

The Tiling Method

Max von Hippel

March 23, 2021

Consider a two-party communication problem, in which the participants



(a) Alice

and

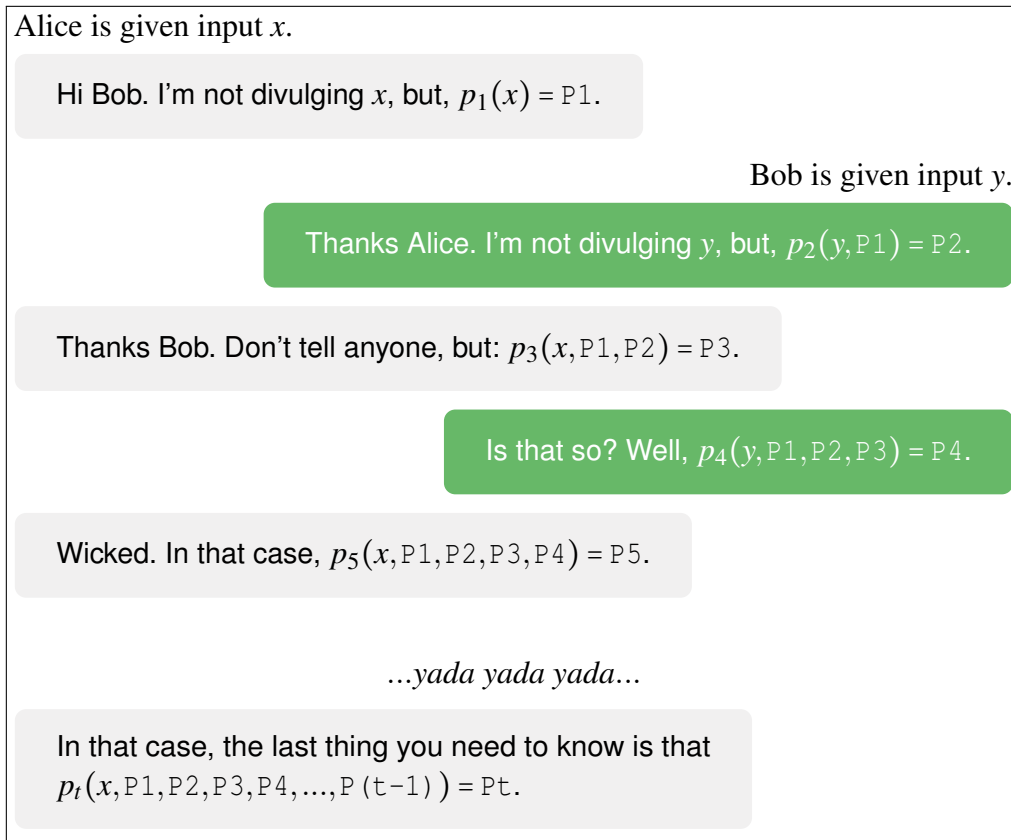


(b) Bob

participate to compute a function:

$$f: \underbrace{\mathbb{B}^n}_{\text{Alice's input}} \times \underbrace{\mathbb{B}^n}_{\text{Bob's input}} \rightarrow \underbrace{\mathbb{B}}_{\text{global output}}$$

The players can come up with a *protocol* $\Pi = (p_1, \dots, p_t)$, namely, for some natural $t \in \mathbb{N}$, a sequence of t -many functions $p_i: \mathbb{B}^* \rightarrow \mathbb{B}^*$ such that the communication between the players looks like this:



Suppose that there is a protocol Π for f consisting of t messages, but, there does not exist any protocol Π' for f consisting of fewer than t messages. Then we say t is the *communication complexity* of f , and we write $C(f) = t$.

Given some such function f , it would be nice if we could automatically compute a reasonable lower bound on its communication complexity. One way to do this is with the *tiling method*. We will give the method immediately, and in tandem, we will illustrate the method using the function $f(x, y) = x < y$ where x, y are integers in $\{0, 1, 2, 3\}$, encoded in \mathbb{B} oolean. First, let $M(f)$ be the *matrix of f* , namely, the $2^n \times 2^n$ matrix whose (x, y) th entry is the value $f(x, y)$.

	0	1	2	3	4
0	0	1	1	1	1
1	0	0	1	1	1
2	0	0	0	1	1
3	0	0	0	0	1
4	0	0	0	0	0

Table 1: The matrix $f(<)$ for inputs $x, y \in \{0, 1, 2, 3\}$. Values of x are given in the rows, while values of y are given in the columns. False (i.e. 0) values are marked red for clarity.

TODO...