I Probabilistically Checkable Proofs

1. P. Harsha and M. Sudan.
   Small PCPs with low query complexity.
   (Preliminary Version in 18th STACS, 2001).

   Most known constructions of probabilistically checkable proofs (PCPs) either blow up the proof size by a large polynomial or have a high (though constant) query complexity. For instance, the proof size could be decreased to almost linear (i.e., $n^{1+\epsilon}$), but at the expense of a large query complexity (e.g., in excess of $10^6$) (see Polishchuk and Spielman). On the other hand, the PCP constructions of Hastad achieve the optimal query complexity of 3, but at the expense of blowing up the proof size by a very large exponent.

   In this paper we give a transformation with slightly-super-cubic blowup in proof size, with a low query complexity. Specifically, the verifier probes the proof in 16 bits and rejects every proof of a false assertion with probability arbitrarily close to 1/2, while accepting correct proofs with probability one. The proof is obtained by revisiting known constructions and improving numerous components therein.

   Robust PCPs of proximity, shorter PCPs and applications to coding.
   (Preliminary Version in 36th STOC, 2004).

   In this paper, we study the trade-off between the length of probabilistically checkable proofs (PCPs) and their query complexity. While doing so, we revisit the proof of the PCP Theorem and introduce a new variant of PCPs, that we call robust PCPs of proximity. These new PCPs facilitate proof composition, a central ingredient in the construction of PCP systems. These new PCPs, besides allowing for a much simpler construction of PCPs, also naturally lend themselves to the construction of locally testable codes and a relaxed notion of locally decodable codes.

   The main quantitative results of the papers are as follows (these results refer to proofs of satisfiability of circuits (of size $n$))
   (a) We present PCPs of length $n \cdot \exp(o(\log \log n)^2)$ that can be verified by making $o(\log \log n)$ Boolean queries.
   (b) For every $\epsilon > 0$, we present PCPs of length $n \cdot 2^{\log^\epsilon n}$ that can be verified by making a constant number of Boolean queries.

   In both cases, false assertions are rejected with constant probability (which may be set to be arbitrarily close to 1). The multiplicative overhead on the length of the proof, introduced by transforming a proof into a probabilistically checkable one, is just quasi polylogarithmic in the first case (of query complexity $o(\log \log n)$), and is $2^{(\log n)^\epsilon}$, for any $\epsilon > 0$, in the second case (of constant query complexity).

   Short PCPs verifiable in polylogarithmic time.
In this paper, we show that the recent improvements in the construction of short PCPs can be accompanied by efficient verifiers (efficient with respect to running time). The time complexity of the verifier and the size of the proof were the original emphases in the definition of holographic proofs, due to Babai et al. (STOC 91), and our work is the first to return to these emphases since their work. It is to be noted that all short PCP constructions after the work of Babai et al. achieved their improvements with respect to PCP size by sacrificing efficiency of the verifier. We show that this need not be the case and give efficient (in the sense of running time) versions of the shortest known PCPs, due to Ben-Sasson et al. (STOC 04) and Ben-Sasson and Sudan (STOC 05), respectively.

More formally, we show that every language in NP has a probabilistically checkable proof of proximity (i.e., proofs asserting that an instance is “close” to a member of the language), where the verifiers running time is polylogarithmic in the input size and the length of the probabilistically checkable proof is only polylogarithmically larger that the length of the classical proof.

4. E. Ben-Sasson, P. Harsha, O. Lachish, and A. Matsliah. 
   *Sound 3-query PCPPs are long*, 2007. 

In this paper, we ask the question if there exists a 3-query probabilistically checkable proof (PCP) that is simultaneously short (i.e., polynomial sized) and has maximal soundness that can be guaranteed by a 3-query verifier? We show that the answer is negative in the case of probabilistically checkable proofs of proximity (PCPPs). More precisely, our main result is that a PCPP verifier limited to querying a short proof cannot obtain the same soundness as that obtained by a verifier querying a long proof. A language is said to have a PCPP if there exists a probabilistic verifier that can distinguish inputs in the language from inputs that are far from the language by merely probing the input and an additional proof at a few locations. Since all known constructions of PCPs yield PCPPs, the above negative result shows that completely new techniques are required to construct PCPs that are both simultaneously short and maximally sound.

