Memory-Efficient Algorithms for Finding Needles in Haystacks

Abstract:

One of the most common tasks in cryptography and cryptanalysis is to find some interesting event (a needle) in an exponentially large collection (haystack) of \(N=2^n\) possible events, or to demonstrate that no such event is likely to exist. In particular, we are interested in finding needles which are defined as events that happen with an unusually high probability of \(p>>1/N\) in a haystack which is an almost uniform distribution on \(N\) possible events. When the search algorithm can only sample values from this distribution, the best known time/memory tradeoff for finding such an event requires \(O(1/Mp^2)\) time given \(O(M)\) memory.

In this talk I will describe much faster needle searching algorithms in the common cryptographic setting in which the distribution is defined by applying some deterministic function \(f\) to random inputs. Such a distribution can be modeled by a random directed graph with \(N\) vertices in which almost all the vertices have \(O(1)\) predecessors while the vertex we are looking for has an unusually large number of \(O(pN)\) predecessors. When we are given only a constant amount of memory, we propose a new search methodology which we call NestedRho. As \(p\) increases, such random graphs undergo several subtle phase transitions, and thus the log-log dependence of the time complexity \(T\) on \(p\) becomes a piecewise linear curve which bends four times. Our new algorithm is faster than the \(O(1/p^2)\) time complexity of the best previous algorithm in the full range of \(1/N < p < 1\), and in particular it improves the previous time complexity by a significant factor of \(\sqrt{N}\) for any \(p\) in the range \(N^{(-0.75)} < p < N^{(-0.5)}\). When we are given more memory, we show how to combine the NestedRho technique with the parallel collision search technique in order to further reduce its time complexity. Finally, we show how to apply our new search technique to more complicated distributions with multiple peaks when we want to find all the peaks whose probabilities are higher than \(p\).

Joint work with Itai Dinur, Orr Dunkelman and Nathan Keller.