Abstract:

An Oblivious RAM (ORAM), introduced by Goldreich and Ostrovsky (JACM 1996), is a (probabilistic) RAM that hides its access pattern, i.e. for every input the observed locations accessed are similarly distributed. Great progress has been made in recent years in minimizing the overhead of ORAM constructions, with the goal of obtaining the smallest overhead possible.

We revisit the lower bound on the overhead required to obliviously simulate programs, due to Goldreich and Ostrovsky. While the lower bound is fairly general, including the offline case, when the simulator is given the reads and writes ahead of time, it does assume that the simulator behaves in a “balls and bins” fashion. That is, the simulator must act by shuffling data items around, and is not allowed to have sophisticated encoding of the data.

We prove that for the OFFLINE case, showing a lower bound without the above restriction is related to the size of the circuits for sorting. Our proof is constructive, and uses a bit-slicing approach which manipulates the bit representations of data in the simulation. This implies that without obtaining yet unknown superlinear lower bounds on the size of such circuits, we cannot hope to get lower bounds on offline (unrestricted) ORAMs.