Collaborators : None

Sources : Lecture Notes

Q2) Implausible consequences of superstrong nonlocality

a) Given: A & B have access to 'N' magic "non-local boxes" such that $\boxed{a_i + b_i = x_i \cdot y_i \pmod 2}$ $\forall\ i \in \{1, 2, \dots, N\}$

To prove: There is a way to compute $\boxed{IP_2(x_1, \dots, x_N, y_1, \dots, y_N) = \sum_{i=1}^{N} x_i \cdot y_i \pmod 2}$ with only 1-bit classical communication from Alice to Bob.

Algorithm:

1) Alice & Bob each input their respective x_i & y_i bits into their respective non-local boxes, which respectively outputs a_i & b_i such that $a_i + b_i = x_i \cdot y_i \pmod 2$.

2) Alice then computes the partial sum of her outputs $a_1, a_2, \dots, a_N$ & so does Bob for b_i, both $\pmod 2$.
Let Alice's sum be denoted by $\boxed{A = \sum_{i=1}^{N} a_i \pmod 2}$ & Bob's sum by $\boxed{B = \sum_{i=1}^{N} b_i \pmod 2}$

3) Alice then sends 'A' bit to Bob, who computes $\boxed{A + B \pmod 2}$.

$$A + B \pmod 2 = \sum_{i=1}^{N} a_i \pmod 2 + \sum_{j=1}^{N} b_j \pmod 2 = \sum_{i=1}^{N} (a_i + b_i) \pmod 2$$
$$= \sum_{i=1}^{N} x_i \cdot y_i \pmod 2$$

Thus, $\boxed{A + B \pmod 2 = IP_2(x_1, \dots, x_N, y_1, \dots, y_N)}$

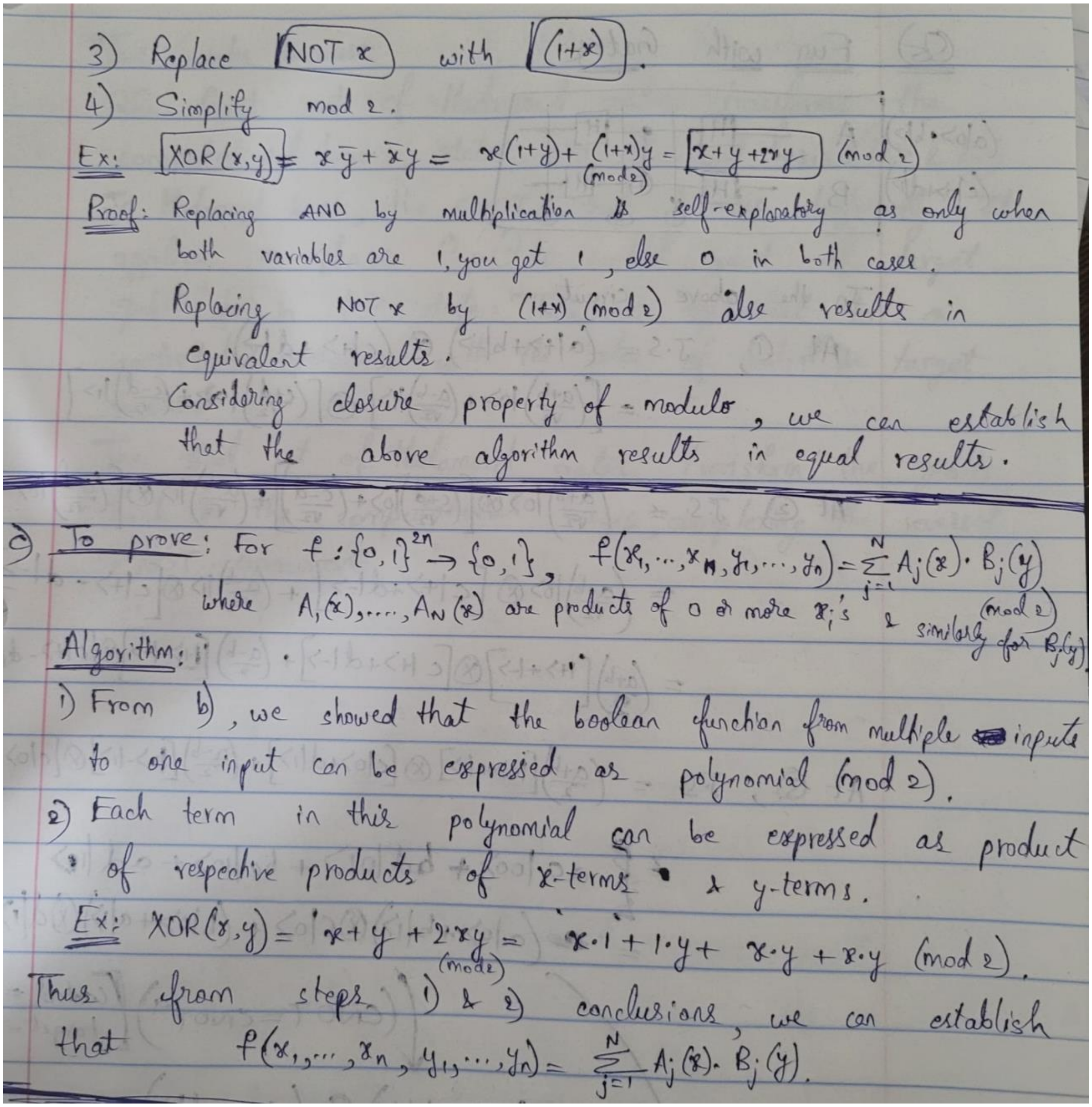
Here Alice sent only only one-bit of classical information to Bob.