II Property Testing

5. E. Ben-Sasson, P. Harsha, and S. Raskhodnikova. 
   *Some 3CNF properties are hard to test.* 

The main result of this paper demonstrates that there are some properties which are very easy to decide, but are however very hard to test. Property testing deals with the question of how many queries are needed to distinguish (with high probability) between a input that satisfies a given property and one that is $\epsilon$-far (in Hamming distance) from satisfying the given property. We show (by a probabilistic argument) the existence of a property expressible by a 3CNF formula, such that the number of queries required for any test is the worst possible, that is, linear in the total number of variables. We prove this result using two intermediate results which are interesting in their own right.

(a) A random low density parity check (LDPC) code is not locally testable (with high probability).
(b) For testing linear properties (i.e., properties expressible by linear constraints), adaptivity and 2-sided error do not help.
III Information Theory and Communication Complexity

   The communication complexity of correlation.
   Diego, California, 13–16 June 2007.

In this paper, we ask the following question. Let $(X, Y)$ be a joint distribution. Suppose Alice
is given a sample $x$ distributed according to $X$ and needs to send a message $z$ to Bob so that he
can generate a correlated sample $y$ distributed according to the conditional distribution $Y|_{X=x}$.
What is the minimum number of bits (i.e., $|z|$) that Alice needs to transmit in order to achieve
this? Clearly, Alice needs to send at least the mutual information $I[X : Y]$ number of bits. We
show that there are distributions $(X, Y)$ for which Alice needs to send exponentially many more
bits. However, if Alice and Bob, share a random string (independent of $(X, Y)$), then Alice need
send Bob no more than approximately $I[X : Y]$ number of bits. This result gives a nice alternate
characterization of mutual information (up to lower order logarithmic terms).

As an intermediate step, we prove that the greedy rejection sampling procedure to generate one
distribution from another gives a similar characterization of relative entropy.

This characterization of mutual information immediately yields an improved direct sum result in
communication complexity.

IV Proof Complexity

7. E. Ben-Sasson and P. Harsha.
   Lower bounds for bounded depth Frege proofs via Buss-Pudlák games.

In this paper, we give an exposition of the exponential lower bounds (of Pitassi et al. and Karijicek
et al.) for the pigeonhole principle in the bounded-depth Frege proof system. This exposition
uses the interpretation of proofs as two player games by Pudlak and Buss and is simpler than
earlier proofs and relies on tools and intuition that are well-known in the context of computation
complexity.

V Resource Tradeoffs

   Communication vs. computation.

In this paper, we initiate a study of tradeoffs between communication and computation in well-
known communication models and in other related models. The fundamental question we investi-
gate is the following: Is there a computational task that exhibits a strong tradeoff behavior
between the amount of communication and the amount of time needed for local computation?
Under various standard assumptions, we prove the existence of such strong tradeoffs for the
following computation models: (1) two-party randomized communication complexity; (2) query
complexity; (3) property testing. For instance, we show that there is a function $f$ such that
$f(x, y)$ is easy to compute (knowing $x$ and $y$), and has low communication complexities (when
one player knows $x$ and the other knows $y$). However, under reasonable complexity assumption,
all low-communication protocols for computing $f(x, y)$ are hard to compute.
VI Oblivious Network Routing

Minimizing average latency in oblivious routing. 

In this paper, we consider the problem of minimizing average latency cost while obliviously routing traffic in a network with linear latency functions. This is roughly equivalent to minimizing the function $\sum_e (\text{load}(e))^2$, where for a network link $e$, $\text{load}(e)$ denotes the amount of traffic that has to be forwarded by the link.

We show that for the case when all routing requests are directed to a single target, there is a routing scheme with competitive ratio $O(\log n)$, where $n$ denotes the number of nodes in the network. As a lower bound we show that no oblivious scheme can obtain a competitive ratio of better than $\Omega(\sqrt{\log n})$.

VII Automata Theory

10. K. Krithivasan, S. Balan, and P. Harsha. 
Distributed processing in automata. 

In this paper, we introduce the notion of distributed automata for finite state automata and push-down automata and analyze their acceptance power. Informally, a distributed automaton refers to a group of automata working in unison to accept one language. We compare the acceptance power of distributed automata to their centralized counterparts according to various types of acceptance modes. We show that distributed finite state automata do not have any additional power over the centralized ones, while distributed push-down automata with just two components is as powerful as Turing machines.