X_2, \ldots . X_n) \equiv C \bmod p, \tag{1.1}$$

where $C \neq 0$.

We now specify the first few segments of the proof system in detail:

Segment 1.1:
Note that Merlin's claim (congruence (1.1) above) is equivalent to the claim that $\Pi_{X_1}(h_1^1(X_1)) \equiv C \bmod p$, where h_1^1 is the linear, univariate polynomial in X_1 that results from evaluating all of the operators in congruence (1.1) except Π_{X_1}.

Arthur challenges Merlin to send him $h_1^1(X_1)$. Merlin sends him a linear, univariate polynomial $s_1^1(X_1)$. As in the sum-check protocol used in the proof system for coSAT, s_1^1 will be equal to h_1^1 if and only if Merlin is making a correct claim.

Arthur checks that $s_1^1(0) \cdot s_1^1(1) \equiv C \bmod p$; he rejects and halts the protocol if this check fails. Otherwise, he chooses a_1^1 uniformly at random from $\mathbb{Z}_p$ and sends it to Merlin.

Segment 1.2:
Merlin's claim is that

$$L_1 \Sigma_{X_2} L_1 L_2 \cdots \Sigma_{X_n} L_1 L_2 \cdots L_n P_\phi(a_1^1, X_2, \ldots . X_n) \equiv s_1^1(a_1^1) \bmod p. \tag{1.2}$$

This is equivalent to the claim that $L_1(h_1^2(a_1^1)) \equiv s_1^1(a_1^1) \bmod p$, where $h_1^2(X_1)$ is the quadratic,[1] univariate polynomial in X_1 that results from evaluating all of the operators in congruence (1.2) except the leftmost L_1.

[1] The question of why (and, in fact, whether) h_1^2 is quadratic is addressed below.

Arthur challenges Merlin to send him $h_1^2(X_1)$. Merlin sends him a quadratic, univariate polynomial $s_1^2(X_1)$. It will be equal to h_1^2 if and only if Merlin is making a correct claim.

Arthur checks that $(1-a_1^1)\cdot s_1^2(0)+a_1^1\cdot s_1^2(1) \equiv s_1^1(a_1^1) \bmod p$; he rejects and halts the protocol if this check fails. Otherwise, he chooses a_1^2 uniformly at random from $\mathbb{Z}_p$ and sends it to Merlin.

Segment 2.1:
Merlin's claim is that

$$\Sigma_{X_2} L_1 L_2 \cdots \Sigma_{X_n} L_1 L_2 \cdots L_n P_\phi(a_1^2, X_2, \ldots . X_n) \equiv s_1^2(a_1^2) \bmod p. \qquad (2.1)$$

This is equivalent to the claim that $\Sigma_{X_2}(h_2^1(X_2)) \equiv s_1^2(a_1^2) \bmod p$, where h_2^1 is the linear, univariate polynomial in X_2 that results from evaluating all of the operators in congruence (2.1) except Σ_2.

Arthur challenges Merlin to send him h_2^1. Merlin sends him a linear, univariate polynomial s_2^1 in X_2. It will be equal to h_2^1 if and only if Merlin is making a correct claim.

Arthur checks that $s_2^1(0) + s_2^1(1) \equiv s_1^2(a_1^2) \bmod p$; he rejects and halts the protocol if this check fails. Otherwise, he chooses a_2^1 uniformly at random from $\mathbb{Z}_p$ and sends it to Merlin.

Segment 2.2:
Merlin's claim is that

$$L_1 L_2 \cdots \Sigma_{X_n} L_1 L_2 \cdots L_n P_\phi(a_1^2, a_2^1, \ldots . X_n) \equiv s_2^1(a_2^1) \bmod p. \qquad (2.2)$$

This is equivalent to the claim that $L_1(h_2^2(a_1^2)) \equiv s_2^1(a_2^1) \bmod p$, where h_2^2 is the quadratic, univariate polynomial in X_1 that results from evaluating all of the operators in congruence (2.2) except the leftmost L_1.

Arthur challenges Merlin to send him h_2^2. Merlin sends him s_2^2. It will be equal to h_2^2 if and only if Merlin is making a correct claim.

Arthur checks that $(1-a_1^2)\cdot s_2^2(0)+a_1^2\cdot s_2^2(1) \equiv s_2^1(a_2^1) \bmod p$; he rejects and halts the protocol if this check fails. Otherwise, he chooses a_1^3 uniformly at random from $\mathbb{Z}_p$ and sends it to Merlin.

Segment 2.3:
Merlin's claim is that

$$L_2 \cdots \Sigma_{X_n} L_1 L_2 \cdots L_n P_\phi(a_1^3, a_2^1, \ldots . X_n) \equiv s_2^2(a_1^3) \bmod p. \qquad (2.3)$$

This is equivalent to the claim that $L_2(h_2^3(a_2^1)) \equiv s_2^2(a_1^3) \bmod p$, where h_2^3 is the quadratic, univariate polynomial in X_2 that results from evaluating all of the operators in congruence (2.3) except the leftmost L_2.

Arthur challenges Merlin to send him h_2^3. Merlin sends him s_2^3. It will be equal to h_2^3 if and only if Merlin is making a correct claim.

Arthur checks that $(1-a_2^1)\cdot s_2^3(0)+a_2^1\cdot s_2^3(1) \equiv s_2^2(a_1^3) \bmod p$; he rejects and halts the protocol if this check fails. Otherwise, he chooses a_2^2 uniformly at random from $\mathbb{Z}_p$ and sends it to Merlin.

Segment 3.1:
Merlin's claim is that

$$\Pi_{X_3} L_1 L_2 L_3 \cdots \Sigma_{X_n} L_1 L_2 \cdots L_n P_\phi(a_1^3, a_2^2, \ldots . X_n) \equiv s_2^3(a_2^2) \bmod p. \qquad (3.1)$$

... and so forth.

Left as exercises for you are the specifications of segments (i, j), for $3 \leq i \leq n$ and $i \leq j \leq n$, and the inductive proof of the correctness of the upper bound on the error probability (given on page 162 of your textbook) in the case that Merlin is making a false claim.

In the case of a segment $i.j$ in which the operator being handled is L_{j-1}, why might the polynomial h_i^j be quadratic? Reading the sequence of operators in the congruence $(i.j)$ from right to left, consider what has happened as all operators except the leftmost L_{j-1} have been evaluated. After the previous L_{j-1} was evaluated (at which point the partial result was a multivariate polynomial that was linear in X_{j-1}), an arithmetic operator was applied. If that was a product operator, then it produced a multivariate polynomial that is quadratic in X_{j-1}. This degree doubling does not occur if the intervening arithmetic operator is a sum. However, it is more convenient to have a uniform specification for the protocol segments, and we can do so if we make the worst-case assumption that h_i^j is quadratic. A linear polynomial is in fact just a quadratic polynomial with leading coefficient 0, and applying the linearization operator to a polynomial that is already linear just leaves it unchanged.

It remains to explain why it suffices to use a prime p that is singly exponential in n and m (i.e., a p that can be represented using a number of bits that is polynomial in n and m). If we use the arithmetization from the interactive proof system for #SAT in Section 8.3.2 (not the arithmetization of Lecture 16 that we used in an interactive proof system for coSAT), then (1), evaluated over $\mathbb{Z}$, is equal to 0 if ψ is false and 1 if ψ is true. Therefore, wraparound is not an issue, and (1) has the correct value modulo p for any p. Because the error probability is 0 in the case that Merlin is making a correct claim and $\frac{poly(n,m)}{p}$ in the case that Merlin is lying, we can use a p that is singly exponential in n and m.