b) Computing Boolean functions as polynomials (mod 2):

Algorithm (To express any boolean function as a polynomial (mod 2)):

1) Express the function as a sum of minterms (i.e, Disjunctive normal form (DNF)).

2) Replace $\boxed{\text{AND}}$ with $\boxed{\text{multiplication}}$.



3) Replace $\boxed{\text{NOT } x}$ with $\boxed{(1+x)}$.

4) Simplify mod 2.

Ex: $\boxed{XOR(x,y)} = x\bar{y} + \bar{x}y = x(1+y) + (1+x)y \underset{(\text{mod} 2)}{=} \boxed{x+y+2xy}$ (mod 2).

Proof: Replacing AND by multiplication is self-explanatory as only when both variables are 1, you get 1, else 0 in both cases.

Replacing NOT x by $(1+x)$ (mod 2) also results in equivalent results.

Considering closure property of modulo, we can establish that the above algorithm results in equal results.

c) To prove: For $f : \{0,1\}^{2n} \to \{0,1\}$, $f(x_1, \dots, x_n, y_1, \dots, y_n) = \sum_{j=1}^{N} A_j(x) \cdot B_j(y)$ (mod 2)

where $A_1(x), \dots, A_N(x)$ are products of 0 or more x_i's & similarly for $B_j(y)$

Algorithm:

1) From b), we showed that the boolean function from multiple inputs to one input can be expressed as polynomial (mod 2).

2) Each term in this polynomial can be expressed as product of respective products of x-terms & y-terms.

Ex: $XOR(x,y) = x + y + 2 \cdot xy \underset{(\text{mod} 2)}{=} x \cdot 1 + 1 \cdot y + x \cdot y + x \cdot y$ (mod 2).

Thus from steps 1) & 2) conclusions, we can establish that $f(x_1, \dots, x_n, y_1, \dots, y_n) = \sum_{j=1}^{N} A_j(x) \cdot B_j(y)$.

d) <u>To show</u>: It is possible to compute $f(x_1,\dots,x_N,y_1,\dots,y_N)$ by 1-bit classical communication from Alice to Bob on the condition that they can use indefinite number of magical non-local boxes.

Algorithm:

1) Express f as $\sum_{i=1}^{N}\left[A_i(x)\cdot B_i(y)\right] \pmod 2$ as in part-c).

2) Alice computes all $A_i(x)$'s & Bob computes all $B_i(y)$'s

3) They use 'N' non-local boxes to:
   - Alice inputs $A_i(x)$ into i^{th} box & gets output a_i.
   - Bob inputs $B_i(y)$ into i^{th} box & gets output b_i.

4) Alice computs $A = a_1+a_2+\dots+a_N \pmod 2$ & sends this single bit to Bob.

5) Bob computes $B = b_1+b_2+\dots+b_N \pmod 2$

6) Bob can now compute $f(x_1,\dots,x_N,y_1,\dots y_N) = A+B \pmod 2$